Analyzing BERT Cross-lingual Transfer Capabilities in Continual Sequence Labeling

Juan M. Coria[1*] and **Mathilde Veron**[1,2*] and **Sahar Ghannay**[1]
Guillaume Bernard[2] and **Hervé Bredin**[3] and **Olivier Galibert**[2] and **Sophie Rosset**[1]

[1]Université Paris-Saclay CNRS, LISN, Orsay, France; [2]LNE, Trappes, France;
[3]IRIT, Université de Toulouse, CNRS, Toulouse, France
[1]{lastname}@lisn.fr;
[2]{name.lastname}@lne.fr; [3]{name.lastname}@irit.fr

Abstract

Knowledge transfer between neural language models is a widely used technique that has proven to improve performance in a multitude of natural language tasks, in particular with the recent rise of large pre-trained language models like BERT. Similarly, high cross-lingual transfer has been shown to occur in multilingual language models. Hence, it is of great importance to better understand this phenomenon as well as its limits. While most studies about cross-lingual transfer focus on training on independent and identically distributed (*i.e.* *i.i.d.*) samples, in this paper we study cross-lingual transfer in a continual learning setting on two sequence labeling tasks: slot-filling and named entity recognition. We investigate this by training multilingual BERT on sequences of 9 languages, one language at a time, on the MultiATIS++ and MultiCoNER corpora. Our first findings are that forward transfer between languages is retained although forgetting is present. Additional experiments show that lost performance can be recovered with as little as a single training epoch even if forgetting was high, which can be explained by a progressive shift of model parameters towards a better multilingual initialization. We also find that commonly used metrics might be insufficient to assess continual learning performance.

1 Introduction

State-of-the-art models for Natural Language Processing (NLP) usually leverage deep neural networks. In particular, pre-trained Transformer-based (Vaswani et al., 2017) language models like BERT (Devlin et al., 2019) have proven to perform very well on various NLP tasks, often achieving state-of-the-art results (Raffel et al., 2020; Brown et al., 2020). These models are pre-trained in a self-supervised way on large text corpora and rely on knowledge transfer to solve downstream tasks, where the pre-trained model is fine-tuned on the target task. Multilingual versions of these models have also been trained and demonstrate high cross-lingual transfer as well (K et al., 2020; Wang et al., 2020; Conneau et al., 2020; Xue et al., 2020). Given the interest in these models for cross-lingual transfer, it is of great importance to better understand this phenomenon as well as its limits.

In this work, we analyse the cross-lingual transfer capabilities of multilingual BERT and we work on sequence labeling, where each token of a sentence must be annotated with a specific label. This problem regroups various NLP tasks like Named Entity Recognition (NER), Part-Of-Speech (POS) Tagging, text chunking and slot-filling. We focus our study on two of these tasks using two multilingual corpora[1]: MultiATIS++ for slot-filling (Xu et al., 2020) and MultiCoNER for NER (Malmasi et al., 2022a,b). Experimenting on different corpora allows us to identify which observations may generalize and which ones may be corpus specific.

While most cross-lingual transfer studies about slot-filling or NER focus either on joint training or training on a source and a target language (Xu et al., 2020; Schuster et al., 2019; Arkhipov et al., 2019; Mueller et al., 2020; Wang et al., 2020), our main contribution is a study with special focus on *continual* cross-lingual transfer, where the model performs one single task but is progressively adapted over a sequence of languages.

We believe this experimental setup to be interesting not only as a novel way of studying cross-lingual transfer but also because it is better suited to real case scenarios. Indeed, adaptation to new data over time is a highly desirable feature of most NLP models: oftentimes, collecting data and an-

[*] These authors have contributed equally. The order is alphabetical.

[1] We do not work on the recent MASSIVE (FitzGerald et al., 2022) corpus as we consider it too similar to MultiATIS++. We also avoid Universal Depedencies (Nivre et al., 2020) because we consider POS tagging to be too simple for this type of study. Moreover, the amount of per-language data in the latter could bias the transfer we observe.

Proceedings of the 29th International Conference on Computational Linguistics, pages 15–25
October 12–17, 2022.

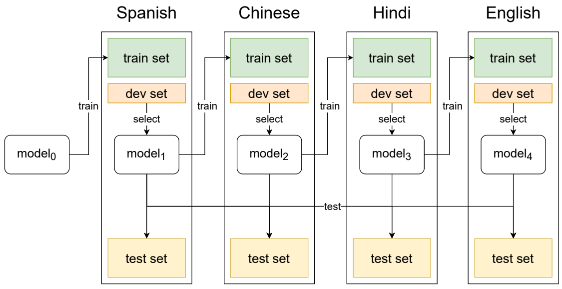


Figure 1: Depiction of a training sequence across 4 languages. For each language in the given order, we train the model on its training set, select the best epoch on the development set and then test on all test sets independently.

Language	Utterances *train*	*dev*	*test*	Labels
MultiATIS++				
Hindi	1,440	160	893	75
Turkish	578	60	715	71
Others	4,488	490	893	84
MultiCoNER				
All	15,3K	800	≥138K	6

Table 1: Number of sentences per subset and number of unique labels (without B and I prefix) for each language in MultiATIS++ (Xu et al., 2020) and MultiCoNER (Malmasi et al., 2022a).

notating them is expensive, which makes training data scarce or incomplete at the beginning of a project. Additionally, model requirements might also evolve with time based on the needs of the users. This means that the model has to adapt sequentially as training data becomes available. An example of this could be a dialogue system that is gradually deployed in different countries. Unfortunately, naive solutions to adapt a previously trained model are costly, as they require either re-training from scratch or maintaining many distinct models.

