WECHSEL: Effective initialization of subword embeddings for cross-lingual transfer of monolingual language models

Anonymous ACL submission

Abstract

Large pretrained language models (LMs) have become the central building block of many NLP applications. Training these models requires ever more computational resources and most of the existing models are trained on English text only. It is exceedingly expensive to train these models in other languages. To alleviate this problem, we introduce a novel method – called WECHSEL – to efficiently and effectively transfer pretrained LMs to new languages. WECHSEL can be applied to any model which uses subword-based tokenization and learns an embedding for each subword. The tokenizer of the source model (in English) is replaced with a tokenizer in the target language and token embeddings are initialized such that they are semantically similar to the English tokens by utilizing multilingual static word embeddings covering English and the target language. We use WECHSEL to transfer the English RoBERTa and GPT-2 models to four languages (French, German, Chinese and Swahili). We also study the benefits of our method on very low-resource languages. WECHSEL improves over proposed methods for cross-lingual parameter transfer and outperforms models of comparable size trained from scratch with up to 64x less training effort. Our method makes training large language models for new languages more accessible and less damaging to the environment. We make our code and models publicly available.

1 Introduction

Large LMs based on the Transformer architecture (Vaswani et al., 2017) have become increasingly popular since GPT (Radford et al., 2018) and BERT (Devlin et al., 2019) were introduced, prompting the creation of many large LMs pretrained on English text (Yang et al., 2019; Clark et al., 2020; Lewis et al., 2020; Joshi et al., 2020; Ram et al., 2021). There is a tendency towards training larger and larger models (Brown et al., 2020; Fedus et al., 2021) while the main focus is on the English language. Recent work has called attention to the costs associated with training increasingly large LMs, including environmental and financial cost (Bender et al., 2021). If training large LMs for English is already costly, it is prohibitively expensive to train new, similarly powerful models to cover other languages.

One approach to address this issue is creating massively multilingual models (Devlin et al., 2019; Conneau et al., 2020; Xue et al., 2021) trained on a concatenation of texts in many different languages. These models show strong natural language understanding capabilities in a wide variety of languages, but suffer from what Conneau et al. (2020) call the *curse of multilinguality*: beyond a certain number of languages, overall performance decreases on monolingual as well as cross-lingual tasks. Consistent with this finding, Nozza et al. (2020) observe that monolingual LMs often outperform massively multilingual models. It is thus desirable to train monolingual models in more languages. Training monolingual models in non-English languages is commonly done by training a new model with randomly initialized parameters (Antoun et al., 2020; Louis, 2020; Chan et al., 2020; Martin et al., 2020). However, to train a model with capabilities comparable to that of an English model in this way, presumably a similar amount of compute to what was used to train the English model would be required.

To address this issue, we introduce WECHSEL[1], a novel method to transfer monolingual language models to a new language. WECHSEL uses multilingual static word embeddings between the source language and the target language to initialize model parameters. WECHSEL first copies all inner (non-embedding) parameters of the English model, and exchanges the tokenizer with a tokenizer for the tar-

[1] Word Embeddings Can Help initialize Subword Embeddings in a new Language.

get language. Next, in contrast to prior work doing random initialization (de Vries and Nissim, 2021), the token embeddings in the target language are initialized such that they are close to semantically similar English tokens by mapping multilingual static word embeddings to subword embeddings. The latter step is particularly important considering that token embeddings take up roughly 31% of the parameters of RoBERTa (Liu et al., 2019) and roughly 33% of the parameters of GPT2 (Radford et al., 2019). Intuitively, semantically transferring embeddings instead of randomly initializing one third of the model should result in improved performance. Our parameter transfer provides an effective initialization in the target language, requiring significantly fewer training steps to reach high performance than training from scratch. As multilingual static word embeddings are available for many languages (Bojanowski et al., 2017), WECHSEL is widely applicable.

We conduct our experiments on RoBERTa and GPT-2 as representative models of encoder and decoder language models, respectively. We transfer the English RoBERTa model to four languages (French, German, Chinese and Swahili), and the English GPT-2 model to the same four plus another four very low-resource languages (Sundanese, Scottish Gaelic, Uyghur and Malagasy). We evaluate the transferred RoBERTa models on Neural Entity Recognition (NER), and Natural Language Inference (NLI) tasks in the respective languages. The transferred GPT-2 models are evaluated in terms of Language Modelling Perplexity (PPL) on a held-out set. We compare WECHSEL with randomly initialized models (denoted as FullRand), as well as the recently proposed TransInner method which only transfers the inner (non-embedding) parameters (de Vries and Nissim, 2021). All mentioned models are trained under the same conditions (around 4 days on a TPUv3-8). We also compare our model with models of comparable size trained from scratch under significantly larger training regimes, in particular CamemBERT (Martin et al., 2020) (French), $\text{GBERT}_{\text{Base}}$ (Chan et al., 2020) (German), and $\text{BERT}_{\text{Base}}$-Chinese (Devlin et al., 2019).

Results show that models initialized with WECHSEL outperform randomly initialized models and models initialized with TransInner across all languages and all tasks, for both RoBERTa and GPT-2. In addition, strong performance is reached at a fraction of the training steps of other methods. Our contribution is summarized as follows.

- We propose WECHSEL, a novel method for transferring monolingual language models to a new language by utilizing multilingual static word embeddings between the source and the target language.
- We show effective transfer of RoBERTa and GPT-2 using WECHSEL to four and eight languages, respectively, achieved after minimal training effort.
- We release more effective GPT-2 and RoBERTa models than previously published non-English models, achieved under our more efficient training setting. Our code and models are publicly available at `github.com/anonymized`.

In the following, we review related work in Section 2. We introduce the WECHSEL method in Section 3, followed by explaining the experiment setup in Section 4. We show and discuss results in Section 5.

2 Related Work

Large Language Models. Training Language Models is usually done in a self-supervised manner i. e. deriving labels from the training text instead of needing explicit annotations. One optimization objective is Masked Language Modelling (Devlin et al., 2019, MLM), where randomly selected tokens in the input are replaced by a special `[MASK]` token, and the task is to predict the original tokens. Another common objective is Causal Language Modelling (CLM), where the task is to predict the next token. These two objectives highlight a fundamental distinction between language models: models can be trained as encoders (e.g. with MLM) or as decoders (e.g. with CLM).

Instead of words, the vocabulary of recently proposed language models commonly consists of subwords (Clark et al., 2020; Liu et al., 2019; Devlin et al., 2019).

Multilingual representations. There has been a significant amount of work in creating multilingual static word embeddings. A common method is learning embeddings from scratch using data in multiple languages (Luong et al., 2015; Duong et al., 2016). Alternatively, multilinguality can be

achieved by aligning existing monolingual word embeddings using a bilingual dictionary, so that the resulting embeddings share the same semantic space (Xing et al., 2015; Joulin et al., 2018). Recent studies improve on this by reducing (or even completely removing) the need for bilingual data (Artetxe et al., 2017, 2018; Lample et al., 2018).

