An Empirical Comparison of Vocabulary Expansion and Initialization Approaches for Language Models

Nandini Mundra[1,*] **Aditya Nanda Kishore**[1,*] **Raj Dabre**[1,3,6]
Ratish Puduppully[4,†] **Anoop Kunchukuttan**[1,5] **Mitesh M. Khapra**[1,2]
[1]Indian Institute of Technology Madras [2]Nilekani Centre at AI4Bharat
[3]National Institute of Information and Communications Technology, Japan
[4]IT University of Copenhagen, Denmark [5]Microsoft India
[6]Indian Institute of Technology Bombay
Correspondence: miteshk@cse.iitm.ac.in, raj.dabre@nict.go.jp

Abstract

Language Models (LMs) excel in natural language processing tasks for English but show reduced performance in most other languages. This problem is commonly tackled by continually pre-training and fine-tuning these models for said languages. A significant issue in this process is the limited vocabulary coverage in the original model's tokenizer, leading to inadequate representation of new languages and necessitating an expansion of the tokenizer. The initialization of the embeddings corresponding to new vocabulary items presents a further challenge. Current strategies require cross-lingual embeddings and lack a solid theoretical foundation as well as comparisons with strong baselines. In this paper, we first establish theoretically that initializing within the convex hull of existing embeddings is a good initialization, followed by a novel but simple approach, *Constrained Word2Vec (CW2V)*, which does not require cross-lingual embeddings. Our study evaluates different initialization methods for expanding RoBERTa and LLaMA 2 across four languages and five tasks. The results show that CW2V performs equally well or even better than more advanced techniques. Additionally, simpler approaches like multivariate initialization perform on par with these advanced methods indicating that efficient large-scale multilingual continued pretraining can be achieved even with simpler initialization methods. We release our code publicly.[1]

1 Introduction

Language models are adept at a wide spectrum of natural language processing (NLP) tasks (Liu et al., 2023; Chung et al., 2024; Chowdhery et al., 2023; Wei et al., 2024; Goyal et al., 2023; Touvron et al., 2023). However, the best-performing language models work well for English but have inferior capabilities in other languages. A common method to improve the capabilities of other languages is to continually pre-train and finetune the English model for other languages (Conneau and Lample, 2019). This approach builds upon the capabilities acquired through large-scale English pre-training and focuses on aligning the English and other language spaces, making efficient re-use of compute and data resources (Cahyawijaya et al., 2023; Zhang et al., 2023). One of the major challenges for LLM adaptation is the lack of vocabulary coverage in the original model's tokenizer for the new language. This would mean the inability to represent the new language if the vocabulary is totally different or inefficient tokenization with high fertility in the case of inadequate vocabulary representation.

A solution is to expand the tokenizer to incorporate new vocabulary and then perform continual pre-training on monolingual data from the new language to adapt the model to the new language (Cui et al., 2023; Nguyen et al., 2023; Minixhofer et al., 2022). In this scenario, an important question is: *How do we initialize the embeddings of the new vocabulary items?* Various methods have been proposed in the literature for the initialization of the new token embeddings, from simple random initialization (Antoun et al., 2020; Martin et al., 2020) to the mean of embeddings (Gee et al., 2022) to sophisticated methods such as OFA among others (Minixhofer et al., 2022; Dobler and de Melo, 2023; Tran, 2020; Liu et al., 2024) that learn the new embeddings as a function of existing embeddings using external resources and tools like cross-lingual word-vectors and bilingual dictionaries. However, there is no theoretical basis for what constitutes a *good initialization*. Furthermore, in existing works, comparisons with simple, naive initialization methods across different model sizes are missing.

[*]Equal contribution.
[†]Work done while the author was at A*STAR, Singapore.
[1]https://github.com/AI4Bharat/VocabAdaptation_LLM/tree/CW2V

Proceedings of the 28th Conference on Computational Natural Language Learning, pages 84–104
November 15-16, 2024

In this paper, we theoretically define and analyze the properties of a *good initialization*. We prove that initializing embeddings of new vocabulary embeddings to be in the convex hull of original embeddings ensures that the greedy generation of the existing language(s) is not impacted by the new vocabulary items on initialization. Based on these insights, we propose a simple learnable initialization approach which we dub as *Constrained Word2Vec (CW2V)* which ensures initializations in the convex hull without needing cross-lingual embeddings. We conducted a comparative analysis of CW2V alongside 5 existing initialization strategies including OFA on two models containing varying parameters, namely RoBERTa (125M) and LLaMa2 (7B), examining their impact through 5 downstream tasks across 4 languages. Our analysis of various initialization methods demonstrates that CW2V achieves better if not comparable performance with the previous best methods. Additionally, we find that simpler methods like multivariate or mean initialization, which ensure new embeddings remain within the convex hull, are comparable with more advanced approaches such as OFA.

2 Related Work

Multilingual Models: To create a multilingual model for specific languages, one method is to train the model from scratch on the target languages using MLM and CLM objectives (Workshop et al., 2023; Conneau et al., 2020). However, this requires significant computational resources and data. A more efficient approach is to adapt an existing pre-trained language model (PLM) (Devlin et al., 2019; Touvron et al., 2023; Team, 2023) to the desired target language. There are two ways to adapt a PLM to a new language. The first is to fully adapt the model to the new language, replacing the source tokenizer and focusing only on the new language's performance (Minixhofer et al., 2022; Artetxe et al., 2020). The second is to keep the original language support and add the new language, ensuring the model still performs well on the source language (Garcia et al., 2021; Liu et al., 2024). In this work, we focus on extending the language support of the PLM rather than replacing it. We do this by extending the source tokenizer, which requires effectively initializing the model's embedding layer and LM head for the added tokens in the vocabulary.

Embedding Initialization Strategies: Previous work has focused on different initialization strategies. Methods like FVT (Gee et al., 2022) and Hewitt (2021) use the mean of source PLM embeddings, while WECHSEL (Minixhofer et al., 2022), RAMEN (Tran, 2020), FOCUS (Dobler and de Melo, 2023), and OFA (Liu et al., 2024) utilize external cross-lingual word vectors and source embeddings. However, these approaches rely on static embeddings. In contrast, we propose initialization strategy that learns new embeddings from the source PLM model and doesn't require static embeddings.

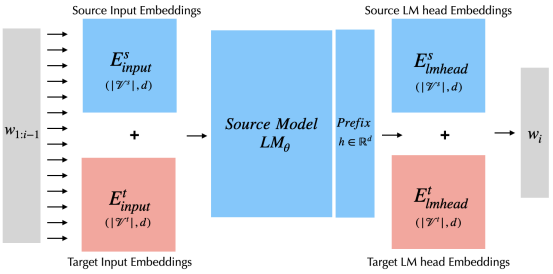

Figure 1: Setup for Vocabulary Expansion. Source model is shown in blue blocks, and expanded vocabulary embeddings are represented in red blocks. Source model parameters remain unchanged.