On the other hand, progressively training on multiple datasets that become available one by one is at the heart of continual learning (Hadsell et al., 2020), where the goal is for a model to improve itself both on past and new data. We refer to these datasets and the order in which they appear as a *training sequence* (f.i. see Figure 1). Traditional training schemes assume that training examples (in our case annotated sentences) are independent and identically distributed (*i.i.d.*), which does not usually hold when data becomes available sequentially. Moreover, access to previous data is not allowed[2], as this represents a linear use of resources with respect to the length of the sequence, which can in theory be infinite. In this context, transfer is generally divided in two: forward and backward (Hadsell et al., 2020; Lopez-Paz and Ranzato, 2017; Arora et al., 2019), defined in our case as improvement on future and already acquired languages respectively. The biggest challenge of continual learning systems is catastrophic forgetting (Hadsell et al., 2020; French, 1999), which is defined as a strong performance loss in previously acquired knowledge (*i.e.* negative backward transfer). While previous studies on continual learning tend to focus on the domain axis for the slot-filling task (Lee, 2017; Madotto et al., 2020), or on the class axis for the NER task (Monaikul et al., 2021; Xia et al., 2022), we concentrate on the axis of language adaptation.

Similar work also investigates cross-lingual transfer of multilingual BERT fine-tuned on sequence labeling tasks, namely NER and POS-Tagging (Liu et al., 2021). They focus on preserving masked language modeling performance and cross-lingual ability after fine-tuning on one of the two tasks on English only, with a method developed as part of continual learning. Conversely, our work focuses on fine-tuning on a single task over a sequence of many languages.

In this paper, we first describe in Section 2 and 3 the task, the corpora and the model we are working with. Then in Section 4 we define the different continual learning metrics that we use in our experiment. Our study is guided by the following research questions, as presented in Section 5: does cross-lingual transfer exist during continual training or does catastrophic forgetting prevent it? How much transfer can we expect relative to monolingual and multilingual *i.i.d.* training? In Section 6 we perform an extensive analysis on MultiATIS++ in order to understand how transfer is affected by the training sequence. Finally, in Section 7 we investigate whether lost performance (due to forgetting) can be recovered and at what cost.

2 Task and corpora

2.1 Sequence labeling

In sequence labeling, each token of a sentence must be annotated with a specific label. Hence, it is appropriate to identify concepts or entities in sen-

[2] Access to previous data is sometimes allowed if limited (Robins, 1995)

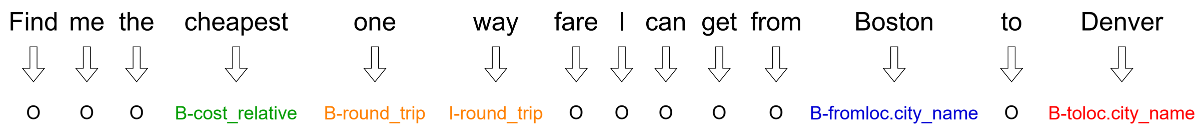


Figure 2: Example of slot filling IOB (Ramshaw and Marcus, 1995) labels for an utterance of MultiATIS++ (Xu et al., 2020) in English. Label "O" (from *outside*) denotes that no concept is mentioned, "B" (from *beginning*) denotes the first word of a concept and "I" (from *inside*) the continuation of a concept. Different slot types are shown in different colors.

tences. In our case, the labels to predict are the same across languages so that the task remains unchanged over the continual learning process.

Sequences are labeled using the IOB format (Ramshaw and Marcus, 1995), where labels consist of a prefix (B,I or O) and an optional type that categorizes the identified concept. While O indicates that the token is not part of a concept (O for outside), B and I indicate that it is the beginning or continuation of a concept, thus allowing the identification of multi-token concepts. An example of this labeling scheme is shown in Figure 2.

This task is usually evaluated using the slot micro F1 score (Tjong Kim Sang and Buchholz, 2000).

2.2 MultiATIS++

The MultiATIS++ multilingual corpus comes from the Air Travel Information System (ATIS) corpus (Hemphill et al., 1990), consisting in utterances of users asking for flight information. The corpus focuses on the slot-filling task, which is related to task-oriented dialogue systems. It enables the system to identify the important concepts mentioned by the user that are needed to successfully continue the dialogue. These concepts are related to the system's domain and to the tasks that the system should perform. This corpus is the manual translation of the original English (EN) ATIS sentences into 6 different languages: Spanish (ES), Portuguese (PT), German (DE), French (FR), Chinese (ZH) and Japanese (JA). It also includes two additional languages: Hindi (HI) and Turkish (TR), that were added as part of MultiATIS in (Upadhyay et al., 2018).

Contrary to the translations added in MultiATIS++, the number of utterances of Hindi and Turkish translations are not as many as for the other languages. More details on the composition of MultiATIS++ are shown in Table 1.

2.3 MultiCoNER

The MultiCoNER corpus was proposed as part of the SemEval 2022 Task 11 (Malmasi et al., 2022a,b) and focuses on the NER task. While it is usually a generic task consisting in identifying entities like people, organizations, locations or dates in written texts, this corpus focuses on detecting ambiguous and complex entities in short and low-context settings. These entities are person, location, group, corporation, product and creative work. MultiCoNER also aims at stimulating the research on multilingual models, as it contains annotations in 11 languages. For a fair comparison with MultiATIS++, we restrict these experiments to also contain 9 languages, namely Bengali (BN), German (DE), English (EN), Spanish (ES), Hindi (HI), Korean (KO), Dutch (NL), Turkish (TR) and Chinese (ZH). More details on the composition of MultiCoNER are shown in Table 1. In the rest of the paper and for both corpora we denote the *train*, *dev* and *test* sets of a given language i with a subscript (e.g. $train_i$).

3 Model

We use the multilingual BERT (Devlin et al., 2019) base model, consisting of 12 multi-head attention layers with 12 heads and hidden size of 768 (177M parameters). This model was trained on large Wikipedia dumps from 104 different languages using masked language modelling and next sentence prediction objectives.

