On the Importance of Word Order Information in Cross-lingual Sequence Labeling

Zihan Liu, Genta I Winata, Samuel Cahyawijaya,
Andrea Madotto, Zhaojiang Lin, Pascale Fung
Center for Artificial Intelligence Research (CAiRE)
The Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong
zihan.liu@connect.ust.hk, pascale@ece.ust.hk

Abstract

Cross-lingual models trained on source language tasks possess the capability to transfer to target languages directly. However, since word order variances generally exist in different languages, cross-lingual models that overfit into the word order of the source language could have sub-optimal performance in target languages. In this paper, we hypothesize that reducing the word order information fitted into the models can improve the adaptation performance in target languages. To verify this hypothesis, we introduce several methods to make models encode less word order information of the source language and test them based on cross-lingual word embeddings and the pre-trained multilingual model. Experimental results on three sequence labeling tasks (i.e., part-of-speech tagging, named entity recognition, and slot filling tasks) show that reducing word order information injected into the model can achieve better zero-shot cross-lingual performance. Further analysis illustrates that fitting excessive or insufficient word order information into the model results in inferior cross-lingual performance. Moreover, our proposed methods can also be applied to strong cross-lingual models and further improve their performance.

Introduction

Neural-based data-driven supervised approaches have achieved remarkable performance in sequence labeling tasks (e.g., named entity recognition) (Lample et al. 2016; Devlin et al. 2019). However, these methods are not applicable to low-resource languages, where extensive training data are absent. Recently, numerous cross-lingual adaptation methods have been applied to this data-scarcity scenario, where zero or very few target language training samples are utilized (Wisniewski et al. 2014; Schuster et al. 2019b; Artetxe and Schwenk 2019; Liu et al. 2019; Chen et al. 2019).

Despite the focus on cross-lingual methods, the word order differences across languages is a less studied problem of the cross-lingual task. For cross-lingual models, sequence encoders that are based on LSTM (Hochreiter and Schmidhuber 1997) or Transformer (Vaswani et al. 2017) inevitably model the word order information in the source language (Xie et al. 2018; Liu et al. 2019), which we characterize the *order-sensitive* property (Ahmad et al. 2018). Since different languages have different word orders, models that fit into the word order of the source language could hurt the performance in the target languages.

In this paper, we hypothesize that making models less sensitive to word orders can boost cross-lingual performance, and then we introduce four methods to construct *order-reduced* models to verify this hypothesis. First, we propose an Order-Reduced Transformer (ORT), which removes the positional embeddings from Transformer and utilizes one-dimensional convolutional networks to replace the linear layer as the feed-forward layer to encode partial order information; thus, the model becomes less dependent on the word order. Second, we permutate the word order of the training samples, and order-sensitive models trained with them become insensitive to the word order. Third, we hypothesize that the positional embeddings in multilingual BERT (M-BERT) (Devlin et al. 2019) are order-agnostic given the surprising cross-lingual ability that it has (Wu and Dredze 2019). Hence, we take the positional embeddings from M-BERT to initialize the positional embeddings in Transformer, and we freeze them in the training phase to make the model order-agnostic. Additionally, based on the third method, we propose to freeze the positional embeddings when we fine-tune M-BERT to downstream cross-lingual tasks, which makes the model avoid fitting into the word order of the source language.

We conduct experiments on zero-shot cross-lingual sequence labeling tasks, namely, part-of-speech tagging (POS), named entity recognition (NER), and slot filling (SF),[1], and we compare our models with order-sensitive sequence encoders, such as LSTM and Transformer. We summarize our insights as follows:

- Order-reduced models are robust to word order shuffled sequences and consistently outperform order-sensitive models, including the state-of-the-art model.
- Retaining the order-agnostic property of M-BERT positional embeddings gives a better capability to generalize to target languages.
- Encoding excessive or insufficient word order information leads to sub-optimal cross-lingual performance, and

[1] SF is a critical task in natural language understanding (NLU) for dialog systems.

models that do not encode any word order information (i.e., the most insensitive to word order) give a poor performance on both source and target languages.

Related Work

Cross-lingual Adaptation

Recently, cross-lingual sequence labeling methods that circumvent the need for extensive training data in target languages have achieved remarkable performance (Kim, Snyder, and Sarikaya 2015; Kim et al. 2017; Ni, Dinu, and Florian 2017; Mayhew, Tsai, and Roth 2017; Liu et al. 2020a,b; Huck, Dutka, and Fraser 2019). Chen et al. (2019) utilized the similarity between the target language and each individual source language to achieve promising results on the cross-lingual NER task, while Liu et al. (2019) utilized task-related keywords to build robust cross-lingual natural language understanding (NLU) systems. Taking this further, cross-lingual language models (Pires, Schlinger, and Garrette 2019; Lample and Conneau 2019; Huang et al. 2019; Conneau et al. 2019; Liang et al. 2020) pre-trained on a large data corpus achieved the state-of-the-art performance in multiple cross-lingual adaptation tasks.

Coping with Word Order Differences

Word order differences across languages have been considered in cross-lingual dependency parsing (Tiedemann and Agic 2016; Zhang, Zhang, and Fu 2019) by using Treebank translation. For the same task, Ahmad et al. (2018), on the other hand, leveraged a relative positional self-attention encoder (Shaw, Uszkoreit, and Vaswani 2018) to make the sequence encoder less sensitive to word order and increase the adaptation robustness for target languages that are topologically different from the source language. Compared to the previous works, we conduct extensive experiments and analyses to illustrate the effectiveness of order-reduced models for cross-lingual sequence labeling tasks.