Continual Pre-training: A good initialization strategy provides a solid start for adapting a PLM to a new language by effectively initializing the new tokens in the embedding and LM head layers. However, to fully adapt the extended model to the new language, continued pre-training (CPT) (Wang et al., 2022; Alabi et al., 2022; Zhao et al., 2023) is essential. Therefore, we performed CPT on target languages post initialization.

3 Methodology

We describe the core methodology in this work followed by theoretical proofs of *good initializations* which motivate our own initialization approach, namely, Constrained Word2Vec.

3.1 Vocabulary Expansion

We adapt the same vocabulary expansion problem formulation as Hewitt (2021). Let θ be the parameters of a pre-trained neural source language model LM^s_θ, and let $\mathcal{V}^s = \{v^s_1, v^s_2, \ldots, v^s_n\}$ be the vocabulary of LM^s_θ. We will refer to $\mathcal{V}^s$ as the source vocabulary henceforth. Let $e^s_i \in \mathbb{R}^d$ be the sub-word embedding for word $i \in \mathcal{V}^s$. Let E^s denote the language modeling head's (henceforth *LM head*) embedding matrix of LM^s_θ and this is

our source embedding matrix. The probability of occurrence of the next word w_i given the previous word sequence $w_{1:i-1}$, $p_\theta(w_i \mid w_{1:i-1})$, is given by

$$p_\theta(w_i \mid w_{1:i-1}) = \frac{\exp\left(h_{i-1}^\top e^s_{w_i}\right)}{\sum_{j \in \mathcal{V}^s} \exp\left(h_{i-1}^\top e^s_j\right)},$$

where the prefix $h_{i-1} = \phi(w_{1:i-1}; LM^s_\theta) \in \mathbb{R}^d$ is the neural representation of the input using LM^s_θ.

In vocabulary expansion, we add n' new subwords $\notin \mathcal{V}^s$ forming the target vocabulary $\mathcal{V}^t = \{v^t_1, v^t_2, \ldots, v^t_{n'}\}$. This implies we need a new word embedding e^t_j for each $j \in \mathcal{V}^t$ comprising in E^t. The new language model $LM^t_{\theta'}$ has parameters $\theta' = \theta \cup \{e^t_j; j \in \mathcal{V}^t\}$. The output distribution of $LM^t_{\theta'}$ given by $p_{\theta'}(w_i \mid w_{1:i-1})$ is defined similarly as $p_\theta(w_i \mid w_{1:i-1})$ but with the normalization factor involving $\mathcal{V}^s \cup \mathcal{V}^t$.

Our goal is to find initializations for E^t such that the extended model not only retains its previous behavior but also can lead to good downstream performance for the languages corresponding to the new vocabulary with minimal continual pretraining. Retaining performance in English is particularly beneficial, as the knowledge embedded in English models often supports performance in other languages (Pires et al., 2019). Figure 1 gives an overview of our approach. Note that in our notations so far we have only mentioned the LM head, but just as the LM head has an expansion (E^t_{lmhead}), the input embedding matrix also has an expansion (E^t_{input}). This is trivial if both matrices are shared but in case they are not, we also need to find initializations for the latter. Following Hewitt (2021), we can use the same approach to initialize E^t_{input} as we do for E^t_{lmhead}.

3.2 What is a 'good' embedding initialization?

As we are ensuring that the model parameters θ remain unchanged at the initialization step, we can safely say that for the same word sequence $w_{1:i-1}$, where each word in the sequence belongs to $\mathcal{V}^s$, the prefix h_{i-1} at the output layer remains the same. Thus, the output word w_i strictly depends on the embeddings of the new words added to the vocabulary, as they determine the new partition function and the output probability distribution. The main goal of our analysis is to identify the set of initializations of new words that give us the same output before and after expansion for the prefixes formed by the original tokens. In other words, for the same input word sequence $w_{1:i-1}$, where $w_k \in \mathcal{V}^s \; \forall \; k \in [i-1]$, if w_i and w'_i represent the words predicted by language models LM^s_θ and $LM^t_{\theta'}$ respectively, i.e., $w_i = \text{argmax}_{j \in \mathcal{V}^s} \; p_\theta(j \mid w_{1:i-1})$ and $w'_i = \text{argmax}_{j \in \mathcal{V}^s \cup \mathcal{V}^t} \; p_{\theta'}(j \mid w_{1:i-1})$, we need $w_i = w'_i$. Let $e^t_1, e^t_2, \ldots, e^t_{n'} \in \mathbb{R}^d$ be the embedding initializations for words in $\mathcal{V}^t$. Therefore, a *good initialization* is an initialization $\left\{e^t_j; j \in \mathcal{V}^t\right\}$ that ensures, for any prefix $h_{i-1} \in \mathbb{R}^d$, the set of prefixes formed by word sequences from the source vocabulary, that is $'w_i = w'_i$.

3.3 Theorems

Theorem 1. *: A good initialization preserves the pre-expansion behavior.*

Let $e^s_1, e^s_2, e^s_3, ..., e^s_n \in \mathbb{R}^d$ be the embeddings of words in $\mathcal{V}^s$. Let $e^t_1, e^t_2, \ldots, e^t_{n'} \in \mathbb{R}^d$ be the embedding initializations for words in $\mathcal{V}^t$. If

$$\sup_{k \in \mathcal{V}^t}(h^T e^t_k) \leq \sup_{k \in \mathcal{V}^s}(h^T e^s_k) \tag{1}$$

holds for all $h \in \mathbb{R}^d$, then $\left\{e^t_j; j \in \mathcal{V}^t\right\}$ is a 'good' initialization.