Beside static word embeddings, multilinguality is also well studied in the area of contextualized representations. One approach to learn multilingual contextualized representations is through training a model on a concatenation of corpora in different languages. Some models created based on this approach are mBERT (Devlin et al., 2019), XLM-R (Conneau et al., 2020) and mT5 (Xue et al., 2021), trained on text in 104, 100, and 101 languages, respectively. As shown by Pires et al. (2019), a multilingual model such as mBERT can enable cross-lingual transfer by using task-specific annotations in one language to fine-tune the model for evaluation in another language. Despite the benefits, recent studies outline a number of limitations of massively multilingual LMs. Wu and Dredze (2020) empirically show that in mBERT "the 30% languages with least pretraining resources perform worse than using no pretrained language model at all". Conneau et al. (2020) report that beyond a certain number of languages in the training data, the overall performance decreases on monolingual as well as cross-lingual tasks. These studies motivate our work on introducing an efficient approach for creating effective monolingual LMs for more languages.

Cross-lingual transfer of monolingual LMs. Studies in this area can be divided into two categories:

- **Bilingualization of a monolingual LM** is concerned with extending a model to a new language while preserving its capabilities in the original language. Artetxe et al. (2020) approach this problem by replacing the tokenizer and relearning the subword embeddings, while freezing other (non-embedding) parameters. Such a model becomes bilingual, since the initial tokenizer and embeddings can be used for tasks in the source language, while the new tokenizer and embeddings can be used for tasks in the target language. Thus, a model can be finetuned on annotated task data in the source language, and then zero-shot transferred to the target language. Tran (2020) follow a similar approach, while instead of randomly initializing embeddings, they utilize static word embeddings to initialize embeddings in the target language close to semantically similar English tokens. They then continue training the model on an English text corpus as well as on the target language in order to preserve model capabilities in English.

- **Creating a new monolingual LM in the target language** is, in contrast, concerned with transferring a model from a source to a target language without the necessity to preserve its capabilities in the source language. Zoph et al. (2016) and Nguyen and Chiang (2017) show that cross-lingually transferring a machine translation model can improve performance, especially for low-resource languages. Zoph et al. (2016) use embeddings of random tokens in the original vocabulary to initialize token embeddings in the new vocabulary, while Nguyen and Chiang (2017) utilize vocabulary overlap between the source and target language. More recently, de Vries and Nissim (2021) follow a similar approach to the one of Artetxe et al. (2020) for transferring a GPT-2 model to a new language. de Vries and Nissim (2021) add an additional step, where they train the entire model for some amount of steps to allow adapting to the target language beyond the lexical level. We refer to the method of de Vries and Nissim (2021) as TransInner and consider it as a baseline in our experiments.

Our WECHSEL method belongs to the second category. WECHSEL can be seen as an extension to the method proposed by Tran (2020) with the goal of creating a new monolingual LM instead of bilingualizing the LM. This allows removing the constraints imposed by the need to preserve the model's capabilities in the source language. In addition, we generalize the semantic subword mapping done by Tran (2020) to consider an arbitrary number of semantically similar subword with an arbitrary temperature. We are the first to show that a cross-lingually transferred model can outperform monolingual models which have been trained extensively from scratch in the target language, while requiring substantially less computational resources.

3 Methodology

To initialize the model in the target language, we copy the inner (non-embedding) parameters from the source model. Our goal, then, is given the tokenizer T^s in the source language with vocabulary $\mathbb{U}^s$, the corresponding token embeddings $\boldsymbol{E}^s$, and a tokenizer T^t in the target language with vocabulary $\mathbb{U}^t$, to find a good initialization of the embeddings $\boldsymbol{E}^t$ by using $\boldsymbol{E}^s$. To this end, we use existing bilingual word embeddings enriched with subword information, containing a set of words and subword n-grams in the source and target language and their aligned vectors. We denote the set of words and n-grams in the source and target language as $\mathbb{V}^s$ and $\mathbb{V}^t$ respectively, and the aligned static embeddings as $\boldsymbol{W}^s$ and $\boldsymbol{W}^t$. In Appendix D we consider an alternative method if no subword information is available in the bilingual word embeddings.

First, independently for both languages, we compute static subword embeddings for tokens in the tokenizer vocabulary in the same semantic space as the static word embeddings (Section 3.1). This results in subword embeddings $\boldsymbol{U}^s$ and $\boldsymbol{U}^t$ for the source and target language, respectively. Next, we use $\boldsymbol{U}^s$ and $\boldsymbol{U}^t$ to compute the semantic similarity of every subword in $\mathbb{U}^s$ to every subword in $\mathbb{U}^t$. Using these semantic similarities, we initialize the embeddings in $\boldsymbol{E}^t$ through a convex combination of embeddings in $\boldsymbol{E}^s$ (Section 3.2). By applying WECHSEL, the vectors of $\boldsymbol{E}^t$ are in the same semantic space as $\boldsymbol{E}^s$, where a subword in the target language is semantically similar to its counterpart(s) in the source language. These steps are summarized in Figure 1 and explained in more detail in the following.

3.1 Subword Embedding Computation

The process of mapping word embeddings to subword embeddings is done individually for the source and the target language. Given a tokenizer T with vocabulary $\mathbb{U}$ and embeddings $\boldsymbol{W}$, the goal is to find subword embeddings $\boldsymbol{U}$ for subwords in $\mathbb{U}$ in the same semantic space as $\boldsymbol{W}$. To this end, we decompose subwords in $\mathbb{U}$ into n-grams and compute the embedding by taking the sum of the embeddings of all occuring n-grams, equivalent to how embeddings for out-of-vocabulary words are computed in fastText (Bojanowski et al., 2017).

$$\boldsymbol{u}_x = \sum_{g \in \mathbb{G}^{(x)}} \boldsymbol{w}_g$$

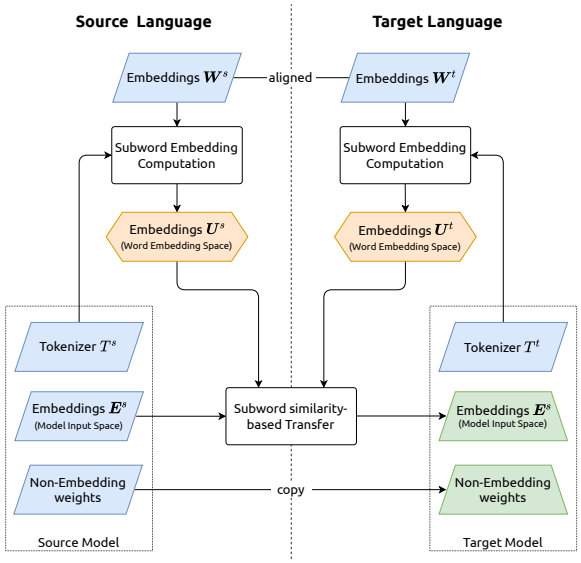

Figure 1: Summary of our WECHSEL method. We show inputs, intermediate results and outputs.

where $\mathbb{G}^{(x)}$ is the set of n-grams occuring in the subword x and $\boldsymbol{w}_g$ is the embedding of the n-gram g. Subwords in which no known n-gram occurs are initialized to zero.