As we use the model for sequence labeling, we append a two-layer feed-forward classifier with hidden size 768 and ReLU (rectified linear unit) activation (Nair and Hinton, 2010). The input of the classifier are the last layer word hidden states after applying dropout with $p = 0.1$.

Following (Xu et al., 2020), we train the model on MultiATIS++ using the Adam optimizer (Kingma and Ba, 2015) with a learning rate of 10^{-5} and a batch size of 32 utterances for

50 epochs (unless stated otherwise), selecting the model with the highest slot F1 on the corresponding *dev* set. We train the model on MultiCoNER the same way, except for the learning rate (optimized on *dev* and set to 5×10^{-5}) and the number of epochs, which is set to 15. We evaluate the model on all $test_i$ sets for every language i using the slot F1 calculated with the `seqeval` library (Nakayama, 2018).

4 Continual Learning Metrics

Cross-lingual transfer can be defined as the performance improvement of a model on a particular language based on knowledge of other languages. This can take several forms depending on the training structure. In an *i.i.d.* context, where all data are available from the start, we think of transfer in terms of joint training. If training on language i and j jointly (multilingual) yields better performance on j than training only on j (monolingual), then there is transfer from i to j.

However, continual learning adds a different dimension. Indeed, when training on a language sequence we can identify two types of transfer: forwards and backwards (Hadsell et al., 2020; Lopez-Paz and Ranzato, 2017). Forward transfer denotes the performance and learning efficiency improvement on a given language thanks to previously acquired knowledge of other languages. Conversely, backward transfer denotes the performance improvement on a previously acquired language when learning a new one. More formally, and similarly to Lopez-Paz and Ranzato (2017), given a sequence of L languages, we define the performance matrix $P \in \mathbb{R}^{L \times L}$, where P_{ij} is the performance of language i after learning language j. In this context, backward transfer of i is defined as:

$$\mathrm{BT}_i = P_{iL} - P_{ii} \tag{1}$$

Negative backward transfer is also called forgetting, as it denotes performance loss on previous languages. Since P_{11} is equivalent to monolingual performance mono_1, we can define backward transfer of the first language after learning language j:

$$\mathrm{BT}_{1j} = P_{1j} - \mathrm{mono}_1 \tag{2}$$

Conversely, we define forward transfer as:

$$\mathrm{FT}_i^{\mathrm{mono}} = P_{ii} - \mathrm{mono}_i \tag{3}$$

where mono_i denotes monolingual performance on language i. By comparing performance with a different baseline like multilingual, we can measure how close forward transfer is to joint transfer:

$$\mathrm{FT}_i^{\mathrm{multi}} = P_{ii} - \mathrm{multi}_i \tag{4}$$

where multi_i denotes the multilingual performance on language i. These definitions will be useful for the analysis in Section 6.

5 Cross-lingual Transfer

Does transfer exist during continual training or does catastrophic forgetting prevent it?

Before studying the continual learning scenario, we first measure transfer when training the model on all languages at once (*i.e. joint* transfer). Then, having this frame of reference, we investigate transfer when training the model on each language sequentially (*i.e. continual* transfer).

5.1 Joint Transfer

In order to measure transfer in unstructured *i.i.d.* training, we train the model on all languages together (multilingual) and compare the performance we obtain with monolingual training. Note that multilingual training corresponds to concatenating all $train_i$ for training and all dev_i for validation. We report the mean and standard deviation of *test* slot F1 per language across 5 runs to reduce the effect of randomness.

Results on MultiATIS++ are reported in Table 2. We observe that multilingual is always stronger than monolingual (except for Chinese and Japanese), which confirms the existence of joint cross-lingual transfer. European languages (German, English, Spanish, French and Portuguese) show modest but visible gains from transfer, whereas Asian languages (Chinese and Japanese) do not seem to benefit from it. However, transfer for the two low resource languages (Hindi and Turkish) is outstanding, with an absolute 4.8% and 13.9% improvement. As noted in (Do et al., 2020), MultiATIS++ translations keep the same (unrealistic) slot values for particular labels (e.g. American *departure city* and *destination city* in Turkish utterances). We suspect this may be the reason why transfer is particularly high in this corpus. The fact that the corpus contains less training data for Hindi and Turkish than for the other languages might also

Training	DE	EN	ES	FR	PT	ZH	JA	HI	TR	Model Cost		Data Cost
										Time	Space	Space
Monolingual	94.4 (0.2)	95.6 (0.1)	88.9 (0.4)	93.2 (0.1)	90.3 (0.6)	93.3 (0.4)	93.1 (0.4)	82.4 (0.5)	71.3 (0.9)	≤224K	1.6B	≤4K
Multilingual	95.0 (0.2)	96.0 (0.2)	90.4 (0.4)	94.0 (0.3)	91.4 (0.2)	93.6 (0.2)	93.0 (0.1)	87.2 (0.3)	85.2 (0.6)	1.7M	178M	33K
Joint transfer	+0.6	+0.4	+1.5	+0.8	+1.1	+0.3	-0.1	+4.8	+13.9	-	-	-
Continual (P_{LL})	94.9 (0.2)	95.9 (0.1)	89.9 (0.5)	93.9 (0.3)	91.3 (0.3)	93.9 (0.3)	93.1 (0.3)	85.6 (0.7)	84.0 (0.6)	≤224K	178M	≤4K
FT_{1L}^{mono}	+0.5	+0.3	+1.0	+0.7	+1.0	+0.6	+0.0	+3.2	+12.7	-	-	-
Continual (P_{1L})	94.0 (0.7)	95.5 (0.2)	89.2 (0.5)	91.4 (1.7)	88.4 (4.9)	92.0 (1.0)	91.7 (0.7)	80.5 (1.8)	68.1 (3.5)	≤224K	178M	≤4K
BT_{1L}	-0.4	-0.1	+0.3	-1.8	-1.9	-1.3	-1.4	-1.9	-3.2	-	-	-

Table 2: Slot F1 performance on MultiATIS++ on *test*$_i$ sets for monolingual, multilingual and continual experiments. The latter are calculated as the average of the first (P_{1L}) or last (P_{LL}) language (indicated by the column) at the end of the sequence. See Equations 2 and 3 for the definition of BT_{1L} and FT_{1L}^{mono}. Reported values are the average of 5 runs with standard deviation shown in parenthesis. Model time cost denotes the cost of adding a new language to the model measured in iterations. Model space cost is the size of the model measured in number of parameters. Data space cost represents the maximum number of training sentences stored in memory at the same time.