Proof. Let $h = h_{i-1} \in \mathbb{R}^d$ be a prefix formed by a word sequence $w_{1:i-1}$, where $w_k \in \mathcal{V}^s \; \forall \; k \in [i-1]$. As condition 1 holds for all $h \in \mathbb{R}^d$, we can say that,

$$\sup_{k \in \mathcal{V}^t}(h^T e^t_k) \leq \sup_{k \in \mathcal{V}^s}(h^T e^s_k)$$
$$\implies \sup_{k \in \mathcal{V}^t} \exp(h^T e^t_k) \leq \sup_{k \in \mathcal{V}^s} \exp(h^T e^s_k)$$
$$\implies \sup_{k \in \mathcal{V}^t} \frac{\exp(h^T e^t_k)}{Z'} \leq \sup_{k \in \mathcal{V}^s} \frac{\exp(h^T e^s_k)}{Z'}$$

where, $Z' = \sum_{j \in \mathcal{V}^s \cup \mathcal{V}^t} \exp\left(h^\top e^t_j\right)$ is the new partition function, which is a positive constant as prefix and all the embeddings are given. We know that, $\frac{\exp(h^T e^t_k)}{Z'}$ represents the probability of occurrence of word corresponding to the embedding e^t_k at time step i. Thus, the inequality just says that probability of occurrence of any word from target vocabulary $\mathcal{V}^t$ is less than or equal to probability of occurrence of a word from source vocabulary. As the decoding at output layer is greedy, the output word is going to come from source vocabulary. We can guarantee that it remains the same as pre-expansion model's output word because the prefix remains the same before and after expansion as $w_k \in$ source vocabulary $\mathcal{V}^s \; \forall \; k \in [i-1]$.

Hence, as $w_i = w'_i$ and the embedding initialization $\left\{e^t_j; j \in \mathcal{V}^t\right\}$ is 'good'. □

Theorem 2. *: An initialization in the convex hull of source embeddings is good.*

If $y \in \mathcal{S}$, where $\mathcal{S}$ is the convex hull of the embeddings $e^s_1, e^s_2, e^s_3, \ldots, e^s_n$, then $(h^T y) \leq \sup_{k \in \mathcal{V}^s}(h^T e^s_k)$ for all $h \in \mathbb{R}^d$. Moreover, if $e^t_i \in \mathcal{S}$ for all $i \in \mathcal{V}^t$, then the initialization is 'good'.

Proof. Given $y \in \mathcal{S}$. Thus y can be written as $y = \sum_{j \in \mathcal{V}^s} \alpha_j e^s_j$ where $\sum_{j \in \mathcal{V}^s} \alpha_j = 1$ and $0 \leq w_j \leq 1 \; \forall \; j \in \mathcal{V}^s$. Thus, $\forall \; h \in \mathbb{R}^d$,

$$h^T y = \sum_{j \in \mathcal{V}^s} \alpha_j h^T e^s_j$$

.

As $0 \leq \alpha_j \leq 1 \; \forall \; j \in \mathcal{V}^s$,

$$(h^T y) \leq \sup_{k \in \mathcal{V}^s} (h^T e^s_k)$$

Given $e^t_i \in \mathcal{S} \;\; \forall \; i \in \mathcal{V}^t$

$$\implies (h^T e^t_i) \leq \sup_{k \in \mathcal{V}^s} (h^T e^s_k) \;\; \forall \; i \in \mathcal{V}^t \;\; \forall \; h \in \mathbb{R}^d$$

$$\implies \sup_{k \in \mathcal{V}^t} (h^T e^t_k) \leq \sup_{k \in \mathcal{V}^s} (h^T e^s_k) \;\; \forall \; h \in \mathbb{R}^d$$

Thus, from theorem 1 we can say that if $e^t_i \in \mathcal{S} \;\; \forall \; i \in \mathcal{V}^t$, then the initialization is *good*. □

We have showed that as long as we initialize every target embedding vector as a weighted average of source embeddings, the model output remains the same for the same prefix as long as it is obtained from a word sequence formed only by source vocabulary, thereby making it *good*. Table 5 verifies this empirically. In Appendix B we provide some additional theoretical analysis where we show a weaker converse of Theorem 2, that any *strongly good* initialization lies in the convex hull of source embeddings.

3.4 Our Approach: Constrained Word2Vec

Having established that a initializing in the convex hull of existing embeddings is *good*, we now propose *Constrained Word2Vec (CW2V)*, a novel approach to learn these initializations. Specifically, we constrain E^t as WE^s where $\sum_{j \in \mathcal{V}^s} W_{ij} = 1 \; \forall \; i \in \mathcal{V}^t$ and $W_{ij} \geq 0 \; \forall \; j \in \mathcal{V}^s, i \in \mathcal{V}^t$. Here, $E^s \in \mathbb{R}^{(|\mathcal{V}^s|, d)}$ is the source embedding matrix, $E^t \in \mathbb{R}^{(|\mathcal{V}^t|, d)}$ is the target embedding matrix and $W \in \mathbb{R}^{(|\mathcal{V}^t|, |\mathcal{V}^s|)}$ is the weight matrix that transforms E^s to E^t while ensuring the target embedding vectors reside inside the convex hull of the source embedding vectors. Our goal is to learn W.

Let $\mathcal{E}^t$ be the post-expansion embedding matrix of size $(|\mathcal{V}^s \cup \mathcal{V}^t|, d)$. In other words, $\mathcal{E}^t = [E^s; WE^s]$ where ; indicates concatenation along the vocabulary axis. By using $\mathcal{E}^t$ as the embedding matrix with W as the only learnable parameters, we propose a mechanism similar to Skip-gram (Mikolov et al., 2013) to obtain $\mathcal{E}^t$. In many modern PLMs, such as LLaMA, the input and output embedding layers are not tied, necessitating separate weight matrices for the input embedding and the LM head defined as $\mathcal{E}^s_{input} = [E^s_{input}; softmax(W_{input})E^s_{input}]$, $\mathcal{E}^s_{LM-head} = [E^s_{LM-head}; softmax(W_{LM-head})E^s_{LM-head}]^T$ with sizes $(|\mathcal{V}^s \cup \mathcal{V}^t|, d)$, $(d, |\mathcal{V}^s \cup \mathcal{V}^t|)$, respectively. The *softmax* operation ensures that the weights in each row add up to 1, thus assuring that the target embedding vectors remain in the convex hull of pre-expansion embeddings.

We set these embedding matrices $\mathcal{E}^t_{input}$ and $\mathcal{E}^t_{LM-head}$ up in the traditional Skip-gram architecture (Mikolov et al., 2013) as the word and context representation matrices. Similar to OFA (Liu et al., 2024), in order to make the learning computationally more efficient, we can also factorise W_{input} and $W_{LM-head}$ and learn the resulting parameters. This methodology can be extended to any PLM. If both the embedding layers are tied for a PLM (RoBERTa), we still learn two weight matrices and choose either for initializing E^t. To align target language embeddings with English, we trained the CW2V model on monolingual data from all languages and bilingual English-to-target dictionaries.

4 Experimental Setting

We now describe the models we focus on, the languages, downstream tasks and datasets, and implementation details.

4.1 Models

We use RoBERTa (Liu et al., 2019), an encoder based architecture and LLaMA2-7B (Touvron et al., 2023), a decoder based model and employ these models as the source models for our multilingual vocabulary expansion experiments.