Methodology

In this section, we introduce the proposed methods to reduce the word order of the source language fitted to order-sensitive sequence encoders.

Order-Reduced Transformer

Given that Transformer (Vaswani et al. 2017) relies on positional embeddings to encode word order information, we propose to remove them so as to reduce the word order information injected into input sentences. Note that given a linear layer as the feed-forward layer for Transformer, as introduced in Vaswani et al. (2017), removing the positional embeddings module would mean getting rid of all the word order information, which would lead to a large performance drop in the source language and low performance for the cross-lingual transfer. Therefore, we utilize the one-dimensional convolutional network (Conv1d) (Kim 2014) as the feed-forward layer to extract n-gram features from the Multi-Head Attention features. Specifically, we formulate the encoding process as follows:

$$g[1:n] = \texttt{MultiHead}(E(X[1:n])), \tag{1}$$

where $X[1:n]$ represents the n-token input sequence; E denotes the embedding layer; and $g[1:n] \in R^{n \times d}$, where d is the hidden size of Transformer, represents the sequence features generated by Multi-Head Attention.

After that, a feature c_i is generated from the window of features $g[i:i+h-1]$ by

$$c_i = \texttt{Conv1d}(g[i:i+h-1]), \tag{2}$$

where h is the kernel size of Conv1d and the dimension of c_i is equal to the number of output channels in Conv1d. We add padding for this convolution process to ensure the output feature length is the same as the length of the input tokens. Finally, the output feature sequences from Conv1d are the concatenation of c_i, where $i \in [1,n]$.

In this way, we fit the model with less word order information since the model only encodes the local n-gram features, and the prediction for each token is made based on the token itself and its neighbor tokens.

Shuffling Word Order

Instead of removing positional embeddings, we propose to permutate the word order of input sequences in the source language training samples so as to train models to be robust for different word orders. Meanwhile, we keep the order of tokens in each entity the same and consider them as one "word" to ensure we don't break entities in the sequences.

We follow Lample et al. (2018) to generate permutations similar to the noise observed with word-by-word translation (i.e., word order differences across languages). Concretely, we apply a random permutation σ to the input sequence, verifying the condition $\forall i \in \{1, n\}, |\sigma(i) - i| \leq k$, where n is the length of the input sentence and k is a tunable parameter that controls the shuffling degree. We use the order-shuffled training samples to train the models and make them less sensitive to word orders.

Order-Agnostic Positional Embeddings

Another alternative method is to make the positional embeddings of Transformer order-agnostic to encode less order information. In light of M-BERT's astonishing cross-lingual performance (Pires, Schlinger, and Garrette 2019; Wu and Dredze 2019), we speculate that the positional embeddings in M-BERT are order-agnostic. Hence, we leverage M-BERT's positional embeddings to initialize the positional embeddings for Transformer, and we freeze them in the training phase to prevent them from fitting into the source language word order. In the experiments, since M-BERT's positional embeddings are based on its tokenizer, we leverage M-BERT's tokenizer to tokenize sequences and generate cross-lingual embeddings from M-BERT. Then, a Transformer encoder with M-BERT's positional embeddings is added on top of the M-BERT embeddings. We freeze the parameters of M-BERT in the training phase to ensure the cross-lingual embeddings from M-BERT do not fit into the source language word order.

Fine-tuning M-BERT

The original fine-tuning of M-BERT to downstream cross-lingual tasks is done by adding a linear layer on top of M-

	Named Entity Recognition Task					Slot Filling Task			
	en	**es**	**nl**	**de**	**avg**	**en**	**es**	**th**	**avg**
Dist. to English	0.00	0.12	0.14	0.14	0.13	0.00	0.12	0.31	0.22
Frozen Word-level Embeddings									
BiLSTM	87.99	33.71	25.28	15.28	24.76	**94.87**	59.51	20.63	40.07
w/ shuffled data	83.85	30.09	22.87	13.22	22.06	93.57	62.02	21.43	41.73
Transformer (TRS)	**88.67**	30.76	30.54	18.53	26.61	94.78	62.67	22.33	42.50
w/ shuffled data	82.75	28.54	28.43	16.17	24.38	92.07	63.86	24.17	44.02
Ahmad et al. (2018)	87.86	32.49	31.83	19.24	27.85	94.23	62.07	23.14	42.61
ORT	88.41	**34.33**	**33.54**	**24.14**	**30.67**	94.50	**66.84**	**25.53**	**46.19**
Frozen M-BERT Embeddings									
Transformer (TRS)	89.53	58.93	46.28	63.15	56.12	**94.93**	46.75	9.76	28.26
w/ M-BERT PE	88.44	58.27	**47.63**	64.12	56.67	94.53	47.23	**10.06**	28.65
Ahmad et al. (2018)	**89.96**	**60.55**	45.43	61.58	55.85	94.38	47.80	8.83	28.32
ORT	89.46	58.35	45.95	**66.31**	**56.87**	94.55	**48.42**	9.92	**29.17**
M-BERT Fine-tuning									
Fine-tune M-BERT	**91.95**	74.49	69.13	77.32	73.65	**95.97**	69.41	10.45	39.93
w/ frozen PE	91.87	**74.98**	**70.22**	**77.63**	**74.28**	95.90	**70.30**	**12.53**	**41.42**

Table 1: Zero-shot cross-lingual results on NER and SF tasks (averaged over three runs) for the three settings. We freeze the word-level embeddings in the training stage to ensure their cross-lingual alignment is preserved. We use "w/ shuffled data" to denote the models trained with the word order shuffled source language training samples. "PE" denotes positional embeddings, and we use "w/ M-BERT PE" to represent that the model initialized with the frozen M-BERT positional embeddings. "avg" denotes the average performance over the target languages (English is excluded).