3.2 Subword similarity-based Transfer

Applying the previous step to both source and target language results in the subword embeddings $\boldsymbol{U}^s$ and $\boldsymbol{U}^t$ over the subword vocabularies $\mathbb{U}^s$ and $\mathbb{U}^t$, respectively. Our aim is to leverage these embeddings to find an effective transformation from $\boldsymbol{E}^s$ to $\boldsymbol{E}^t$. We first compute the cosine similarity of every subword $x \in \mathbb{U}^t$ to every subword $y \in \mathbb{U}^s$, denoted as $s_{x,y}$.

$$s_{x,y} = \frac{\boldsymbol{u}_x^t \boldsymbol{u}_y^{sT}}{\|\boldsymbol{u}_x^t\| \|\boldsymbol{u}_y^s\|}$$

We now exploit these similarities to initialize embeddings in $\boldsymbol{E}^t$ by a convex combination of embeddings in $\boldsymbol{E}^s$. In particular, each subword embedding in $\boldsymbol{E}^t$ is defined as the weighted mean of the k nearest embeddings in $\boldsymbol{E}^s$ according to the similarity values. The weighting is done by a softmax of the similarities with temperature τ.

$$\boldsymbol{e}_x^t = \frac{\sum_{y \in \mathcal{J}_x} \exp\left(s_{x,y}/\tau\right) \cdot \boldsymbol{e}_y^s}{\sum_{y' \in \mathcal{J}_x} \exp\left(s_{x,y'}/\tau\right)}$$

where $\mathcal{J}_x$ is the set of k neighbouring subwords in the source language. Subword embeddings for which $\boldsymbol{U}^t$ is zero are initialized from a random normal distribution $\mathcal{N}(\mathbb{E}[\boldsymbol{E}^s], \mathrm{Var}[\boldsymbol{E}^s])$.

4 Experiment Design

We evaluate our method by transferring the English RoBERTa (Liu et al., 2019) and the English GPT-2 model (Radford et al., 2019) to French, German, Chinese and Swahili. We refer to these languages as *medium-resource languages*. In addition, we study the benefits of our method on four *low-resource languages*, namely Sundanese, Scottish Gaelic, Uyghur and Malagasy.

We evaluate WECHSEL-RoBERTa by fine-tuning on XNLI (Conneau et al., 2018), and on the balanced train-dev-test split of WikiANN (Rahimi et al., 2019; Pan et al., 2017) to evaluate NLI and NER performance, respectively. The hyperparameters used for fine-tuning are reported in Appendix B. GPT-2 is evaluated by Perplexity (PPL) on a held-out set from the same corpus on which the model was trained on.

Due to the difficulty of extrinsic evaluation on low-resource languages, we only train GPT-2 models in these languages, and evaluate their performance intrinsically via Language Modelling Perplexity on a held-out set. We use the pretrained models $\text{RoBERTa}_{\text{Base}}$ with 125M parameters, and the small GPT-2 variant with 117M parameters provided by HuggingFace's Transformers (Wolf et al., 2020) in all experiments.

To ensure our method does not depend on excessive amounts of training data in the target language, we use a subset of 4GiB from the OSCAR corpus (Ortiz Suárez et al., 2019) for German, French and Chinese. For the other languages, we use data from the CC-100 corpus (Conneau et al., 2020) which contains 1.6GiB, 0.1GiB, 0.1GiB, 0.4GiB and 0.2GiB for Swahili, Sundanese, Scottish Gaelic, Uyghur and Malagasy, respectively. To obtain aligned word embeddings between the source and the target language we use monolingual fastText word embeddings[2] (Bojanowski et al., 2017). We align these embeddings using the Orthogonal Procrustes method (Schönemann, 1966; Artetxe et al., 2016) with bilingual dictionaries from MUSE[3] (Conneau et al., 2017) for French, German and Chinese and a bilingual dictionary from FreeDict[4] (Bański and Wójtowicz, 2009) for Swahili. For the low-resource languages, we use bilingual dictionaries scraped from Wiktionary.[5]

[2]https://fasttext.cc
[3]https://github.com/facebookresearch/MUSE
[4]https://freedict.org
[5]available at `github.com/anonymized`

Model	Tokens trained on	Factor
WECHSEL-RoBERTa	65.5B	1.0x
TransInner-RoBERTa	65.5B	1.0x
FullRand-RoBERTa	65.5B	1.0x
CamemBERT	419.4B	6.4x
$\text{GBERT}_{\text{Base}}$	255.6B	3.9x
$\text{BERT}_{\text{Base}}$-Chinese	131.1B	2.0x

Table 1: Tokens trained on in the target language between our models and previous monolingual models.

We choose temperature $\tau = 0.1$ and neighbors $k = 10$ for WECHSEL by conducting a parameter search over a grid with varying values for k and τ using linear probes (Appendix A). We train tokenizers in the target languages using a vocabulary size of 50k tokens and byte-level BPE (Radford et al., 2019). After applying WECHSEL, we continue training RoBERTa on the MLM objective and GPT-2 on the CLM objective. We compare against two baseline methods.

- **TransInner:** Randomly initializing $\boldsymbol{E}^t$ while transferring all other parameters from the English model as in de Vries and Nissim (2021). After training only embeddings for a fixed amount of steps while freezing other parameters, the entire model is trained for the remaining steps. In preliminary experiments reported in Appendix E, we compare the method by Zoph et al. (2016) with TransInner, observing superior performance of TransInner, so we choose TransInner as the baseline for cross-lingual transfer in all our experiments.

- **FullRand:** Training from scratch in the target language, as is commonly done when training BERT-like or GPT-like models in a new language (Antoun et al., 2020; Louis, 2020; Chan et al., 2020; Martin et al., 2020).

All models are trained for 250k steps with the same hyperparameters across all languages (reported in Appendix B). Training one model takes around 4 days on a TPUv3-8. For WECHSEL and FullRand we use a learning rate (LR) schedule with linear warmup from zero to the peak LR for the first 10% of steps, followed by a linear decay to zero. For TransInner, we perform two warmup phases from zero to peak LR, once for the first 10% of steps for training embeddings only, then again for the remaining steps while training the entire model.