4.2 Tokenizers

We use the RoBERTa tokenizer as the source tokenizer for experiments with RoBERTa and the LLaMA2 tokenizer as the source tokenizer for experiments with LLaMA2. Since we are focusing on multilingual transfer, we train a SentencePiece (Kudo and Richardson, 2018) tokenizer using textual data in target languages (German, Russian, Hindi, Tamil) and merge the obtained tokenizer with LLaMA2's tokenizer. The resulting tokenizer has 57K subwords in its vocabulary, and this merged tokenizer serves as the unified target tokenizer for all of our experiments, even for experiments with RoBERTa. We identify common subwords using a 'fuzzy' search similar to FOCUS (Dobler and de Melo, 2023) and OFA (Liu et al., 2024). We report the fertility score of the target tokenizer in all four target languages in Appendix D. Vocabulary expansion significantly reduces the fertilities for the languages considered.

4.3 Datasets and Languages

We extended the source model (English) to four target languages: Hindi, Tamil, Russian, and German. For all training, the Hindi and Tamil datasets were sourced from SANGRAHA (Khan et al., 2024), while the Russian, German, and English datasets were sourced from OSCAR (Ortiz Su'arez et al., 2020). To train the multilingual tokenizer, we used a monolingual dataset of 3 million sentences per target language, sourced from the tokenizer training data used in IndicTrans2 (Gala et al., 2023). For the constrained word2vec model training, we used a monolingual corpus of 2 million tokens per target language. Additionally, we incorporated bilingual dictionary datasets: Hindi and Tamil from (Kanojia et al., 2018), German from url [2] processed by (Bojar et al., 2014), and Russian from url [3]. Each expanded and initialized model underwent further pre-training on a multilingual dataset of 2.5 billion tokens, combining 500 million tokens per target language and 500 million English tokens.

4.4 Baselines

OFA The One For All (OFA) Framework (Liu et al., 2024) (Liu et al., 2024) uses multilingual static word vectors to inject alignment knowledge into the new subword embeddings. Regardless of the factorization approach, OFA initializes all new target embeddings using a weighted average of the source vocabulary embeddings, making OFA a 'strongly good' initialization.

Univariate Each target embedding is initialised by drawing values from 1-D Gaussian distributions parameterized by the mean and standard deviation of the source embeddings for each dimension. This was the primary baseline considered by OFA (Liu et al., 2024).

Multivariate Every target embedding is sampled from the multivariate gaussian distribution of embeddings whose mean and covariance come from the original embeddings E^s.

Mean Every target embedding is the average of pre-expansion embeddings. Mean initialization is used to initialize target vocabulary in FVT (Gee et al., 2022) and Hewitt (2021). This is a 'strongly' good initialization as mean of original embeddings belongs to the convex hull of original embeddings.

Random Every target embedding is randomly sampled from the $d-$dimensional guassian distribution $\mathcal{N}(0, 0.02I)$ where I is a $d-$dimensional identity matrix.

4.5 Constrained Word2Vec Training

We trained the constrained word2vec model using a similar setup to skip-gram (Mikolov et al., 2013) training. The context window size was set to 10 and negative sampling to 5. Additionally, we factorized the W_{input} and $W_{LM-head}$ matrices, with a factorized dimension of 1024. This factorization was done to reduce the number of trainable parameters, similar to OFA ((Liu et al., 2024)). Factorizing the weight matrices in the constrained word2vec model for RoBERTa reduced the number of trainable parameters from 758M to 59M, and for LLaMA2, it reduced from 1660M to 118M.

Model	Task Category	Task	Metric
RoBERTa	Sentence Classification	XNLI	Acc.
	Question Answering	QA	F1
	Token Classification	NER	F1
LLaMA2	Sentence Classification	XNLI	Acc.
	Machine Translation	FLORES	CHRF
	Question Answering	QA	F1
	Sentence Summarisation	XLSUM	BLEURT

Table 1: A summary of the tasks, datasets and metrics.

[2] https://nlp.stanford.edu/projects/nmt/data/wmt14.en-de/dict.en-de

[3] https://github.com/Badestrand/russian-dictionary

4.6 Downstream Tasks

We evaluated RoBERTa and LLaMA on various tasks, as shown in Table 1. For XNLI, we used XNLI (Conneau et al., 2018) for German, Russian, Hindi, and English, and IndicXNLI (Aggarwal et al., 2022) for Tamil. For NER, we used WikiANN (Pan et al., 2017). For QA, we used SQuAD (Rajpurkar et al., 2018) for German, Russian, Hindi, and English, and IndicQA (Doddapaneni et al., 2023) for Tamil. For Machine Translation, we used FLORES (Team et al., 2022). RoBERTa MLM checkpoints were fine-tuned on English and evaluated zero-shot on target languages. LLaMA CLM checkpoints were evaluated with 4-shot prompting. The metrics for each task are also listed in Table 1.

5 Results

We now describe the results of our investigation, where we first evaluate different initialization methods without continual pre-training or fine-tuning for RoBERTa and LLaMA2. We follow this up with results for continual pre-training and fine-tuning for RoBERTa, and continual pre-training and few-shot prompting for LLaMA2.

5.1 Impact of Initialization Methods

For the encoder-only RoBERTa model: Table 2 presents the performance of the expanded RoBERTa model initialized with Constrained Word2Vec, alongside baseline models, across three downstream tasks: XNLI, NER, and QA. The expanded and initialized model was not continually pre-trained but was fine-tuned till convergence on downstream task data. Firstly, looking at the columns labeled **en**, we can see that CW2V is better than any baseline for English, even OFA, indicating that it preserves the pre-expansion behavior of RoBERTa better than any other methods. Next, the scores under the **avg** columns indicate that CW2V is competitive with other approaches, especially OFA but tends to be slightly inferior. This means that CW2V mildly sacrifices the performance on other languages while strongly preserving the English performance.

For the decoder-only LLaMA2 model: Table 2 shows the performance of the expanded LLaMA2 model initialized with Constrained Word2Vec, alongside baselines, on the following downstream tasks: XNLI, Machine Translation, QA and XL-SUM (summarization). Here as well, the expanded and initialized model was not continually pre-trained but was evaluated using few-shot prompting. Different from the case of RoBERTa, the CW2V model significantly outperforms the OFA model across all tasks and languages despite not being continually pre-trained. CW2V achieves higher CHRF scores, averaged over all translation directions, in MT (17.02 En-X and 27.26 X-En) compared to OFA's 11.17 and 16.17, respectively. Similarly, for XNLI, QA and XLSUM, we observe that the average (**avg** column) performance over all languages for CW2V is vastly better than any other approach. The English-only performance (**en** column) however is comparable across all approaches with CW2V being only slightly better. This proves that in decoder-only models while CW2V is as good as any other approach for preserving the pre-expansion English-only performance, it is substantially better than other approaches for the new languages via vocabulary expansion.