BERT and fine-tuning all the parameters of the model to the source language task (Pires, Schlinger, and Garrette 2019; Wu and Dredze 2019). This inescapably fits the model with the source language word order. To circumvent this issue, we freeze the positional embeddings in M-BERT in the fine-tuning stage. By doing so, the positional embeddings can still provide the word order information for M-BERT to encode input sequences, and the model avoids fitting the word order of the source language.

Experiments

Datasets

We test our methods on three sequence labeling tasks in the cross-lingual setting, namely, part-of-speech tagging (POS), named entity recognition (NER), and slot filling (SF). For the POS task, we choose the same language set as Ahmad et al. (2018) (31 languages in total) from the Universal Dependencies (Nivre et al. 2017) to evaluate our methods. And we use the CoNLL 2002 and CoNLL 2003 datasets (Tjong Kim Sang 2002; Sang and De Meulder 2003), which contain English (en), German (de), Spanish (es), and Dutch (nl), to evaluate our methods for the NER task. Finally, for the SF task, we use the multilingual natural language understanding (NLU) (containing the intent detection and slot filling tasks) dataset introduced by Schuster et al. (2019a), which contains English (en), Spanish (es), and Thai (th) across weather, alarm and reminder domains. The data statistics for these datasets are in the appendix.

Experimental Settings

Our Models and Baselines All our models and baseline models consist of a sequence encoder to produce features for input sequences and a conditional random field (CRF) layer (Lample et al. 2016; Ma and Hovy 2016) to make predictions based on the sequence features. For the sequence encoder, we use Bidirectional LSTM (**BiLSTM**), Transformer (**TRS**) using sinusoidal functions as positional embeddings, Relative Positional Transformer (**RPT**) proposed in Ahmad et al. (2018)[2], or Order-Reduced Transformer (**ORT**). All Transformer-based encoders use Conv1d as the feed-forward layer for a fair comparison. Word order shuffling is applied to BiLSTM and TRS baselines to make them less sensitive to word orders. We fine-tune M-BERT by adding a linear layer on top of it, and we compare two different fine-tuning M-BERT methods (with and without freezing the positional embeddings).

Training Details We evaluate our models with cross-lingual word embeddings (word-level) and M-BERT embeddings (subword-level). For the word-level embeddings, we leverage RCSLS (Joulin et al. 2018) for the POS and NER tasks, and we use the refined RCSLS in Liu et al. (2019) for the SF task since it is specifically refined for this task. We set the kernel size as 3 for the feed-forward layer Conv1d in the Transformer encoder. For the word order shuffled data, we generate ten different word order shuffled samples with $k = \infty$ (can generate any permutation) for each source language training sample. Note that the word order shuffling can not be applied for M-BERT-based models since they are pre-trained based on the correct language order, and it is not suitable to feed them with order-shuffled sequences. For all the tasks, we use English as the source language and other languages as target languages. We follow Ahmad et al. (2018) to calculate the language distances between target languages and English. We use the standard BIO-based F1-score for evaluating the NER and SF tasks, as in Lample et al. (2016), and accuracy score for evaluating the POS task, as in Kim et al. (2017). More details are in the appendix.

[2] They utilize the relative positional embeddings proposed in Shaw, Uszkoreit, and Vaswani (2018) to encode less word order information for cross-lingual adaptation.