Training	BN	DE	EN	ES	HI	KO	NL	TR	ZH	Model Cost		Data Cost
										Time	Space	Space
Monolingual	41.6 (3.2)	64.1 (0.8)	61.3 (0.6)	59.0 (0.8)	43.1 (1.2)	56.7 (0.7)	61.4 (0.9)	45.7 (0.7)	57.6 (0.8)	765K	1.6B	15K
Multilingual	44.9 (1.6)	66.9 (0.4)	64.4 (0.7)	63.8 (0.4)	46.4 (1.2)	59.4 (0.8)	66.5 (0.5)	50.6 (1.0)	58.2 (1.0)	6.9M	178M	138K
Joint transfer	+3.3	+2.8	+3.1	+4.8	+3.3	+2.7	+5.1	+4.9	+0.6	-	-	-
Continual (P_{LL})	43.4 (1.8)	66.0 (0.6)	63.0 (0.6)	62.1 (0.9)	44.2 (1.0)	57.0 (0.7)	64.6 (0.6)	50.1 (0.8)	56.2 (1.3)	765K	178M	15K
FT_{1L}^{mono}	+1.8	+1.9	+1.7	+3.1	+1.1	+0.3	+3.2	+4.4	-1.4	-	-	-
Continual (P_{1L})	31.7 (4.5)	50.9 (1.5)	52.5 (2.6)	51.1 (2.3)	32.2 (2.4)	43.2 (2.4)	55.4 (3.4)	37.4 (1.9)	40.0 (2.8)	765K	178M	15K
BT_{1L}	-9.9	-13.2	-8.8	-7.9	-10.9	-13.6	-6.0	-8.3	-17.6	-	-	-

Table 3: Slot F1 performance on MultiCoNER on *test*$_i$ sets for monolingual, multilingual and continual experiments. Same comments from Table 2 apply.

explain why joint transfer is much higher for these two languages.

Table 3 shows results on MultiCoNER. Monolingual results are much lower than in MultiATIS++ even if the number of labels to predict is much lower, suggesting that MultiCoNER is more difficult than MultiATIS++. Although the corpus is not parallel, we observe significant joint cross-lingual transfer (except for Chinese where it is negligible). This is somehow surprising considering that only a maximum of 8% of entity mentions appearing in the test set of a given language are common to those appearing in the train set of other languages.

However, multilingual training assumes that all languages are available at once. As mentioned before, this is not always true in practice, since utterances may be scarce and annotations expensive. Moreover, given N the maximum number of utterances per language and L the number of languages, training on a new language has time cost $O(LN)$, as the whole model needs to be trained from scratch. A naive solution is to use multiple monolingual models, raising however the space cost to $O(LN)$. Reducing both costs to $O(N)$ motivates our decision to structure training as a sequence.

5.2 Continual Transfer

Given a training sequence (a list of languages in a given order), continual learning consists in training the model on *train*$_i$ (and validating on *dev*$_i$) for each language i in the given order, as depicted in Figure 1. Although having all languages at once is not required and the language addition cost is the lowest, this approach is prone to forgetting previously learned languages.

In the experiments of this section, we report for both forward and backward transfer the average performance per language. The experiments consist of 3 sequences per language and per transfer type repeated 5 times to reduce the effect of randomness, making a total of 54 sequences and 270 experiments. These 3 sequences per language are chosen randomly and maximizing the Kendall rank correlation coefficient (Abdi, 2007) as a distance criterion so that they are as dissimilar as possible.

We first investigate whether forward transfer exists in continual training by looking at the average P_{LL} performance (*e.g.* model$_4$ evaluated on English in Figure 1) against monolingual and multilingual. Notice that we look at the performance of the last language, as this allows us to measure whether the model leverages past knowledge to learn a new language. This has the advantage of isolating the effect of forward transfer from that of backward transfer. When generating the sequences we also make sure that each language appears at the *end* of the sequence the same number of times.

Similarly, we look at backward transfer by comparing the average P_{1L} performance (*e.g.* model$_4$ evaluated on Spanish in Figure 1) against monolingual, making sure that each language appears at the *beginning* of the sequence the same number of times. This way we can determine whether the initial performance (equal to monolingual) improves with the introduction of new languages to the model. We also look at the performance of the first language, so that the effect of backward transfer is isolated from that of forward transfer.

Notice that whether we focus on the first or the last language, we always look at the performance at the end of the training sequence so that the comparison to multilingual is fair.

Results on MultiATIS++ are reported in Table 2. We observe that continual training benefits from cross-lingual forward transfer. Indeed, P_{LL} is on average closer to multilingual than to monolingual performance. However, although transfer is present for the last language, P_{1L} suffers from the opposite effect, even falling under monolingual performance. Our results show that contrary to what we expected from the identical slot values of MultiATIS++ (e.g. American *departure city* and *destination city* in Turkish utterances), the naturally occurring cross-lingual transfer completely vanishes in previous languages.

Similar observations can be made from MultiCoNER continual experiments from Table 3. Although forward transfer is high in general, it is also lower than the standard deviation for Bengali, Hindi and Korean, and even negative for Chinese. The negative backward transfer values also show that the model forgets a lot about the first language it learnt.