5.2 Impact of Continual Pretraining

Here we show the compounding effects of continual pre-training and various initialization strategies to understand whether initialization matters or not when monolingual adaptation data exists.

For the encoder-only RoBERTa model: We evaluate the performance of expanded RoBERTa models initialized with Constrained Word2Vec (CW2V) and other baseline methods with CPT. We evaluate 15 checkpoints from one epoch of CPT (plus the initial checkpoint prior to CPT) on 3 downstream tasks. The results are depicted in Figure 2. Here, again, CW2V demonstrates comparable or superior performance to OFA, especially towards the latter stages of CPT. As illustrated in Figure 2, CW2V quickly converges with OFA (within less than 4 checkpoints) across all three tasks. Additionally, simpler baselines such as mean and multivariate also achieve comparable performance to OFA and CW2V shortly thereafter (in NER and QA, Multivariate catches up to CW2V within two checkpoints), demonstrating strong performance. This suggests that straightforward baselines like multivariate can be as effective as sophisticated methods such as Constrained Word2Vec and OFA. Furthermore, our analysis consistently shows that Univariate and Random initialization methods underperform in comparison to CW2V, OFA, Multivariate[4], and Mean. This highlights that Univariate

[4]Multivariate initialization has a high probability of re-

	RoBERTa						LLaMA2							
	XNLI		NER		QA		MT		XNLI		QA		XLSUM	
	en	avg	en	avg	en	avg	En-X	X-En	en	avg	en	avg	en	avg
CW2V	**86.0**	36.0	**82.2**	21.5	**90.7**	9.0	**17.0**	**27.3**	60.4	**38.1**	**77.7**	**35.8**	**0.6**	**0.4**
OFA	85.6	**37.7**	81.9	**21.7**	90.6	**12.0**	11.2	16.2	60.4	37.1	76.0	26.0	0.6	0.3
Multivariate	85.7	35.7	81.8	18.3	90.4	9.5	11.1	16.1	60.4	37.2	77.5	28.7	0.5	0.2
Univariate	85.6	36.6	82.0	22.0	90.7	10.3	11.1	16.0	60.4	37.2	77.4	28.7	0.5	0.3
Mean	85.5	36.0	81.5	20.3	90.5	8.8	11.1	16.2	**60.5**	37.2	77.4	28.7	0.5	0.3
Random	85.8	35.9	81.6	21.0	90.3	9.6	0.0	0.0	33.3	33.3	0.0	0.0	0.0	0.0

Table 2: Performance of the expanded RoBERTa and LLaMA2 models initialized with Constrained Word2Vec and baselines on downstream tasks across 5 languages.

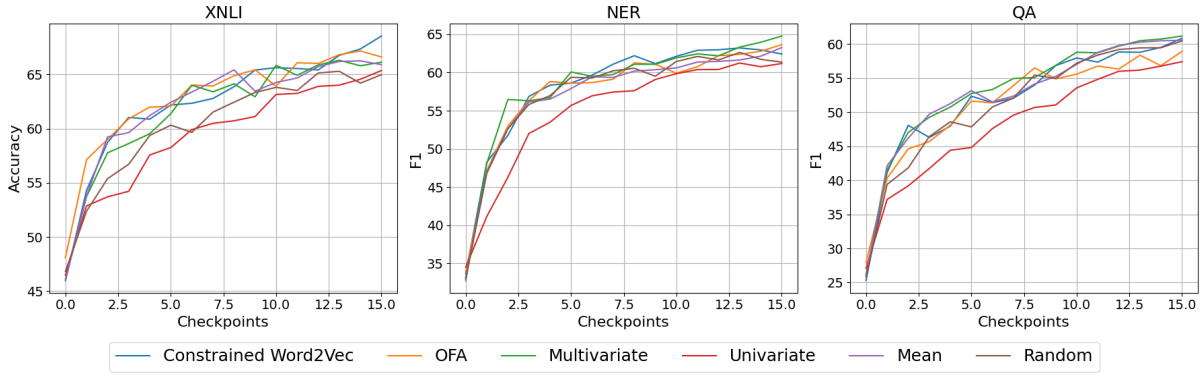

Figure 2: Evaluation of different initialization methods on expanded RoBERTa models using three multilingual tasks (XNLI, NER, QA) at 15 CPT checkpoints. The plots show average performance across five languages.

and Random methods, despite being used as primary baselines in previous work, are inadequate for comparison.

For the decoder-only LLaMA2 model: Similarly, we observe the performance of the expanded LLaMA2 models initialized with Constrained Word2Vec and the baselines. We evaluate 5 checkpoints from one epoch of CPT (plus the initial checkpoint prior to CPT) on 4 downstream tasks. The results are depicted in Figure 3. For MT and QA, both generative tasks, on average, CW2V is better if not comparable with OFA while being consistently better than all other approaches. We see that CW2V quickly surpasses OFA in 2-3 checkpoints. In the case of XLSUM, however, OFA tends to be better during intermediate checkpoints (1, 2, 3), but CW2V eventually performs just as well afterwards. Once again, CW2V (and OFA) are significantly better than other baselines.

XNLI is the only confounding task since no clear trends can be observed over various CPT stages. Furthermore, all models perform almost equally poorly, indicating that neither vocabulary expansion nor CPT is sufficient to improve XNLI performance. We suppose that fine-tuning on an XNLI dataset may shed further light on this, but due to limited compute, we did not pursue fine-tuning for any task and hence leave it as future work. Overall, CW2V is a highly effective initialization strategy for CPT, particularly benefiting languages that we aim to support more effectively through vocabulary expansion.