In addition to the mentioned baselines trained under this setting, we compare the results of

Lang	Model	Score@0			Score@25k			Score@250k			Score (more training)		
		NLI	NER	Avg	NLI	NER	Avg	NLI	NER	Avg	NLI	NER	Avg
French	WECHSEL-RoBERTa	78.25	86.93	82.59	81.63	90.26	85.95	**82.43**	**90.88**	**86.65**	-	-	-
	TransInner-RoBERTa	60.86	69.57	65.21	65.49	83.82	74.66	81.75	90.34	86.04	-	-	-
	FullRand-RoBERTa	55.71	70.79	63.25	69.02	84.24	76.63	75.28	89.30	82.29	-	-	-
	CamemBERT	-	-	-	-	-	-	-	-	-	80.88	90.26	85.57
	$\text{XLM-R}_{\text{Base}}$	-	-	-	-	-	-	-	-	-	79.25	89.48	84.37
German	WECHSEL-RoBERTa	75.64	84.53	80.08	81.11	89.05	85.08	**81.79**	**89.72**	**85.76**	-	-	-
	TransInner-RoBERTa	58.51	65.23	61.87	64.78	82.05	73.42	80.75	89.30	85.02	-	-	-
	FullRand-RoBERTa	54.82	66.84	60.83	68.02	81.53	74.77	75.48	88.36	81.92	-	-	-
	$\text{GBERT}_{\text{Base}}$	-	-	-	-	-	-	-	-	-	78.64	89.46	84.05
	$\text{XLM-R}_{\text{Base}}$	-	-	-	-	-	-	-	-	-	78.58	88.76	83.67
Chinese	WECHSEL-RoBERTa	63.23	72.79	68.01	77.19	79.07	78.13	**78.32**	80.55	**79.44**	-	-	-
	TransInner-RoBERTa	46.95	69.06	58.01	52.96	73.35	63.16	76.99	80.00	78.49	-	-	-
	FullRand-RoBERTa	44.24	57.95	51.09	58.34	64.84	61.59	71.38	78.35	74.86	-	-	-
	$\text{BERT}_{\text{Base}}$-Chinese	-	-	-	-	-	-	-	-	-	76.55	**82.05**	79.30
	$\text{XLM-R}_{\text{Base}}$	-	-	-	-	-	-	-	-	-	76.41	78.36	77.38
Swahili	WECHSEL-RoBERTa	60.28	74.38	67.33	73.87	87.63	80.75	**75.05**	**87.39**	**81.22**	-	-	-
	TransInner-RoBERTa	54.67	64.46	59.56	58.85	80.27	69.56	74.10	87.05	80.57	-	-	-
	FullRand-RoBERTa	50.59	62.35	56.47	63.79	83.49	73.64	70.34	87.34	78.84	-	-	-
	$\text{XLM-R}_{\text{Base}}$	-	-	-	-	-	-	-	-	-	69.18	87.37	78.28

Table 2: Results from fine-tuning RoBERTa models. We report accuracy for NLI on XNLI and micro F1 score for NER on WikiANN. Results are averaged over 3 runs. We report scores before training (**Score@0**), after 10% of steps (**Score@25k**) and after training (**Score@250k**). We also report results from fine-tuning prior monolingual models and XLM–R (**Score (more training)**), all trained on more tokens than our models. For each language, the best results in every column are indicated with underlines. The overall best results including the comparison with existing monolingual/multilingual models of comparable size are shown in bold.

RoBERTa models with existing comparable models trained from scratch with more training effort. We consider the total number of tokens the model has encountered in the target language, computed as the product of batch size $\times$ sequence length $\times$ train steps (shown in Table 1) as a proxy for training effort. We evaluate the performance of CamemBERT (Martin et al., 2020) (French), $\text{GBERT}_{\text{Base}}$ (Chan et al., 2020) (German), and $\text{BERT}_{\text{Base}}$-Chinese (Devlin et al., 2019) as existing monolingual LMs,[6] as well as $\text{XLM-R}_{\text{Base}}$ (Artetxe et al., 2020) as a high-performing multilingual LM.

5 Results

We present our results on transferring RoBERTa and GPT-2 from English to other languages, followed by analyzing training behavior. In Appendix C, we provide a qualitative assessment of how well subword tokens are mapped between the source and the target languages.

5.1 Transferring RoBERTa

Table 2 reports the evaluation results of RoBERTa. As shown, models initialized with WECHSEL outperform models trained from scratch and models initialized with TransInner across all languages. Surprisingly, close relatedness of the source and target language is not necessary to achieve effective transfer, as e. g. on NLI WECHSEL improves absolute accuracy by 7.15%, 6.31%, 6.94% and 4.71% over models trained from scratch for French, German, Chinese and Swahili, respectively.

We observe that our parameter transfer-based model consistently outperforms the previously released LMs on both monolingual and multilingual settings, while these models benefit from much larger training resources in terms of computation time and corpus size. In particular, the results show an improvement over $\text{XLM-R}_{\text{Base}}$ by an average 3.54% accuracy for NLI and 1.14% micro F1 score for NER. For NLI, we improve over the prior monolingual models by 1.55%, 3.15% and 1.77% absolute accuracy for French, German and Chinese, respectively. For NER, we observe improvements over monolingual models with 0.62% and 0.26% absolute micro F1 score improvement for French and German, respectively. For Chinese, the monolingual model $\text{BERT}_{\text{Base}}$-Chinese still outperforms our method by 1.5% absolute micro F1 score. We suspect that the discrepancy between NLI and NER is due to the limited training corpus size (max. 4GiB), while a larger corpus can potentially improve NER as more named entities appear (Martin et al., 2020).

The first two columns of Figure 2 show the performance of RoBERTa models on downstream

[6] To the best of our knowledge there is no monolingual model available for Swahili.