Overall we can see that continual training benefits from forward transfer, although still not performing as well as the multilingual topline, whereas forgetting is clearly present.

6 Training Sequence

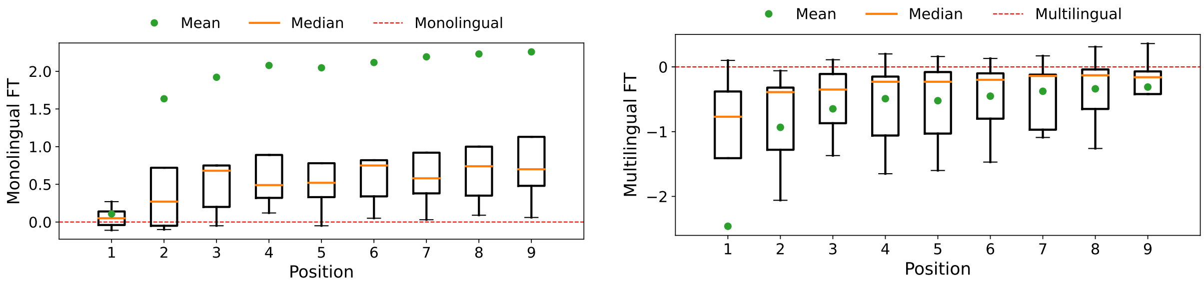
How is transfer affected by the training sequence?

In order to better understand the effect of the training sequence on transfer, we first look at measures of forward transfer at each position relative to monolingual and multilingual. Secondly, we study the impact of the training sequence length on backward transfer measured on the first language. This analysis is conducted only on MultiATIS++ due to time and computational constraints. In the figures of this section, the mean, median and percentiles do take into account eventual outlier languages, while the minimum and maximum do not.

When considering forward transfer, Figure 3a shows that apart from the first position (equal to monolingual), the model consistently benefits from transfer at any point in the sequence, as performance is higher than monolingual. Interestingly, due to some outlier languages (generally Hindi and Turkish), we observe that the means are poor estimates of the distribution when measuring $\mathrm{FT}_i^{\mathrm{mono}}$. This is an indicator that commonly used continual transfer metrics might over- or underestimate real performance when transfer is not uniformly distributed among languages. Indeed, these metrics usually consist of averages across the adaptation axis (Lopez-Paz and Ranzato, 2017). In Figure 3b, we also observe that performance gets closer to multilingual as the sequence advances, although it rarely outperforms it.

As per backward transfer, Figure 4 shows that performance of the first language is in general worse than monolingual for any given sequence length. In particular, we observe that performance loss is not strictly monotonic, which means that measuring forgetting between the beginning and the end of the sequence may not be sufficient to explain how the model forgets. Note that a sequence of $L = 7$ would have shown less forgetting than a sequence of $L = 5$.

Furthermore, as hinted by continual experiments from Table 2, we observe that backward transfer deteriorates as forward transfer improves with the length of the sequence. Since negative backward transfer (*i.e.* forgetting) tends to be linked to a loss of previously acquired knowledge, it is surprising that new language performance keeps increasing while performance of known languages decreases. Our results indicate that the preserved knowledge that facilitates the acquisition of a new language in multilingual BERT for slot filling is not the same knowledge that preserves previous language performance. This might be explained by a progressive shift of model parameters towards a better multilingual initialization for the ATIS task that might however fail to retain the specificities of previous



(a) $\mathrm{FT}_i^{\mathrm{mono}} = P_{ii} - \mathrm{mono}_i$ (higher is better)

(b) $\mathrm{FT}_i^{\mathrm{multi}} = P_{ii} - \mathrm{multi}_i$ (higher is better)

Figure 3: Distributions of forward transfer on $test_i$ relative to monolingual and multilingual for different positions i in the sequence. We average over 54 sequences and 5 runs. Note that forward transfer is 0 when performance is equal to (a) monolingual and (b) multilingual. Outliers not shown for readability.

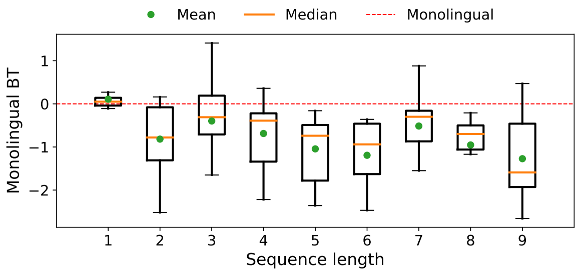


Figure 4: Distributions of first language backward transfer $\mathrm{BT}_{1j} = P_{1j} - \mathrm{mono}_1$ (higher is better) on $test_1$ for different sequence lengths j. We average across 54 sequences and 5 runs. Note that $\mathrm{BT}_{1j} = 0$ if performance is equal to monolingual. Outliers not shown for readability.

languages. This hypothesis motivates our next research question.

7 Fast Recovery

Can lost performance due to forgetting be recovered?

Given that forward transfer does not seem to be affected by forgetting, we investigate in this section whether performance lost as a result of forgetting can be recovered quickly after continual training. The ability to recover is especially interesting for MultiCoNER where forgetting is pretty high, but we still conduct experiments on both corpora. To investigate if this is possible, we first set out to discover whether the model shifts towards a better multilingual initialization. Hence we compare the multilingual performance of the initial model_0 (consisting of BERT and a random classifier) against model_L, the model at the end of training sequence (*e.g.* model_4 in Figure 1). In particular, we train both models on all languages jointly for different numbers of epochs and evaluate on each language. Notice that model_L comes from our continual P_{1L} experiments (see Table 2). The results are presented in Tables 4 and 5.