5.3 Catastrophic Forgetting in English tasks

Here we reveal something concerning about the inevitable negative effect of CPT on the pre-expansion language (English). During continued pre-training on monolingual datasets in both target and source languages, even with the source language (English) constituting 20% of the total dataset, we observed an initial drop in English performance. Figure 4 shows the performance of the expanded RoBERTa models at various CPT checkpoints on only English tasks. Initially, performance drops, after which it begins to improve with prolonged training without comprising performance on non-english tasks. This suggests that adjusting the model to learn new target language data tem-

siding within the convex hull of the source embeddings (Appendix F)

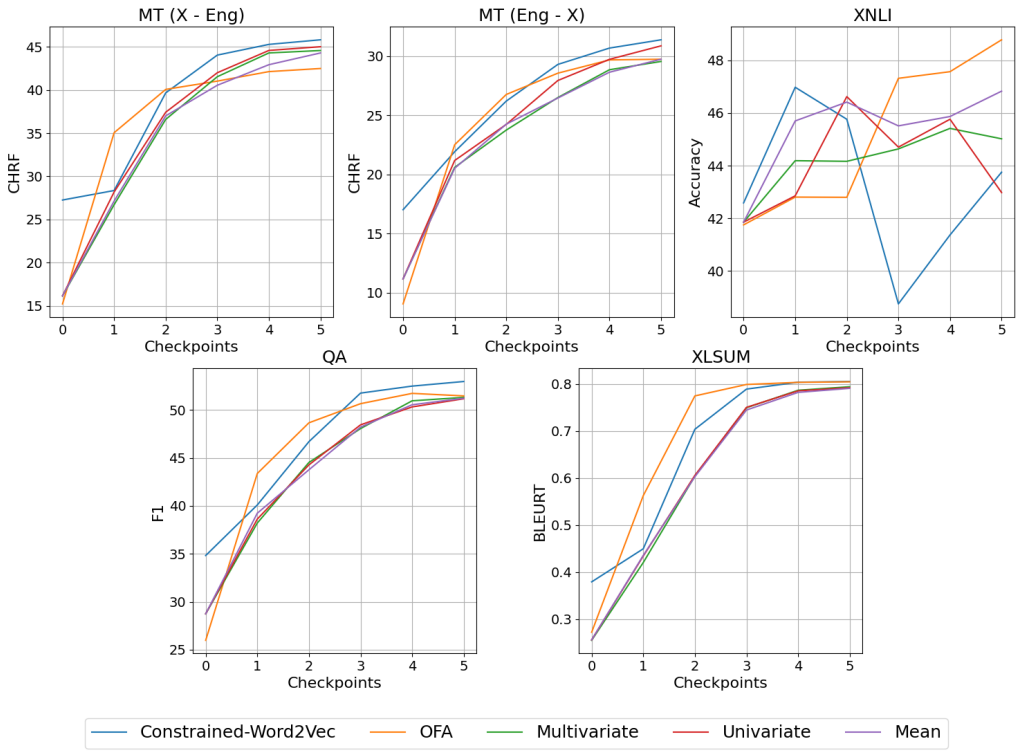

Figure 3: 4-shot XNLI, MT, QA, XLSUM evaluation of different initialization methods on expanded LLaMA2 models at 5 equidistant CPT checkpoints. MT plots show average performance across 4 languages, and XNLI, QA, XLSUM plots show average performance across 5 languages.

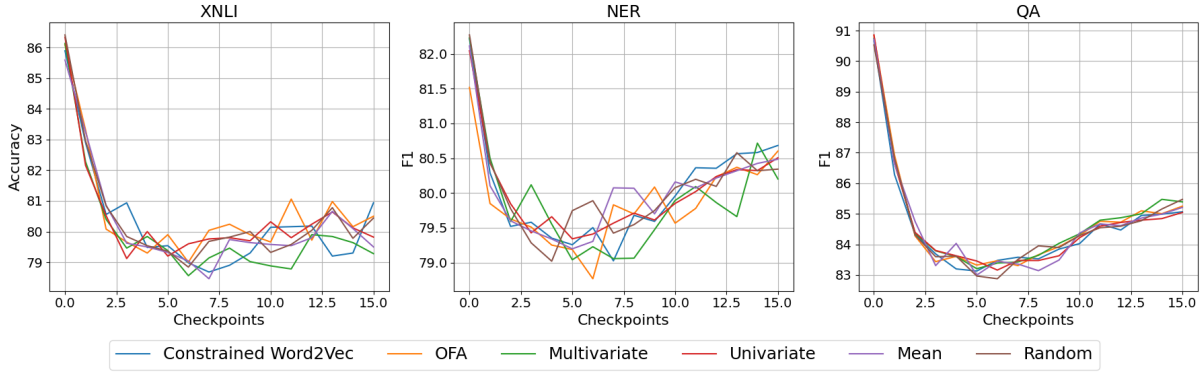

Figure 4: Assessment of English performance for various initialization methods on expanded RoBERTa models across three downstream tasks (XNLI, NER, QA) at 15 CPT checkpoints.

porarily disrupts the weights previously optimized for English but prolonged training could potentially further restore and enhance English performance.

6 Conclusion

In this work, we establish that effective embedding initialization for an expanded vocabulary in language models can be achieved within the convex hull of source vocabulary embeddings. We introduce a data-driven initialization method, *Constrained Word2Vec (CW2V)*, which learns the target embeddings by constraining them in the convex hull of the source embeddings. Our comparison of various initialization methods reveals that Constrained Word2Vec performs on par with other advanced techniques. Additionally, we find that simple methods like Multivariate and Mean, which ensure new embeddings lie within the convex hull

of source embeddings, perform comparably well to more complex approaches. This indicates that efficient large-scale multilingual continued pretraining can be possible even with simpler methods, provided they are *good* initialization strategies.

References

Divyanshu Aggarwal, Vivek Gupta, and Anoop Kunchukuttan. 2022. IndicXNLI: Evaluating multilingual inference for Indian languages. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10994–11006, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Adam Alabi, Saleha Nawaz, and Vincent Ng. 2022. Alabi: A light-weight approach for multilingual biomedical language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9717–9727.

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, 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.

Ondrej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Ale s Tamchyna. 2014. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 12–58, Baltimore, Maryland, USA. Association for Computational Linguistics.

Samuel Cahyawijaya, Holy Lovenia, Tiezheng Yu, Willy Chung, and Pascale Fung. 2023. InstructAlign: High-and-low resource language alignment via continual crosslingual instruction tuning. In *Proceedings of the First Workshop in South East Asian Language Processing*, pages 55–78, Nusa Dua, Bali, Indonesia. Association for Computational Linguistics.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2024. Scaling instruction-finetuned language models. *J. Mach. Learn. Res.*, 25:70:1–70:53.

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 and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel 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*, pages 2475–2485, Brussels, Belgium. Association for Computational Linguistics.

Yiming Cui, Ziqing Yang, and Xin Yao. 2023. Efficient and effective text encoding for chinese llama and alpaca. *arXiv preprint arXiv:2304.08177*.

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.