Lang	**Dist. to English**	**Frozen Word-level Embeddings**						**Frozen M-BERT Embeddings**				**M-BERT Fine-tuning**	
		BiLSTM	BiLSTM w/ shuffled data	TRS	TRS w/ shuffled data	RPT	ORT	TRS	TRS w/ M-BERT PE	RPT	ORT	M-BERT	M-BERT w/ frozen PE
en	0.00	**93.76**	87.66	93.07	86.84	92.77	93.74	92.73	92.36	92.47	**93.07**	97.20	**97.21**
no	0.06	34.48	23.50	44.29	40.37	44.45	**55.05**‡	65.19	66.34	**68.44**	65.73	75.72	**76.11**
sv	0.07	27.74	21.83	43.83	29.66	39.66	**56.92**‡	74.84	76.38	**76.47**	75.75	85.02	**85.48**
fr	0.09	28.92	22.29	50.70	40.16	47.13	**62.72**‡	76.06	77.27	**79.62**	79.24‡	88.57	**88.82**
pt	0.09	41.34	29.71	59.00	48.78	55.14	**66.77**	83.23	83.94	**84.94**	84.64	90.66	**91.10**
da	0.10	51.77	37.61	49.14	49.18	56.87	**63.21**‡	77.69	78.77	**79.53**	79.02	87.19	**87.61**
es	0.12	41.86	31.49	51.82	44.58	50.82	**60.29**	76.58	**79.43**‡	77.93	77.91	86.56	**86.88**
it	0.12	37.14	24.65	54.78	42.88	49.93	**66.55**‡	73.59	**77.46**‡	77.21	77.38‡	88.98	**89.85**‡
hr	0.13	29.35	22.11	45.88	38.85	45.87	**55.40**‡	68.56	71.12‡	**71.70**	71.08‡	82.57	**83.45**‡
ca	0.13	38.20	26.02	50.97	46.24	49.66	**59.56**	74.61	**78.22**‡	75.84	77.96‡	85.85	**86.11**
pl	0.13	43.13	31.41	49.16	32.72	52.02	**62.97**‡	66.93	**68.73**	65.82	68.53	80.11	**80.61**
uk	0.13	29.60	28.99	42.70	35.14	49.40	**56.13**‡	73.45	**75.15**	73.23	75.08	83.83	**84.41**
sl	0.13	33.67	29.08	43.94	40.00	42.80	**60.01**‡	62.76	64.68	**67.42**	64.49	75.71	**76.58**
nl	0.14	34.95	24.08	48.69	34.87	45.02	**63.30**‡	79.28	79.07	79.48	**80.08**	86.90	**87.68**
bg	0.14	28.42	24.71	39.78	29.98	43.98	**54.49**‡	68.63	**70.13**	68.93	69.42	**80.66**	80.60
ru	0.14	29.65	26.83	47.96	40.12	51.53	**59.20**‡	78.39	79.53	77.88	**79.76**	88.80	**89.05**
de	0.14	31.52	25.12	43.24	35.01	41.11	**50.59**	68.41	67.75	**69.00**	68.96	**81.02**	80.66
he	0.14	21.67	19.74	34.82	24.38	33.97	**39.24**	64.30	66.12	65.22	**66.88**‡	69.74	**70.30**
cs	0.14	34.06	27.53	48.30	36.97	51.00	**63.79**‡	75.05	**76.46**	75.17	76.17	85.71	**86.20**
ro	0.15	35.05	28.36	45.43	41.10	46.29	**61.23**‡	69.17	**72.42**‡	71.93	71.37‡	81.38	**82.03**
sk	0.17	39.98	35.62	47.66	38.46	50.28	**61.10**‡	68.13	**69.21**	68.77	68.46	80.50	**81.23**
id	0.17	32.54	24.09	34.11	26.76	**41.61**	39.10	59.97	**62.08**‡	60.13	62.07‡	71.80	**72.86**‡
lv	0.18	49.05	35.21	52.43	50.66	**58.80**	57.87	64.74	**68.46**‡	66.81	67.94‡	79.01	**79.64**
fi	0.20	37.88	29.52	44.83	38.53	47.16	**53.85**	68.65	**72.04**‡	69.22	71.87‡	81.09	**81.98**‡
et	0.20	30.13	26.50	41.86	25.64	42.00	**49.81**	57.50	**63.00**‡	58.27	62.29‡	75.66	**75.80**
zh	0.23	26.66	23.95	28.92	24.81	27.94	**31.20**	53.37	55.58‡	53.51	**55.66**‡	63.86	**64.06**
ar	0.26	7.75	12.40	**25.77**	4.32	24.97	22.29	25.94	**29.96**‡	24.39	29.01‡	27.68	**36.94**‡
la	0.28	-	-	-	-	-	-	42.22	**46.11**‡	43.32	43.34	45.52	**46.96**‡
ko	0.33	10.79	7.87	**21.82**	5.16	16.71	19.02	35.77	38.17‡	36.08	**39.97**‡	42.18	**45.45**‡
hi	0.40	20.97	17.68	27.66	25.11	29.00	**35.05**	47.95	**52.76**‡	49.45	51.90‡	**56.45**	55.63
ja	0.49	-	-	-	-	-	-	40.59	**43.17**‡	41.46	42.82‡	44.97	**45.08**
avg	0.17	32.44	25.64	43.55	34.66	44.11	**53.10**	64.72	**66.98**	65.91	66.83	75.12	**75.97**

Table 2: Zero-shot cross-lingual results on the POS task (averaged over three runs). Languages are sorted by the word-ordering distance to English. Since the word-level embeddings for la and ja languages are absent, we do not report these results. We use '‡' to denote the performance improvement of the proposed models is higher than their corresponding average improvements.

Applying ORT into Strong Baselines We apply the ORT into two strong cross-lingual models for zero-shot cross-lingual NLU (Liu et al. 2019) and NER (Chen et al. 2019), and both are based on BiLSTM as the sequence encoder, which is order-sensitive. To ensure a fair comparison, we keep all settings as in the original papers, except that the sequence encoder is replaced. For the NLU model from Liu et al. (2019), we replace the BiLSTM with ORT. And for the NER model from Chen et al. (2019), we replace the BiLSTM in the shared feature extractor module with ORT.

Results & Discussion

Zero-shot Adaptation

Order-Reduced Transformer As we can see from Table 1, removing positional embeddings from Transformer (ORT) only makes the performance in the source language (English) drop slightly (around 0.5%). This indicates that leveraging only local order information results in a good performance in sequence labeling tasks. In other words, relying just on the information from the neighboring words (how many neighboring words depend on the kernel size in Conv1d) can ensure relatively good performance for sequence labeling tasks. On the other hand, in terms of zero-shot adaptation to target languages (from Table 1 and 2), ORT achieves consistently better performance than the order-sensitive encoders (i.e., BiLSTM and TRS) as well as the order-reduced encoder RPT (Ahmad et al. 2018). For example, in the SF task, in terms of the average performance of using word-level embeddings, ORT outperforms BiLSTM, TRS, and RPT by 6.12%, 3.69%, and 3.58% on the F1-score, respectively.

Compared to the order-sensitive models, ORT fits the word order of the source language less, which increases its adaptation robustness to target languages. We conjecture that the reason why ORT outperforms RPT is that RPT still keeps the relative word distances. Although RPT reduces the order information that the model encodes, it might not be suitable for target languages that do not have similar relative word distance patterns to English, while ORT removes all the order information in positional embeddings, which makes it more robust to the word order differences.