The comparison between model_0 multilingual and model_L multilingual for both corpora shows two interesting results. On one hand, we observe that even one epoch of multilingual training for model_L achieves better performance than the monolingual baseline (model_0 monolingual) and is even close to the multilingual topline (model_0 multilingual)[3], both of which are trained on the maximum number of epochs (50 or 15). This means that model_L is capable of achieving good multilingual performance with very little training, hence canceling the effect of forgetting. On the other hand, we see that model_L multilingual performance is greatly superior to model_0 multilingual with a single training epoch. This is not surprising given that the classifier is initialized randomly in model_0, but it shows that the model is capable of retaining knowledge from previous languages, although it is not clear whether that knowledge is preserved in the classifier or in BERT.

We dive deeper into this question by training model_L with a random classifier in the same manner (see model_L + rnd clf multi. in Table 4). We observe that performance is still greatly superior to model_0 multilingual with a single epoch. However, performance is not as high as model_L multilingual (although slightly in MultiCoNER), which keeps its continually trained classifier. This indicates most of the knowledge retained from previous languages is stored in BERT, and that the knowledge stored

[3] Except for Chinese on MultiCoNER, which is not surprising considering that its joint transfer is negligible.

Model	Epochs	DE	EN	ES	FR	PT	ZH	JA	HI	TR
model_0 multi. (*i.i.d.*)	1	82.7 (1.2)	83.6 (0.7)	78.2 (0.3)	80.7 (0.7)	79.4 (0.5)	83.5 (0.7)	82.7 (1.0)	79.6 (0.7)	69.8 (1.5)
	5	94.7 (0.2)	95.3 (0.2)	89.9 (0.2)	93.2 (0.2)	90.7 (0.2)	94.0 (0.2)	93.2 (0.5)	85.9 (0.3)	83.6 (0.7)
	50	95.0 (0.2)	96.0 (0.2)	90.4 (0.4)	94.0 (0.3)	91.4 (0.2)	93.6 (0.2)	93.0 (0.1)	87.2 (0.3)	85.2 (0.6)
model_L multi.	1	94.8 (0.3)	95.9 (0.2)	89.7 (0.6)	93.8 (0.3)	91.2 (0.4)	93.6 (0.5)	93.3 (0.3)	85.7 (0.9)	82.8 (1.3)
	5	94.9 (0.2)	95.9 (0.2)	90.0 (0.5)	93.9 (0.3)	91.3 (0.4)	93.7 (0.4)	93.3 (0.3)	86.0 (0.8)	83.4 (1.0)
model_L + rnd clf multi.	1	93.1 (0.5)	93.7 (0.5)	87.9 (0.5)	91.1 (0.5)	88.5 (0.6)	92.6 (0.5)	92.3 (0.6)	83.4 (0.8)	80.8 (1.3)
	5	94.8 (0.2)	95.8 (0.2)	89.9 (0.5)	93.6 (0.3)	91.1 (0.4)	93.7 (0.4)	93.3 (0.3)	86.3 (0.6)	84.1 (0.8)
model_0 mono. (*i.i.d.*)	50	94.4 (0.2)	95.6 (0.1)	88.9 (0.4)	93.2 (0.1)	90.3 (0.6)	93.3 (0.4)	93.1 (0.4)	82.4 (0.5)	71.3 (0.9)
model_L mono.	1	95.1 (0.2)	95.8 (0.2)	90.2 (0.4)	93.6 (0.4)	91.2 (0.4)	93.5 (0.5)	93.4 (0.2)	86.3 (0.6)	79.1 (1.5)
	5	95.0 (0.2)	95.8 (0.2)	90.0 (0.4)	94.0 (0.2)	91.3 (0.2)	93.8 (0.4)	93.4 (0.2)	86.7 (0.4)	81.6 (0.8)
	10	95.1 (0.2)	95.8 (0.2)	90.0 (0.5)	93.9 (0.3)	91.3 (0.4)	93.8 (0.4)	93.4 (0.2)	86.7 (0.4)	82.2 (0.9)

Table 4: Slot F1 performance on $test_i$ sets for MultiATIS++ fast recovery experiments. model_L monolingual performance is averaged over 3 sequences (the P_{1L} experiment ones starting with the language in question), while model_L multilingual is averaged over all 27 sequences from P_{1L} experiments. Both model_0 and model_L experiments are averaged over 5 runs (standard deviation in parenthesis).

Model	Epochs	BN	DE	EN	ES	HI	KO	NL	TR	ZH
model_0 multi. (*i.i.d.*)	1	36.2 (1.4)	63.1 (0.8)	61.6 (0.6)	60.5 (0.6)	40.5 (1.4)	56.9 (0.4)	63.5 (0.7)	45.5 (0.6)	53.1 (2.4)
	5	43.0 (1.1)	66.6 (1.0)	63.9 (0.2)	63.7 (0.6)	45.4 (1.5)	58.9 (0.7)	66.3 (0.7)	49.7 (1.4)	57.7 (1.5)
	15	44.9 (1.6)	66.9 (0.4)	64.4 (0.7)	63.8 (0.4)	46.4 (1.2)	59.4 (0.8)	66.5 (0.5)	50.6 (1.0)	58.2 (1.0)
model_L multi. (*i.i.d.*)	1	42.7 (1.7)	65.8 (0.7)	63.6 (0.7)	63.0 (0.8)	44.8 (1.4)	58.8 (1.0)	65.9 (0.8)	49.8 (1.0)	56.7 (1.3)
	5	43.8 (1.4)	66.4 (0.6)	64.1 (0.5)	63.5 (0.6)	45.4 (1.1)	59.2 (0.8)	66.4 (0.5)	50.6 (0.9)	57.6 (1.2)
model_L + rnd clf multi.	1	42.6 (1.8)	65.5 (0.7)	63.3 (0.6)	62.7 (0.8)	44.7 (1.3)	58.7 (0.8)	65.7 (0.7)	49.6 (1.2)	56.6 (1.4)
	5	43.7 (1.4)	66.3 (0.6)	63.9 (0.6)	63.4 (0.7)	45.2 (1.1)	59.1 (0.8)	66.2 (0.6)	50.4 (1.0)	57.6 (1.1)
model_0 mono. (*i.i.d.*)	15	41.6 (3.2)	64.1 (0.8)	61.3 (0.6)	59.0 (0.8)	43.1 (1.2)	56.7 (0.7)	61.4 (0.9)	45.7 (0.7)	57.6 (0.8)
model_L mono.	1	41.8 (2.4)	65.5 (0.7)	63.7 (0.8)	61.6 (0.5)	44.2 (1.1)	57.6 (0.4)	64.6 (0.7)	49.5 (1.0)	56.0 (0.9)
	5	43.6 (1.8)	66.5 (0.5)	64.0 (0.6)	62.4 (0.6)	45.4 (0.7)	57.9 (0.5)	65.0 (0.8)	50.7 (0.7)	58.3 (0.9)