Konstantin Dobler and Gerard de Melo. 2023. Focus: Effective embedding initialization for monolingual specialization of multilingual models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Sumanth Doddapaneni, Rahul Aralikatte, Gowtham Ramesh, Shreya Goyal, Mitesh M. Khapra, Anoop Kunchukuttan, and Pratyush Kumar. 2023. Towards leaving no Indic language behind: Building monolingual corpora, benchmark and models for Indic languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12402–12426, Toronto, Canada. Association for Computational Linguistics.

Jay Gala, Pranjal A Chitale, A K Raghavan, Varun Gumma, Sumanth Doddapaneni, Aswanth Kumar M, Janki Atul Nawale, Anupama Sujatha, Ratish Puduppully, Vivek Raghavan, Pratyush Kumar, Mitesh M Khapra, Raj Dabre, and Anoop Kunchukuttan. 2023. Indictrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages. *Transactions on Machine Learning Research*.

Xavier Garcia, Noah Constant, Ankur Parikh, and Orhan Firat. 2021. Towards continual learning for multilingual machine translation via vocabulary substitution. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1184–1192, Online. Association for Computational Linguistics.

Leonidas Gee, Andrea Zugarini, Leonardo Rigutini, and Paolo Torroni. 2022. Fast vocabulary transfer for language model compression. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 409–416, Abu Dhabi, UAE. Association for Computational Linguistics.

Tanya Goyal, Junyi Jessy Li, and Greg Durrett. 2023. News summarization and evaluation in the era of gpt-3.

John Hewitt. 2021. Initializing new word embeddings for pretrained language models.

Diptesh Kanojia, Kevin Patel, and Pushpak Bhattacharyya. 2018. Indian Language Wordnets and their Linkages with Princeton WordNet. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Mohammed Khan, Priyam Mehta, Ananth Sankar, Umashankar Kumaravelan, Sumanth Doddapaneni, Suriyaprasaad B, Varun G, Sparsh Jain, Anoop Kunchukuttan, Pratyush Kumar, Raj Dabre, and Mitesh Khapra. 2024. IndicLLMSuite: A blueprint for creating pre-training and fine-tuning datasets for Indian languages. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15831–15879, Bangkok, Thailand. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A method for stochastic optimization.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *CoRR*, abs/1808.06226.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.*, 55(9).

Yihong Liu, Peiqin Lin, Mingyang Wang, and Hinrich Schuetze. 2024. OFA: A framework of initializing unseen subword embeddings for efficient large-scale multilingual continued pretraining. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1067–1097, Mexico City, Mexico. Association for Computational Linguistics.

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. *CoRR*, abs/1907.11692.

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.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

Benjamin Minixhofer, Fabian Paischer, and Navid Rekabsaz. 2022. WECHSEL: Effective initialization of subword embeddings for cross-lingual transfer of monolingual language models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3992–4006, Seattle, United States. Association for Computational Linguistics.

Xuan-Phi Nguyen, Wenxuan Zhang, Xin Li, Mahani Aljunied, Qingyu Tan, Liying Cheng, Guanzheng Chen, Yue Deng, Sen Yang, Chaoqun Liu, et al. 2023. Seallms–large language models for southeast asia. *arXiv preprint arXiv:2312.00738*.

Pedro Javier Ortiz Su'arez, Laurent Romary, and Benoit Sagot. 2020. A monolingual approach to contextualized word embeddings for mid-resource languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1703–1714, Online. Association for Computational Linguistics.

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.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

MosaicML NLP Team. 2023. Introducing mpt-7b: A new standard for open-source, commercially usable llms. Accessed: 2023-05-05.

NLLB Team, Marta R. Costa-jussà, and James Cross et al. 2022. No language left behind: Scaling human-centered machine translation.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, and Amjad Almahairi et al. 2023. Llama 2: Open foundation and fine-tuned chat models.

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

Alex Wang, Yequan Li, Yunpeng Zou, and Tim Menzies. 2022. Multimodal pretraining for ranking multilingual text and code. In *Proceedings of the 2022 ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1044–1053.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2024. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

BigScience Workshop, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, and et al. 2023. Bloom: A 176b-parameter open-access multilingual language model.

Shaolei Zhang, Qingkai Fang, Zhuocheng Zhang, Zhengrui Ma, Yan Zhou, Langlin Huang, Mengyu Bu, Shangtong Gui, Yunji Chen, Xilin Chen, and Yang Feng. 2023. Bayling: Bridging cross-lingual alignment and instruction following through interactive translation for large language models.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. 2023. A survey of large language models.

A Limitations

In this work, we identify the following limitations:

- Due to limited computational resources, we could not explore a variety of pre-trained models beyond RoBERTa and LLaMA2. However, since most language models function similarly, we expect our methods and findings to be generally applicable.
- For LLaMA2 models, we only conduct few-shot prompting for downstream task evaluation due to resource constraints. Nonetheless, based on our observations with RoBERTa, fine-tuning on downstream tasks will likely show that CW2V and OFA are only marginally better than other approaches.
- Although we evaluated only five downstream tasks, we cannot confirm that our observations will apply to all types of tasks. This remains an area for future research.
- We show experiments on four languages—Hindi, German, Russian, and Tamil—due to limited computational resources. However, as we have chosen languages from different scripts, we expect our methods and findings to be generally applicable.

B Further Analysis

Theorem 3. *: All strongly good initializations are in the convex hull.*

Let $e^s_1, e^s_2, e^s_3, ..., e^s_n \in \mathbb{R}^d$ be the embeddings of words in $\mathcal{V}^s$. Let $y \in \mathbb{R}^d$. If $(h^T y) \leq \sup_{k \in \mathcal{V}^s}(h^T e^s_k)$ for all $h \in \mathbb{R}^d$, then $y \in \mathcal{S}$, where $\mathcal{S}$ is the convex hull of the embeddings $e^s_1, e^s_2, e^s_3, ..., e^s_n$.