Shuffling Word Order From Table 1 and 2, we can see that the models trained with word order shuffled data lead to a visible performance drop in English, especially for the POS and NER tasks. For target languages, however, we observe that the performance improves in the SF task by using such data. For example, using the order shuffled data to train the Transformer improves the performance by 1.52%

	Spanish		Thai	
	ID	SF	ID	SF
Liu et al. (2019)	90.20	65.79	73.43	32.24
using TRS	89.71	67.10	74.68	31.20
using ORT	**91.46**	**71.36**	**75.02**	**34.61**

Table 3: Zero-shot results for the intent detection (ID) accuracy and SF F1-score on the NLU task.

	de	es	nl	avg
Chen et al. (2019)	56.00	73.50	72.40	67.30
using TRS	56.89	73.72	72.22	67.61
using ORT	**58.97**	**74.65**	**72.56**	**68.73**

Table 4: Zero-shot results on the NER task.

	$k = 0$	$k = 1$	$k = 2$
BiLSTM	**94.87**	85.16	83.68
TRS	94.78	84.56	83.06
RPT	94.23	84.93	83.86
ORT	94.50	**87.87**	**86.95**

Table 5: Slot F1-scores on different noisy SF test sets in English. $k = 0$ denotes the original English test set.

on the averaged F1-score. For cross-lingual adaptation, performance loss in the source language has a negative impact on the performance in target languages. In the SF task, the performance drop in English is relatively small (∼2%); hence, the benefits from being less sensitive to word orders are greater than the performance losses in English.

On the other hand, for the NER and POS tasks, using order shuffled data makes the performance in target languages worse. For example, for the POS task, the average accuracy drops 8.89% for the Transformer trained with the order shuffled data compared to the one trained without such data. We observe large performance drops for the NER and POS tasks in English caused by using the order shuffled data (for example, for the POS task, the drop is ∼6%) since the models for these tasks are more vulnerable to the shuffled word order. In this case, the performance losses in English are larger than the benefits of being less sensitive to word orders.

Order-Agnostic Positional Embeddings As we can see from Table 1 and 2, compared to TRS, we observe that TRS trained with M-BERT PE (frozen) only results in a slight performance drop in English, while it generally brings better zero-shot adaptation performance to target languages. For example, in the POS task, TRS with M-BERT PE achieves 2.26% higher averaged accuracy than the one without M-BERT PE. Since M-BERT is trained using 104 languages, positional embeddings in M-BERT are fitted to different word orders across various languages and become order-agnostic. Since the pre-trained positional embeddings are frozen, their order-agnostic property is retained, which brings more robust adaptation to target languages.

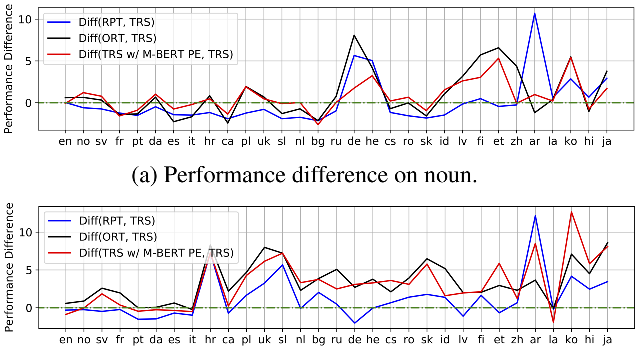
In addition, we notice that ORT achieves similar performance to TRS with M-BERT PE, which further illustrates the effectiveness of encoding partial order information for zero-shot cross-lingual adaptation.

Fine-tuning M-BERT From Table 1 and 2, we observe that the results of fine-tuning M-BERT in the source language, English, are similar for both methods (less than 0.1% difference) while freezing the positional embeddings in the fine-tuning stage generally brings better zero-shot cross-lingual performance in target languages. Although positional embeddings are frozen, they can still provide order information for the model to encode sequences, which ensures the performance in English does not greatly drop. In the meantime, the positional embeddings are not affected by the English word order, and the order-agnostic trait of the positional embeddings is preserved, which boosts the generalization ability to target languages.

Applying ORT to Strong Models As shown in Table 3 and 4, we leverage ORT to replace the order-sensitive encoder, BiLSTM, in the strong zero-shot cross-lingual sequence labeling models proposed in Liu et al. (2019) and Chen et al. (2019). The zero-shot cross-lingual NLU model proposed in Liu et al. (2019) is the current state-of-the-art for the multilingual NLU dataset (Schuster et al. 2019a), and the model in Chen et al. (2019) achieves promising results in the zero-shot cross-lingual NER task. As we can see, replacing the order-sensitive encoders in their models with ORT can still boost the performance. We conjecture that since there are always cross-lingual performance drops caused by word order differences, reducing the word order of the source language fitted into the model can improve the performance.

In addition, we observe that the performance stays similar when we replace BiLSTM with TRS, which illustrates that the performance improvement made by ORT does not come from TRS, but from the model's insensitivity to word order.