Table 5: Slot F1 performance on $test_i$ sets for MultiCoNER fast recovery experiments. Same comments from Table 4 apply.

in the classifier is dependent on the corpus.

Overall, these results lead us to think that for the sequence labeling task, continual training over the language sequence does indeed shift model parameters to a better multilingual initialization. As a result, we explore the possibility to leverage this phenomenon in order to quickly recover lost language specificities due to forgetting for both corpora. To do this, we train model_L on the first language of the sequence a second time (*i.e.* as if it were an $(L+1)^{\text{th}}$ language) and evaluate on the first language only. As shown in Tables 4 and 5, when comparing model_L monolingual to model_0 monolingual (equal to first language performance P_{11}), we see that the performance of the first language can be recovered and improved upon with as little as a single training epoch[3]. These results are outstanding for MultiCoNER considering the high forgetting that we previously observed. On MultiATIS++, model_L monolingual even achieves 50-epoch model_0 multilingual performance in most cases after only one epoch, with the remaining languages still showing a big improvement. In particular, Hindi and Turkish improve an absolute 3.9% and 7.8% from model_0 monolingual respectively.

Note that for MultiATIS++ increasing the number of recovery epochs for the first language does not bring considerable improvements. The only exception to this observation is Turkish, which might be explained by the small size of its training set. In MultiCoNER however, performance still improves after 5 epochs, getting closer to the multilingual topline. Surprisingly, model_L monolingual is even on par with the multilingual topline for Turkish and Chinese. Although the cost of adding a language remains $O(N)$, the ability to recover all languages raises costs to $O(LN)$, making it expensive to use in practice. The design of a strategy taking full advantage of these recovery capabilities to limit forgetting with lower cost is left for future work.

8 Discussion

To summarize, we observe a high level of cross-lingual transfer in the *i.i.d.* setting when learning the sequence labeling task on all languages jointly for both corpora. In a real low resource scenario where data and annotations are scarce, it may be difficult or even impossible to implement either a monolingual or multilingual adaptive approach, as time/space complexity is high and not all languages might be available at once. In a continual learning setting where languages are learned in sequence, these costs are the lowest and cross-lingual transfer is retained in the form of forward transfer. However, forgetting occurs for the first language of the sequence since performance consistently drops below monolingual.

When looking at continual cross-lingual transfer across the entire sequence, we obtain two surprising results. First, commonly used continual transfer metrics may not be a reliable estimate of the performance distribution across languages when transfer is not evenly distributed. Since even in other adaptation axes a considerable variability across datasets is to be expected, we believe a statistic like the median might be a better choice, as we believe it better represents expected performance at any given point. Second, as the sequence progresses, forward transfer improves, while backward transfer diminishes. This might indicate that model parameters remain a good initialization for future languages but that previous language specificities might be lost.

Motivated by this hypothesis, we compare the model at the beginning and at the end of the training sequence. Our results suggest that knowledge from past languages is mostly stored in BERT (as opposed to the task-specific classifier) and that the model may indeed shift towards a better multilingual initialization, making it suitable to quickly recover the performance lost as a result of forgetting. We then measure the recovery capabilities of the model with respect to the first language of the sequence. We empirically show that lost performance can be recovered with as little as a single training epoch even if forgetting is high (like in MultiCoNER). Performance can even greatly improve and approach the *i.i.d.* multilingual topline after only one training epoch for MultiATIS++ and 5 epochs for MultiCoNER.

In light of the above, we believe that effective continual learning methods for this task would benefit from leveraging recovery capabilities (either for a single language or many languages jointly) to limit the effect of forgetting, while preserving or even boosting forward transfer.

9 Conclusion

In this paper, we presented an analysis of cross-lingual transfer in continual learning for the sequence labeling task using multilingual BERT (Devlin et al., 2019) as well as the MultiATIS++ (Xu et al., 2020) and MultiCoNER (Malmasi et al., 2022a) corpora.

Our main finding suggests that although forgetting is present, cross-lingual transfer is retained in the form of forward transfer, which allows the model to have substantial recovery capabilities. Moreover, we empirically show that: 1) high forward transfer is linked to a progressive shift of model parameters towards a better multilingual initialization, and 2) that most knowledge from past languages is stored in the word representation encoder (BERT) and not in the task-specific classifier. Finally, we also find that current continual learning metrics may need to be adapted if we want to better estimate the distribution of transfer across the adaptation axis.

As future work, we would like to reduce training costs by leveraging fast recovery for continual learning across languages. Another interesting research direction would be a study on the continual acquisition of languages not already present in multilingual BERT.

Reproducible Research

In the spirit of reproducible research, we release our code as open source available at `github.com/juanmc2005/ContinualNLU`.

Acknowledgements

This work has been partially funded by the LIHLITH project (ANR-17-CHR2-0001-03), and supported by ERA-Net CHIST-ERA, and the "Agence Nationale pour la Recherche" (ANR, France). It has also been partially funded by French ANRT under CIFRE PhD contract # 2019/0628. It was also possible thanks to the Saclay-IA computing platform and was performed using HPC resources from GENCI-IDRIS (Grant 2022-AD011012609R1).