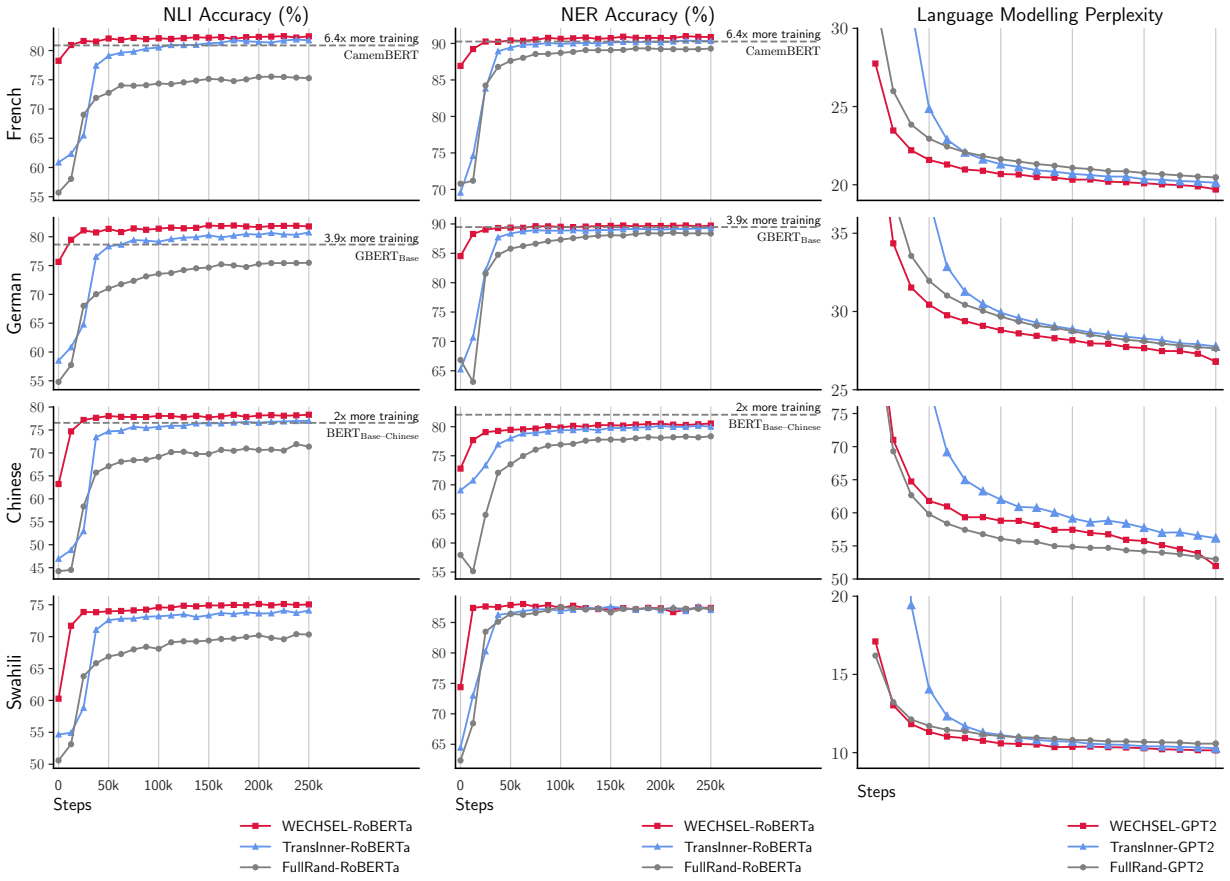

Figure 2: Test scores over training steps from fine-tuning RoBERTa models on NLI (using XNLI) and NER (using WikiANN). Perplexity on the held-out set over training steps of GPT-2 models. We evaluate every 12.5k steps.

Lang	Model	PPL@0	PPL@25k	PPL@250k
French	WECHSEL-GPT2	1.7e+3	23.47	**19.71**
	TransInner-GPT2	1.4e+5	67.97	20.13
	FullRand-GPT2	5.9e+4	25.99	20.47
German	WECHSEL-GPT2	3.7e+3	34.35	**26.80**
	TransInner-GPT2	1.5e+5	121.67	27.76
	FullRand-GPT2	5.8e+4	37.29	27.63
Chinese	WECHSEL-GPT2	2.4e+4	71.02	**51.97**
	TransInner-GPT2	1.5e+5	231.05	56.17
	FullRand-GPT2	5.8e+4	69.29	52.98
Swahili	WECHSEL-GPT2	1.4e+5	13.02	**10.14**
	TransInner-GPT2	1.4e+5	42.95	10.28
	FullRand-GPT2	5.8e+4	13.22	10.58

Table 3: Results of training GPT2 models. We report Perplexity before training (**PPL@0**), after 10% of steps (**PPL@25k**) and after training (**PPL@250k**).

Lang	Model	Best PPL
Sundanese	WECHSEL-GPT2	**111.72**
	TransInner-GPT2	151.86
	FullRand-GPT2	149.46
Scottish Gaelic	WECHSEL-GPT2	**16.43**
	TransInner-GPT2	18.62
	FullRand-GPT2	19.53
Uyghur	WECHSEL-GPT2	**34.33**
	TransInner-GPT2	39.06
	FullRand-GPT2	42.82
Malagasy	WECHSEL-GPT2	**14.01**
	TransInner-GPT2	14.85
	FullRand-GPT2	15.93

Table 4: Results of training GPT2 models on low-resource languages. We report the best Perplexity on the held-out set, evaluated every 2.5k steps.

tasks after each 12.5k training steps. Models initialized with WECHSEL reach high performance in significantly fewer steps than models initialized with FullRand or TransInner.

We expect FullRand-RoBERTa to approach performance of the respective prior monolingual models when trained on the same amount of tokens[7]. For French, WECHSEL-RoBERTa outperforms CamemBERT after 10% of training steps, reducing training effort by 64x. For German, WECHSEL-

[7]It would presumably be slightly worse because we restrict training corpus size to 4GiB.

RoBERTa outperforms $\text{GBERT}_{\text{Base}}$ after 10% of training steps, reducing training effort by 39x. For Chinese, WECHSEL-RoBERTa outperforms $\text{BERT}_{\text{Base}}$-Chinese on NLI, but does not outperform $\text{BERT}_{\text{Base}}$-Chinese on NER.

5.2 Transferring GPT-2

5.2.1 To Medium-Resource Languages

Results on medium-resource languages are shown in Table 3. Similar to the results for WECHSEL-RoBERTa, the GPT-2 models trained with WECHSEL consistently outperform the models trained from scratch and with TransInner across all languages.

The last column of Figure 2 depicts the performance of GPT-2 models after each 12.5k training steps. Comparing the results across all languages throughout training, we observe a stronger dependence on similarity of the source to the target language than for downstream tasks such as NLI or NER. In particular, for French and German, WECHSEL is consistently better than TransInner and FullRand throughout the entire training, while for Chinese, a decrease in perplexity towards the end of training causes WECHSEL to surpass training from scratch.

5.2.2 To Low-Resource Languages

Table 4 reports the perplexity of Language Modelling on the low-resource languages. Again, we observe consistent improvements using WECHSEL on all languages. Furthermore, as discussed in Appendix G, we conduct a sensitivity analysis w. r. t. the amount of available training data on French, studying the relation of performance improvement with training data size.

One difference of the low-resource models with the ones trained on medium-resource languages is that the low-resource LMs are prone to overfitting, and require appropriate model selection even in the early steps of training. Appendix F further elaborates on this by showing the performance of the low-resource LMs throughout training.

5.3 Is freezing necessary?

Previous work using the TransInner method freezes non-embedding parameters for a fixed amount of steps before training the entire model (de Vries and Nissim, 2021). This is done to prevent catastrophic forgetting at the beginning of training. To evaluate if freezing non-embedding parameters is still necessary with our method, we conduct an additional experiment. We train a German GPT-2 model with WECHSEL and a model with TransInner without freezing any parameters, and the same models with freezing of non-embedding parameters for the first 10% of steps. We match hyperparameters of the main experiments except training for 75k steps only. Based on the results shown in Figure 3, we conclude that freezing is necessary when using TransInner, but there is no need for freezing when using WECHSEL.

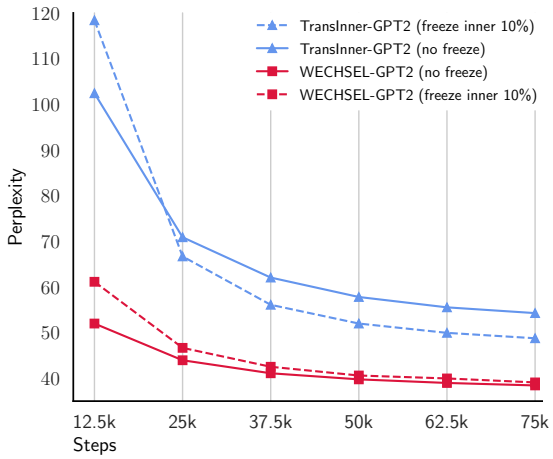

Figure 3: Comparison of German GPT-2 models trained with WECHSEL and TransInner between freezing non-embedding parameters at the start and not freezing any parameters.