Improvements vs. Language Distance As we can see from Table 2, our proposed order-reduced models (e.g., ORT and TRS w/ M-BERT PE) outperform baseline models in almost all languages. In the word-level embeddings setting, we observe that languages that are closer to English benefit more from ORT, since most numbers denoted with '‡' come from languages that are close to English. We conjecture that ORT predicts the label of a token based on the local information of this token (the token itself and its neighbor tokens), and languages that are closer to English could have a more similar local word order to English. Interestingly, for M-BERT fine-tuning and models using M-BERT embeddings, we can see that languages that are further from English benefit more from order-reduced models. We speculate that the alignment quality of topologically close languages in M-BERT is generally good, and the cross-lingual performance for the languages that are closer to English is also overall satisfactory, which narrow down the improvement space for these languages. In contrast, the performance boost for languages that are further from English becomes larger. Surprisingly, freezing PE in the M-BERT fine-tuning significantly improves the performance of Arabic (ar). Given that the cross-lingual performance of the original M-BERT fine-tuning in Arabic is relatively low, we conjecture that one



(a) Performance difference on noun.

(b) Performance difference on verb.

Figure 1: Analysis on specific part-of-speech types. Languages are sorted by the word-ordering distance to English. We use Diff(A, B) to denote how much A outperforms B.

of the major reasons comes from the word order discrepancy between English and Arabic, which leads to a large performance improvement made by freezing the M-BERT PE.

How Order-Insensitive Is Our Model? To test the word order insensitivity of ORT, we follow the order shuffled methods in Section , and set $k = 1$ and $k = 2$ to slightly shuffle the word order of the sequences and create a noisy English test set. As we can see from Table 5, ORT achieves better results than BiLSTM, TRS, and RPT on the noisy SF test set, which further illustrates that ORT is more insensible and resistant to word order differences than the baseline encoders. This property improves the generalization ability of ORT to target language word orders.

Performance Breakdown by Types We compare different models (based on the frozen M-BERT embeddings) on specific part-of-speech types for the POS task. From Figure 1, we observe that, in general, languages that are further from English benefit more from our proposed order-reduced models, which accords with the findings from Table 2. Interestingly, we find that the improvements of order-reduced models (ORT and TRS w/ M-BERT PE) on the verb are larger than on the noun. We speculate that the word orders of the verb's surrounding words are different across languages, while the set of its surrounding words is more likely to remain the same or similar across languages at the semantic-level, which boost the advantages of our proposed models, especially for the ORT which relies greatly on the extracted n-gram features from the neighbor words for the prediction. Additionally, we find that the improvement made by RPT is also more significant on the verb than on the noun. We conjecture that the verb's relative positions with other part-of-speech types are more similar across languages compared to that for the noun. The experiments on more part-of-speech types are shown in the appendix.

Few-shot Adaptation

Since we do not observe the order information for target languages in the zero-shot scenario, the order-reduced models will have a more robust adaptation ability. Then, the question

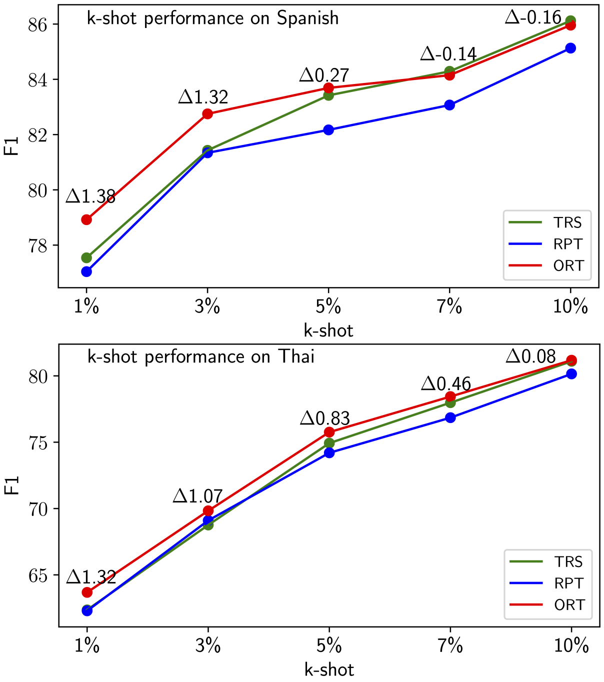


Figure 2: Few-shot F1-scores for the SF task for Spanish and Thai. The x-axis represents the proportion of target language training samples in the training set. The numbers with Δ denote how much ORT outperforms TRS.

we want to ask is whether order-reduced models can still improve the performance if a few training samples in target languages are available. We test with different numbers of target language training samples for the SF task, and the results are shown in Figure 2. We observe that the improvements in the few-shot scenarios are lower than the zero-shot scenario, where ORT improves TRS by 4.17% and 3.22% on Spanish and Thai, respectively (from Table 1), and as the proportion of target language training samples goes up, the improvement made by ORT goes down. This is because the model is able to learn the target language word order based on the target language training samples, which decreases the advantages of the order-reduced models. We also observe that RPT generally achieves worse performance than TRS, and we conjecture that RPT requires more training samples to learn the relative word order information than TRS, which lowers its generalization ability to the target language in the few-shot scenario.

Ablation Study

In this section, we explore the model variations in terms of positional embeddings, the feed-forward layer for TRS and ORT, adding different permutations to the shuffled word order, and whether to use the CRF layer. We test the models' zero-shot performance on the SF task for this ablation study, and results are illustrated in Table 6.

Positional Embeddings We observe that TRS+CRF using trainable positional embeddings achieves similar performance to using sinusoidal positional embeddings.