References

Hervé Abdi. 2007. The Kendall rank correlation coefficient. *Encyclopedia of Measurement and Statistics. Sage, Thousand Oaks, CA*, pages 508–510.

Mikhail Arkhipov, Maria Trofimova, Yurii Kuratov, and Alexey Sorokin. 2019. Tuning multilingual transformers for language-specific named entity recognition. In *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*, pages 89–93.

Gaurav Arora, Afshin Rahimi, and Timothy Baldwin. 2019. Does an LSTM forget more than a CNN? an empirical study of catastrophic forgetting in NLP. In *Proceedings of the The 17th Annual Workshop of the Australasian Language Technology Association*, pages 77–86, Sydney, Australia. Australasian Language Technology Association.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *ACL*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Quynh Do, Judith Gaspers, Tobias Roeding, and Melanie Bradford. 2020. To what degree can language borders be blurred in BERT-based multilingual spoken language understanding? In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2699–2709, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Jack FitzGerald, Christopher Hench, Charith Peris, Scott Mackie, Kay Rottmann, Ana Sanchez, Aaron Nash, Liam Urbach, Vishesh Kakarala, Richa Singh, Swetha Ranganath, Laurie Crist, Misha Britan, Wouter Leeuwis, Gokhan Tur, and Prem Natarajan. 2022. Massive: A 1m-example multilingual natural language understanding dataset with 51 typologically-diverse languages.

Robert M. French. 1999. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4):128 – 135.

Raia Hadsell, Dushyant Rao, Andrei A. Rusu, and Razvan Pascanu. 2020. Embracing Change: Continual Learning in Deep Neural Networks. *Trends in Cognitive Sciences*, 24:1028–1040.

Charles T. Hemphill, John J. Godfrey, and George R. Doddington. 1990. The ATIS spoken language systems pilot corpus. In *Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, June 24-27,1990*.

Karthikeyan K, Zihan Wang, Stephen Mayhew, and Dan Roth. 2020. Cross-Lingual Ability of Multilingual BERT: An Empirical Study. In *International Conference on Learning Representations*.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International-Conference on Learning Representations (ICLR)*.

Sungjin Lee. 2017. Toward continual learning for conversational agents. *arXiv preprint arXiv:1712.09943*.

Zihan Liu, Genta Indra Winata, Andrea Madotto, and Pascale Fung. 2021. Preserving Cross-Linguality of Pre-trained Models via Continual Learning. In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, pages 64–71, Online. Association for Computational Linguistics.

David Lopez-Paz and Marc' Aurelio Ranzato. 2017. Gradient Episodic Memory for Continual Learning. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Andrea Madotto, Zhaojiang Lin, Zhenpeng Zhou, Seungwhan Moon, Paul Crook, Bing Liu, Zhou Yu, Eunjoon Cho, and Zhiguang Wang. 2020. Continual learning in task-oriented dialogue systems. *arXiv preprint arXiv:2012.15504*.

Shervin Malmasi, Anjie Fang, Besnik Fetahu, Sudipta Kar, and Oleg Rokhlenko. 2022a. MultiCoNER: a Large-scale Multilingual dataset for Complex Named Entity Recognition.

Shervin Malmasi, Anjie Fang, Besnik Fetahu, Sudipta Kar, and Oleg Rokhlenko. 2022b. SemEval-2022 Task 11: Multilingual Complex Named Entity Recognition (MultiCoNER). In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Natawut Monaikul, Giuseppe Castellucci, Simone Filice, and Oleg Rokhlenko. 2021. Continual learning for named entity recognition. In *AAAI*.

David Mueller, Nicholas Andrews, and Mark Dredze. 2020. Sources of transfer in multilingual named entity recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8093–8104, Online. Association for Computational Linguistics.

Vinod Nair and Geoffrey E Hinton. 2010. Rectified Linear Units Improve Restricted Boltzmann Machines. In *ICML*.

Hiroki Nakayama. 2018. seqeval: A python framework for sequence labeling evaluation. Software available from https://github.com/chakki-works/seqeval.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France. European Language Resources Association.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Lance Ramshaw and Mitch Marcus. 1995. Text Chunking using Transformation-Based Learning. In *Third Workshop on Very Large Corpora*.

Anthony Robins. 1995. Catastrophic Forgetting, Rehearsal and Pseudorehearsal. *Connection Science*, 7(2):123–146.

Sebastian Schuster, Sonal Gupta, Rushin Shah, and Mike Lewis. 2019. 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)*, pages 3795–3805, Minneapolis, Minnesota. Association for Computational Linguistics.

Erik F. Tjong Kim Sang and Sabine Buchholz. 2000. Introduction to the CoNLL-2000 Shared Task Chunking. In *Fourth Conference on Computational Natural Language Learning and the Second Learning Language in Logic Workshop*.

Shyam Upadhyay, Manaal Faruqui, Gokhan Tür, Hakkani-Tür Dilek, and Larry Heck. 2018. (almost) zero-shot cross-lingual spoken language understanding. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6034–6038.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Zihan Wang, Karthikeyan K, Stephen Mayhew, and Dan Roth. 2020. Extending multilingual BERT to low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2649–2656, Online. Association for Computational Linguistics.

Yu Xia, Quan Wang, Yajuan Lyu, Yong Zhu, Wenhao Wu, Sujian Li, and Dai Dai. 2022. Learn and review: Enhancing continual named entity recognition via reviewing synthetic samples. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2291–2300, Dublin, Ireland. Association for Computational Linguistics.

Weijia Xu, Batool Haider, and Saab Mansour. 2020. End-to-end slot alignment and recognition for cross-lingual NLU. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5052–5063, Online. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2020. mt5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2010.11934*.