6 Conclusion

We introduce WECHSEL, an effective method to transfer monolingual language models to new languages. WECHSEL exploits multilingual static word embeddings to compute an effective initialization of subword embeddings in the target language. We conduct experiments by transferring RoBERTa and GPT-2 models from English to French, German, Chinese and Swahili, as well as English GPT-2 to four low-resource languages. The evaluation results show that the transferred RoBERTa and GPT-2 models are more efficient and effective than strong baselines, and consistently outperform prior monolingual models that have been trained for a significantly longer time. WECHSEL facilitates the creation of effective monolingual LMs for new languages with medium to low resources, particularly in computationally-limited settings. In addition, our work provides strong evidence towards the hypothesis by Artetxe et al. (2020) that deep monolingual language models learn abstractions that generalize across languages. We discuss limitations and risks of our work in Appendix H.

References

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based model for Arabic language understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 9–15, Marseille, France. European Language Resource Association.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2016. Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2289–2294, Austin, Texas. Association for Computational Linguistics.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2017. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462, Vancouver, Canada. Association for Computational Linguistics.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 789–798, Melbourne, Australia. Association for Computational Linguistics.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, Online. Association for Computational Linguistics.

Piotr Bański and Beata Wójtowicz. 2009. Freedict: an open source repository of tei-encoded bilingual dictionaries.

Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 610–623.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6788–6796, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

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 *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2017. Word translation without parallel data. *arXiv preprint arXiv:1710.04087*.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel R. Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Wietse de Vries and Malvina Nissim. 2021. As good as new. how to successfully recycle English GPT-2 to make models for other languages. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 836–846, Online. Association for Computational Linguistics.

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.

Long Duong, Hiroshi Kanayama, Tengfei Ma, Steven Bird, and Trevor Cohn. 2016. Learning crosslingual word embeddings without bilingual corpora. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1285–1295, Austin, Texas. Association for Computational Linguistics.

William Fedus, Barret Zoph, and Noam Shazeer. 2021. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *arXiv preprint arXiv:2101.03961*.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Armand Joulin, Piotr Bojanowski, Tomas Mikolov, Hervé Jégou, and Edouard Grave. 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*, pages 2979–2984, Brussels, Belgium. Association for Computational Linguistics.

Guillaume Lample, Alexis Conneau, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *International Conference on Learning Representations*.

Mike Lewis, Marjan Ghazvininejad, Gargi Ghosh, Armen Aghajanyan, Sida Wang, and Luke Zettlemoyer. 2020. Pre-training via paraphrasing. In *Advances in Neural Information Processing Systems*, volume 33, pages 18470–18481. Curran Associates, Inc.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Antoine Louis. 2020. BelGPT-2: a GPT-2 model pretrained on French corpora. `https://github.com/antoiloui/belgpt2`.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Bilingual word representations with monolingual quality in mind. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 151–159, Denver, Colorado. Association for Computational Linguistics.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. 2020. CamemBERT: a tasty French language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219, Online. Association for Computational Linguistics.

Toan Q. Nguyen and David Chiang. 2017. Transfer learning across low-resource, related languages for neural machine translation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 296–301, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Debora Nozza, Federico Bianchi, and Dirk Hovy. 2020. What the [mask]? making sense of language-specific bert models. *arXiv preprint arXiv:2003.02912*.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipelines for processing huge corpora on medium to low resource infrastructures. Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC-7) 2019. Cardiff, 22nd July 2019, pages 9 – 16, Mannheim. Leibniz-Institut für Deutsche Sprache.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958, Vancouver, Canada. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding with unsupervised learning.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Afshin Rahimi, Yuan Li, and Trevor Cohn. 2019. Massively multilingual transfer for NER. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 151–164, Florence, Italy. Association for Computational Linguistics.

Ori Ram, Yuval Kirstain, Jonathan Berant, Amir Globerson, and Omer Levy. 2021. Few-shot question answering by pretraining span selection. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3066–3079, Online. Association for Computational Linguistics.

Peter H Schönemann. 1966. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31(1):1–10.

Ke Tran. 2020. From english to foreign languages: Transferring pre-trained language models. *arXiv preprint arXiv:2002.07306*.

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.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Shijie Wu and Mark Dredze. 2020. Are all languages created equal in multilingual BERT? In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 120–130, Online. Association for Computational Linguistics.

Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. 2015. Normalized word embedding and orthogonal transform for bilingual word translation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1006–1011, Denver, Colorado. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1568–1575, Austin, Texas. Association for Computational Linguistics.

A Grid search over k and τ

To choose number of neighbors k and temperature τ for WECHSEL we conduct a grid search over linear probes of models with different initialization shown in Table 8. For RoBERTa, we compute scores on NLI (using XNLI) and POS tagging (using the French, German and Chinese GSD corpora in Universal Dependencies) using linear probes of the last hidden state. We probe on NLI by taking a concatenation of the mean of all token representations in the premise with the mean of all token representations in the hypothesis. We probe on POS tagging by taking the mean of all token representations belonging to each word. For GPT2, we compute Language Modelling Perplexity on the held-out set also used to evaluate performance of the trained models.

B Hyperparameters

Hyperparameters used to fine-tune RoBERTa on downstream tasks are shown in Table 5. Hyperparameters used to train models in our main experiments are shown in Table 6.

Parameter	NLI	NER
peak learning rate	2e-5	2e-5
batch size	128	32
sequence length	128	128
Adam ϵ	1e-8	1e-8
Adam β_1	0.9	0.9
Adam β_2	0.999	0.999
train epochs	2	10
warmup	10% of steps	10% of steps
warmup schedule	linear	linear
LR decay	linear to zero	linear to zero

Table 5: Hyperparameters used to fine-tune RoBERTa models on NLI (XNLI) and NER (WikiANN).

Parameter	RoBERTa	GPT2
peak learning rate	1e-4	5e-4
batch size	512	512
sequence length	512	512
weight decay	0.01	0.01
Adam ϵ	1e-6	1e-6
Adam β_1	0.9	0.9
Adam β_2	0.98	0.98
train steps	250k	250k

Table 6: Hyperparameters of the models transferred from RoBERTa and GPT2.

C Qualitative subword correspondence

We show a small random sample of tokens in the target language and their closest English token (according to WECHSEL) in Table 7.

D Using Word Embeddings without subword information

As an alternative to n-gram decomposition, we introduce a method for mapping word embeddings to subword embeddings without using any subword information (shown in Figure 4). For this method, we require word frequency information in addition to the word embeddings. We apply the tokenizer T to every word v in $\mathbb{V}$ resulting in a set of subwords for each word. We define $\mathbb{V}^{(x)}$ as the set of words containing the subword x when tokenized. The embedding $\boldsymbol{u}_x$ of the subword x is then defined as the average of the embeddings of words in $\mathbb{V}^{(x)}$, weighted by the word frequencies.