Feed-forward Layer We can see that the performance of TRS+CRF using linear layers as the feed-forward layer is on par with using Conv1d. We conjecture that the reason is that the positional embeddings in TRS have already encoded the word order information of the whole input sequence which makes the type of feed-forward layer less important. However, when we replace Conv1d with linear layers for the feed-forward layer in ORT+CRF, the performance greatly drops (∼8.5% F1-score drops for Spanish and ∼5% F1-score drops for Thai), and the performance continues to significantly drop when the CRF layer is removed (ORT+Linear). This is because ORT can not encode any order information when the feed-forward layer Conv1d is replaced with the linear layer, and not any word order information is injected into the ORT+Linear model in which the CRF layer is removed. This makes the model perform badly in the source language and then weakens its adaptation ability to target languages. In addition, we observe that the Conv1d feed-forward layer is also important for TRS trained with order-shuffled data. This is because Conv1d encodes the order of tokens in the entity (we do not shuffle the tokens in an entity), which is essential for detecting entities.

Kernel Size vs. Performance Since the kernel size of Conv1d represents the amounts of local word order information that ORT encodes, we explore how the kernel size affects the performance. As shown in Figure 3, with the increase of kernel size, the zero-shot performance of ORT decreases, and the performance of ORT becomes similar to TRS's when the kernel size is 10. This is because the larger the kernel size, the more order information the model will encode. Hence, the model's generalization ability to target languages decreases when the kernel size is too large.

Different Permutations of Word Order We use different permutations (different k) of the word order to generate order-shuffled data. We find that when we slightly shuffle the word order ($k = 2$), the performance becomes worse than not using order-shuffled data. This is because the model fits the slightly shuffled word order, which is not similar to the target languages. After more perturbations are added to word order, TRS becomes more robust to order differences.

Effectiveness of the CRF Layer For sequence labeling tasks, the CRF layer, which models the conditional probability of label sequences, could also implicitly model the source language word order in training. Therefore, we conduct an ablation study to test the effectiveness of the CRF layer for the cross-lingual models. From Table 6, we can see that removing the CRF layer makes the performance worse. We conjecture that although the CRF layer might contain some information on the word order pattern in the source language, It also models the conditional probability for tokens that belong to the same entity so that it learns when the start or the end of an entity is. This is important for sequence labeling tasks, and models that have the CRF layer removed might not have this ability. For example, in the SF task, when the user says "set an alarm for 9 pm", "for 9 pm" belongs to the "DateTime" entity, and the CRF layer learns to model "for" and "pm" as the start and end of the "DateTime" entity, respectively. Without the CRF layer, models treat the features of these tokens independently.

	PE	Feed-forward	k	es	th
TRS + CRF	Trainable	Linear	-	62.13	22.68
TRS + CRF	Sinusoid	Linear	-	62.55	21.82
w/ shuffled data	Sinusoid	Linear	∞	58.89	19.27
TRS + Linear	Sinusoid	Conv1d	-	55.40	19.33
TRS + CRF	Sinusoid	Conv1d	-	62.67	22.33
w/ shuffled data	Sinusoid	Conv1d	2	61.12	21.24
w/ shuffled data	Sinusoid	Conv1d	3	63.20	23.34
w/ shuffled data	Sinusoid	Conv1d	4	63.54	23.59
w/ shuffled data	Sinusoid	Conv1d	∞	**63.86**	**24.17**
ORT + Linear	-	Linear	-	39.65	13.52
ORT + CRF	-	Linear	-	58.27	20.35
ORT + Linear	-	Conv1d	-	61.76	22.44
ORT + CRF	-	Conv1d	-	**66.84**	**25.53**

Table 6: Ablation study on positional embeddings, feed-forward layer, word order shuffling, and CRF layer. Results are the F1-scores for the zero-shot SF task. "-" denotes that the model does not have this module. "+CRF" and "+Linear" denotes using and not using the CRF layer, respectively.

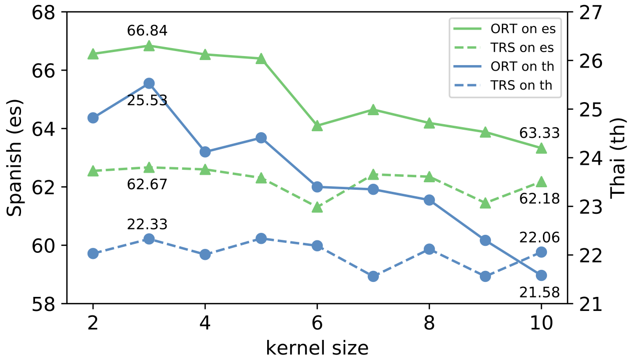


Figure 3: Zero-shot results on the SF task with different kernel sizes for ORT and TRS.

Conclusion

In this paper, we investigate whether reducing the word order of the source language fitted into the models can improve cross-lingual sequence labeling performance. We propose several methods to build order-reduced models, and then compare them with order-sensitive baselines. Extensive experimental results show that order-reduced Transformer (ORT) is robust to the word order shuffled sequences, and it consistently outperforms the order-sensitive models as well as relative positional Transformer (RPT). Taking this further, ORT can also be applied to strong cross-lingual models and improve their performance. Additionally, preserving the order-agnostic property for the M-BERT positional embeddings gives the model a better generalization ability to target languages. Furthermore, we show that encoding excessive or insufficient word order information leads to inferior cross-lingual performance, and models that do not encode any word order information perform badly in both source and target languages.

References

Ahmad, W. U.; Zhang, Z.; Ma, X.; Hovy, E.; Chang, K.-W.; and Peng, N. 2018. On difficulties of cross-lingual transfer with order differences: A case study on dependency parsing. *arXiv preprint arXiv:1811.00570* .

Artetxe, M.; and Schwenk, H. 2019. Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond. *Transactions of the Association for Computational Linguistics* 7: 597–610.