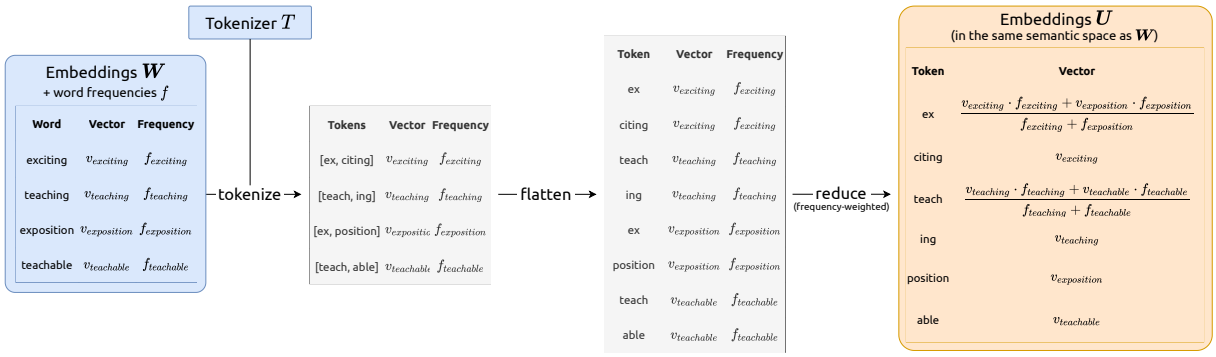

Figure 4: WECHSEL$_{\text{TFR}}$, an alternative subword embedding computation method. First, **tokenize** all words in the word embeddings. Then **flatten** the result by assigning the embeddings of the words in which it occured and their word frequencies to each subword. Finally, **reduce** the embeddings assigned to each subword by taking their mean, weighted by word frequency.

$$\boldsymbol{u}_x = \frac{\sum_{v \in \mathbb{V}(x)} \boldsymbol{w}_v \cdot f_v}{\sum_{v \in \mathbb{V}(x)} f_v}$$

where $\boldsymbol{w}_v$ is the embedding and f_v is the frequency of word v. We call this variant of our method WECHSEL$_{\text{TFR}}$. We evaluate WECHSEL$_{\text{TFR}}$ by training the same models as for WECHSEL. Results are shown in Table 9 for GPT2 and in Table 10 for RoBERTa. We find that, on average, performance is on par with WECHSEL.

E Choosing a transfer baseline

We consider two baseline methods to transfer models to a new language without using any language-specific information. One method is copying non-embedding parameters to the target language and initializing embeddings from a random normal distribution as done by de Vries and Nissim (2021). We refer to this method as TransInner. Another option is copying non-embedding parameters and assigning the embedding of a random token in the source language to each embedding in the target language (effectively "shuffling" the embeddings) as done by Zoph et al. (2016) and Nguyen and Chiang (2017). We refer to this method as TransInnerShuffleEmb. We evaluate these two methods using a setup equivalent to the experiments in Section 5.3 and find that TransInner performs slightly better than TransInnerShuffleEmb (Figure 6), so we use TransInner for subsequent experiments.

F Performance throughout training on low-resource languages

We show Language Modelling Perplexity on the held-out set throughout training in Figure 5.

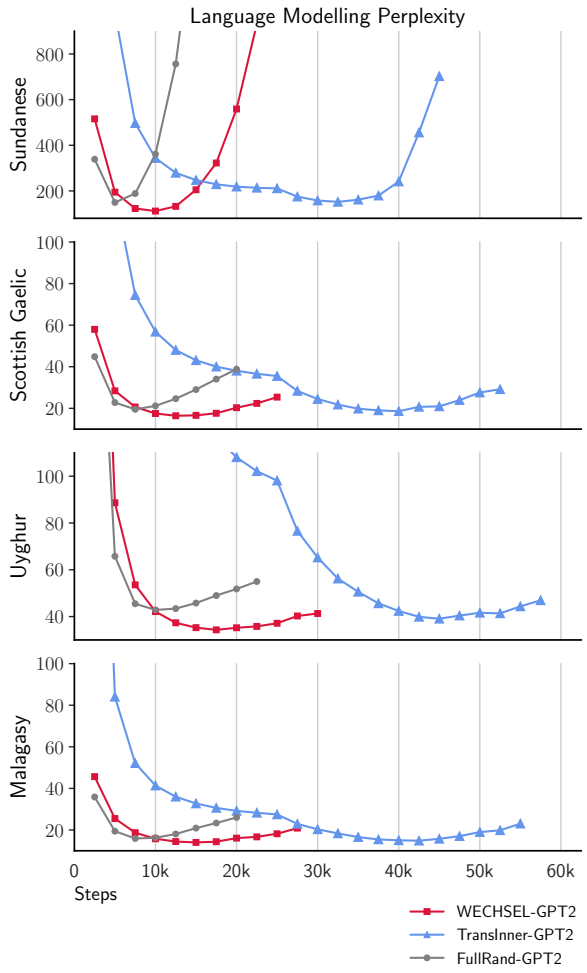

Figure 5: Perplexity throughout training on low-resource languages. We evaluate every 2.5k steps and stop training if Perplexity on the held-out set does not improve for 10k steps. TransInner-GPT2 takes more steps to overfit since all non-embedding parameters are frozen for the first 25k steps (c. f. Section 4).

Lang	Target Token	Closest English Token
French	héritage	legacy
	tremp	soaked
	épiscop	bishop
	scandaleux	udicrous
	vertig	astonishing
	enregistrer	rec
	sucrés	sweets
	Emmanuel	Emmanuel
	entourage	confid
	secrétariat	ariat
German	machen	ize
	mit	with
	Sprichwort	proverb
	erischen	Austrian
	minuten	utes
	Haustechnik	umbing
	dringen	urgent
	verfeinern	refine
	umgebung	vironments
	ternehmen	irms
Chinese	到处	everywhere
	巧合	coinc
	第三	third
	杂交	recomb
	利来	chnology
	政务	Govern
	石	stone
	喊麦	sing
	中海	iterranean
	张某	defendant
Swahili	shirikishe	ive
	Harusi	Marriage
	pesile	ery
	tihani	graduate
	changi	ool
	kuugua	ingestion
	kuzidi	acclaim
	vipigo	Trouble
	dhamiri	conscience
	aliposimama	Slowly

Table 7: Samples of tokens in each language and the corresponding closest tokens from the English vocabulary according to WECHSEL.

G Sensitivity Analysis w. r. t. training data size

Evaluating on languages with different amounts of available data only indirectly measures the effect of training data size on WECHSEL since other factors (e.g. language similarity to English) are also involved. We conduct a sensitivity analysis to make the relation to the amount of training data explicit (Table 11). Due to computational constraints we only do this for French. We find that the improvement from WECHSEL increases as the amount of training data decreases. In addition, we find that using fastText embeddings trained on less data deteriorates performance, but still leaves a clear margin to TransInner and FullRand.