Chen, X.; Hassan, A.; Hassan, H.; Wang, W.; and Cardie, C. 2019. Multi-Source Cross-Lingual Model Transfer: Learning What to Share. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 3098–3112.

Conneau, A.; Khandelwal, K.; Goyal, N.; Chaudhary, V.; Wenzek, G.; Guzmán, F.; Grave, E.; Ott, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116* .

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186.

Hochreiter, S.; and Schmidhuber, J. 1997. Long Short-Term Memory. *Neural Computation* 9(8): 1735–1780.

Huang, H.; Liang, Y.; Duan, N.; Gong, M.; Shou, L.; Jiang, D.; and Zhou, M. 2019. Unicoder: A Universal Language Encoder by Pre-training with Multiple Cross-lingual Tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2485–2494.

Huck, M.; Dutka, D.; and Fraser, A. 2019. Cross-lingual Annotation Projection Is Effective for Neural Part-of-Speech Tagging. In *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects*, 223–233.

Joulin, A.; Bojanowski, P.; Mikolov, T.; Jégou, H.; and Grave, E. 2018. Loss in Translation: Learning Bilingual Word Mapping with a Retrieval Criterion. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

Kim, J.-K.; Kim, Y.-B.; Sarikaya, R.; and Fosler-Lussier, E. 2017. Cross-lingual transfer learning for pos tagging without cross-lingual resources. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, 2832–2838.

Kim, Y. 2014. Convolutional Neural Networks for Sentence Classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1746–1751.

Kim, Y.-B.; Snyder, B.; and Sarikaya, R. 2015. Part-of-speech Taggers for Low-resource Languages using CCA Features. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 1292–1302.

Lample, G.; Ballesteros, M.; Subramanian, S.; Kawakami, K.; and Dyer, C. 2016. Neural Architectures for Named Entity Recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 260–270.

Lample, G.; and Conneau, A. 2019. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291* .

Lample, G.; Conneau, A.; Denoyer, L.; and Ranzato, M. 2018. Unsupervised machine translation using monolingual corpora only. In *International Conference on Learning Representations (ICLR)*.

Liang, Y.; Duan, N.; Gong, Y.; Wu, N.; Guo, F.; Qi, W.; Gong, M.; Shou, L.; Jiang, D.; Cao, G.; et al. 2020. XGLUE: A New Benchmark Dataset for Cross-lingual Pre-training, Understanding and Generation. *arXiv preprint arXiv:2004.01401* .

Liu, Z.; Shin, J.; Xu, Y.; Winata, G. I.; Xu, P.; Madotto, A.; and Fung, P. 2019. Zero-shot Cross-lingual Dialogue Systems with Transferable Latent Variables. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 1297–1303.

Liu, Z.; Winata, G. I.; Lin, Z.; Xu, P.; and Fung, P. 2020a. Attention-informed mixed-language training for zero-shot cross-lingual task-oriented dialogue systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 8433–8440.

Liu, Z.; Winata, G. I.; Xu, P.; Lin, Z.; and Fung, P. 2020b. Cross-lingual Spoken Language Understanding with Regularized Representation Alignment. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 7241–7251.

Ma, X.; and Hovy, E. 2016. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1064–1074.

Mayhew, S.; Tsai, C.-T.; and Roth, D. 2017. Cheap translation for cross-lingual named entity recognition. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, 2536–2545.

Ni, J.; Dinu, G.; and Florian, R. 2017. Weakly Supervised Cross-Lingual Named Entity Recognition via Effective Annotation and Representation Projection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1470–1480.

Nivre, J.; Agić, Ž.; Ahrenberg, L.; et al. 2017. Universal Dependencies 2.0. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University, Prague.

Pires, T.; Schlinger, E.; and Garrette, D. 2019. How Multilingual is Multilingual BERT? In *Proceedings of the 57th*

Annual Meeting of the Association for Computational Linguistics, 4996–5001.

Sang, E. T. K.; and De Meulder, F. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, 142–147.

Schuster, S.; Gupta, S.; Shah, R.; and Lewis, M. 2019a. Cross-lingual Transfer Learning for Multilingual Task Oriented Dialog. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 3795–3805.

Schuster, T.; Ram, O.; Barzilay, R.; and Globerson, A. 2019b. Cross-Lingual Alignment of Contextual Word Embeddings, with Applications to Zero-shot Dependency Parsing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 1599–1613.

Shaw, P.; Uszkoreit, J.; and Vaswani, A. 2018. Self-Attention with Relative Position Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 464–468.

Tiedemann, J.; and Agic, Ž. 2016. Synthetic treebanking for cross-lingual dependency parsing. *Journal of Artificial Intelligence Research* 55(1): 209–248.

Tjong Kim Sang, E. F. 2002. Introduction to the CoNLL-2002 shared task: language-independent named entity recognition. In *proceedings of the 6th conference on Natural language learning-Volume 20*, 1–4.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998–6008.

Wisniewski, G.; Pécheux, N.; Gahbiche-Braham, S.; and Yvon, F. 2014. Cross-lingual part-of-speech tagging through ambiguous learning. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1779–1785.

Wu, S.; and Dredze, M. 2019. Beto, Bentz, Becas: The Surprising Cross-Lingual Effectiveness of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 833–844.

Xie, J.; Yang, Z.; Neubig, G.; Smith, N. A.; and Carbonell, J. G. 2018. Neural Cross-Lingual Named Entity Recognition with Minimal Resources. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 369–379.

Zhang, M.; Zhang, Y.; and Fu, G. 2019. Cross-Lingual Dependency Parsing Using Code-Mixed TreeBank. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 996–1005.