Lang	Model	k	τ	Scores NLI	POS	LM
French		1	1	58.4	85.2	2.5e+5
		10	0.1	59.8	86.8	2.0e+5
	WECHSEL@0	10	1	58.3	84.4	4.8e+5
		50	0.1	57.2	83.6	3.1e+6
		50	1	54.0	81.6	1.8e+7
	FullRand@0	-	-	46.3	60.6	5.7e+6
	CamemBERT	-	-	63.5	93.6	-
German		1	1	55.8	72.7	6e+5
		10	0.1	58.9	76.0	4.2e+5
	WECHSEL@0	10	1	57.5	75.4	8.3e+6
		50	0.1	55.4	75.4	1.0e+7
		50	1	53.6	69.5	5.9e+7
	FullRand@0	-	-	44.5	49.1	6.2e+6
	$\text{GBERT}_{\text{Base}}$	-	-	63.2	81.4	-
Chinese		1	1	47.4	75.4	2.7e+6
		10	0.1	48.0	80.7	2.6e+6
	WECHSEL@0	10	1	48.3	80.3	3.1e+6
		50	0.1	48.3	77.8	3.7e+7
		50	1	47.9	76.5	8.6e+7
	FullRand@0	-	-	37.5	53.7	5.8e+6
	$\text{BERT}_{\text{Base}}$-Chinese	-	-	61.9	91.9	-

Table 8: Grid search over the temperature τ and number of most similar tokens k parameters of WECHSEL.

Lang	Model	PPL@0	PPL@25k	PPL@250k
French	WECHSEL-GPT2	1.7e+3	23.47	19.71
	$\text{WECHSEL}_{\text{TFR}}$-GPT2	2.3e+3	23.45	**19.70**
German	WECHSEL-GPT2	3.7e+3	34.35	**26.80**
	$\text{WECHSEL}_{\text{TFR}}$-GPT2	5.0e+3	34.46	26.82
Chinese	WECHSEL-GPT2	2.4e+4	71.02	**51.97**
	$\text{WECHSEL}_{\text{TFR}}$-GPT2	2.5e+4	72.11	52.07
Swahili	WECHSEL-GPT2	1.4e+5	13.02	10.14
	$\text{WECHSEL}_{\text{TFR}}$-GPT2	1.5e+5	13.03	**10.06**

Table 9: Results of training $\text{WECHSEL}_{\text{TFR}}$ GPT2 models. We report Perplexity before training (**PPL@0**), after 10% of steps (**PPL@25k**) and after training (**PPL@250k**).

H Limitations and Potential Risks

H.1 Limitations

We conduct our experiments on up to eight languages, showing the benefits of our parameter transfer method to both medium- and low-resource languages. However, there are many more languages with diverse linguistic characteristics on which our WECHSEL method is not tested. This is a limitation forced by computational constraints, as we can not ascertain whether transfer to all other languages would result in similar improvements. In addition, our extrinsic evaluation is limited to two tasks (NLI and NER). While this choice is due to the limitations on the available collections in various languages, this evaluation does not necessarily provide a comprehensive view of language understanding tasks.

Lang	Model	Score@0			Score@25k			Score@250k		
		NLI	NER	Avg	NLI	NER	Avg	NLI	NER	Avg
French	WECHSEL-RoBERTa	78.25	86.93	82.59	81.63	90.26	85.95	82.43	**90.88**	86.65
	WECHSEL$_{\text{TFR}}$-RoBERTa	78.25	87.43	82.84	81.86	90.07	85.96	**82.55**	90.80	**86.68**
German	WECHSEL-RoBERTa	75.64	84.53	80.08	81.11	89.05	85.08	81.79	**89.72**	85.76
	WECHSEL$_{\text{TFR}}$-RoBERTa	77.00	84.70	80.85	80.71	89.09	84.90	**82.04**	**89.72**	**85.88**
Chinese	WECHSEL-RoBERTa	63.23	72.79	68.01	77.19	79.07	78.13	**78.32**	80.55	**79.44**
	WECHSEL$_{\text{TFR}}$-RoBERTa	62.75	72.87	67.81	77.07	78.03	77.55	77.99	**80.65**	79.32
Swahili	WECHSEL-RoBERTa	60.28	74.38	67.33	73.87	87.63	80.75	**75.05**	87.39	**81.22**
	WECHSEL$_{\text{TFR}}$-RoBERTa	60.14	75.42	67.78	74.04	87.79	80.92	74.58	**87.66**	81.12

Table 10: Results from fine-tuning WECHSEL$_{\text{TFR}}$-RoBERTa models. Results shown the equivalently as in Table 2.

Model	Subsample Size	Best PPL 16MiB	64MiB	256MiB	1024MiB
WECHSEL-GPT2 (original fastText embeddings)		78.33	44.75	31.63	24.66
WECHSEL-GPT2 (fastText embeddings trained on subsample)		97.42	49.50	32.88	24.75
FullRand-GPT2		281.46	83.43	43.08	27.09
TransInner-GPT2		216.37	77.71	35.27	25.15

Table 11: Sensitivity Analysis w. r. t. the amount of training data on transfer to French. We train models on random subsamples of 16MiB, 64MiB, 256MiB and 1024MiB of the original training data, and evaluate on the same held-out set. For WECHSEL-GPT2, we train two models. One using the original, publicly available fastText embeddings trained on Common Crawl data. The other using fastText embeddings trained only on the corresponding subsample of text.

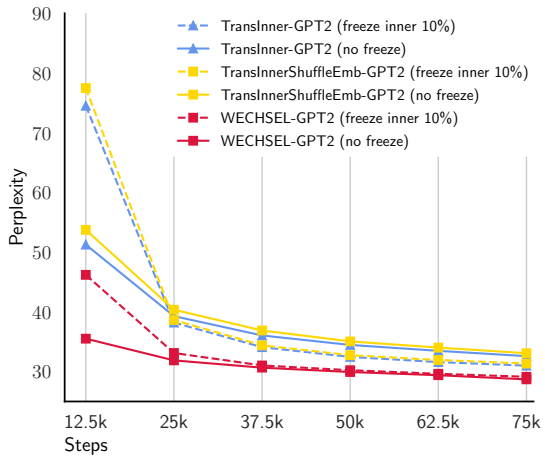

Figure 6: Comparison of German GPT-2 models trained with WECHSEL, TransInner and TransInner-ShuffleEmb between freezing non-embedding parameters at the start and not freezing any parameters.

H.2 Risks

It is well-known that existing LMs trained on English text encode societal biases and stereotypes and using them in downstream tasks might lead to unfair treatment of various social groups. Since we propose a method to transfer the English LMs to new languages, it is highly probable that the existing biases are also transferred to the target LMs. We therefore advocate a conscious and responsible use of the transferred LMs in practice.