Proof. We prove this using contradiction. Say, $y \notin \mathcal{S}$ and $(h^T y) \leq \sup_{k \in \mathcal{V}^s}(h^T e^s_k)$ holds good for all $h \in \mathbb{R}^d$. Since, $\mathcal{S}$ is closed and convex and $y \notin \mathcal{S}$, there exists a hyperplane $\mathbb{H}$ that strictly separates y from $\mathcal{S}$. This hyperplane defines a half space $\mathcal{H}$ containing $\mathcal{S}$. Note that $\mathcal{H}$ contains $\mathcal{S}$ and $y \notin \mathcal{H}$

Let $\vec{b} \in \mathbb{R}^d$ be a point on the hyperplane $\mathbb{H}$. Let $\vec{n} \in \mathbb{R}^d$ denote the normal to the hyperplane $\mathbb{H}$. We choose $\vec{n}$ in such a way that any point $\vec{r} \in \mathcal{S}$ satisfies,

$$(\vec{r} - \vec{b})^T \vec{n} \leq 0$$

Thus, any embedding $e^s \in \{e^s_1, e^s_2, ..., e^s_n\}$ satisfies,

$$(e^s - b)^T \vec{n} \leq 0 \tag{2}$$

and any point $\vec{q} \notin \mathcal{H}$ satisfies,

$$(\vec{q} - \vec{b})^T \vec{n} \geq 0$$

As $y \notin \mathcal{H}$,

$$(y - \vec{b})^T \vec{n} \geq 0 \tag{3}$$

Equations 2 and 3 imply,

$$\vec{n}^T e^s \leq \vec{n}^T y \ \forall \ e^s \in \{e^s_1, e^s_2, ..., e^s_n\} \tag{4}$$

Thus, $\sup_{k \in \mathcal{V}^s}(\vec{n}^T e^s_k) \leq (\vec{n}^T y)$ which contradicts the statement that $(h^T y) \leq \sup_{k \in \mathcal{V}^s}(h^T e^s_k)$ holds good for all $h \in \mathbb{R}^d$ as it fails for $h = \vec{n}$.

Thus, if $(h^T y) \leq \sup_{k \in \mathcal{V}^s}(h^T e^s_k)$ for all $h \in \mathbb{R}^d$, then $y \in \mathcal{S}$, where $\mathcal{S}$ is the convex hull of the embeddings $e^s_1, e^s_2, e^s_3, ..., e^s_n$.

□

Thus, from theorem 3 we can say that any 'strongly good' initialization must lie in the convex hull of pre-expansion embeddings. But for an initialization to be considered 'good', the output word must remain unchanged for prefixes formed by word sequences from the source vocabulary. This implies that the condition 1 only needs to be satisfied for a subset of $\mathbb{R}^d$, rather than for all $h \in \mathbb{R}^d$. Thus, it is not necessary that the converse of Theorem 2 to be true as we can have initializations which are 'good' but not 'strongly good'. However, we can say that if an initialization is 'strongly good', embeddings must lie in the convex hull of pre-expansion embeddings.

C Effect on Initialisation on Model Output

Random initialization of new embeddings can result in a pre-trained language model assigning a probability of 1 to new words and can degrade domain adaptation performance (Hewitt, 2021). Figure 5 shows the outputs of expanded LLaMA2 models for an English sentence prompt. Random initialization of expanded tokens results in gibberish, while the other three methods produce outputs identical to the base LLaMA2 model, as they ensure embeddings lie within the convex hull of source embeddings.

Initialization	Output
LLaMA2- Base	the same thing every day.

CW2V	the same thing every day.
OFA	the same thing every day.
Mean	the same thing every day.
Random	ো্ট উপায়ুক্তৰ উপায়ুক্তৰ উপা

Figure 5: Expanded LLaMA2 Model Outputs for the Prompt : *"I don't want to eat" for various initializations.*

D Fertility Score

Fertility Score	English	Hindi	Tamil	Russian	German
LLaMA2 Tokenizer	2.89	7.47	12.66	4.25	3.88
RoBERTa Tokenizer	2.87	10.85	28.80	9.89	4.42
Extended Tokenizer	**2.87**	**2.83**	**2.83**	**3.74**	**3.88**

Table 3: Fertility scores for the source and the extended tokenizers on all the languages

Table 3 shows the fertility scores of the target tokenizer with respect to source tokenizer on 5 languages considered.

E Tokenizer Coverage

	Target Tokenizer		
	Copied Tokens	Initialized Tokens	Coverage
RoBERTa	22K	35K	38.5 %
LLaMA2	32K	25K	56.14 %

Table 4: : The number of subwords being initialized by copying from the original embeddings from RoBERTa's and LLaMA's tokenizers.

Table 4 shows the size of source vocabulary in experiments with RoBERTa and LLaMA2. As the new vocabulary is extended from LLaMA2, many subword embeddings are directly copied when using LLaMA2 as the source model. We employed a 'fuzzy' search similar to FOCUS (Dobler and de Melo, 2023) to identify the common tokens between the target tokenizer and the RoBERTa tokenizer. This led to a 38.5 % coverage of tokens leading us to a source vocabulary of size 22K for experiments with RoBERTa.

F Do Multivariate and Univariate initializations reside in the hull?

In multivariate initialization, we sample from a multivariate Gaussian that considers correlations across dimensions, unlike the univariate distribution. When dealing with strongly correlated dimensions (positive or negative), a multivariate approach proves advantageous. By considering the correlations across dimensions, we can sample new embeddings that are positioned more effectively within the latent space of original embedding distribution. However, there is no straightforward method to determine if embedding sampled from either distribution lies within the hull. To ensure that multivariate initialization remains within the convex hull with a high confidence, we also scaled the covariance matrix by a factor of 1e-5. In contrast, unscaled univariate initialization was used as a baseline, aligning with previous studies (Liu et al., 2024). (Hewitt, 2021) recommends employing multivariate initialization to incorporate noise. Notably, as illustrated in Figure 2, multivariate initialization significantly outperforms univariate initialization and closely approaches the performance of OFA in encoder-based models. However, a comprehensive theoretical analysis is required to determine if unscaled multivariate initialization has a higher likelihood of being within the convex hull compared to univariate initialization. This aspect is left for future research, given the empirical observation that univariate initializations typically exhibits lower performance compared to scaled multivariate initialization.

G Continued Pretraining Details

All the expanded and initialized RoBERTa models are trained on the same hyperparameters used in OFA (Liu et al., 2024). Specifically, we employ the MLM objective with a standard mask rate of 15%. We utilize the Adam optimizer (Kingma and Ba, 2017) with parameters ($\beta_1 = 0.9, \beta_2 = 0.999$) and $\epsilon = 1 \times 10^{-6}$. The initial learning rate is set to 5×10^{-5}. The only deviation from our approach compared to OFA is the batch size, which is fixed at 32. Each batch consists of training samples concatenated up to the maximum sequence length of 512, randomly selected from all language-scripts described in Section 4.3. We continue to pretrain using the scripts adapted from HuggingFace[5].

For LLaMa2, we used the standard LM objective with a context length of 2048 subwords. We used the Adam optimizer with linear warmup and decay where the peak learning rate was 5×10^{-5} and warmup was done till 10% of training steps. We

[5] https://github.com/huggingface/

trained for 1 epoch over our data saved checkpoints every 20% of an epoch enabling us to study model behavior against increasing training data.

H Complete Results for Each Task and Language

Results for each task in all the languages across all the checkpoints is given in figures 6, 7, 8, 9, 10, 11, 12

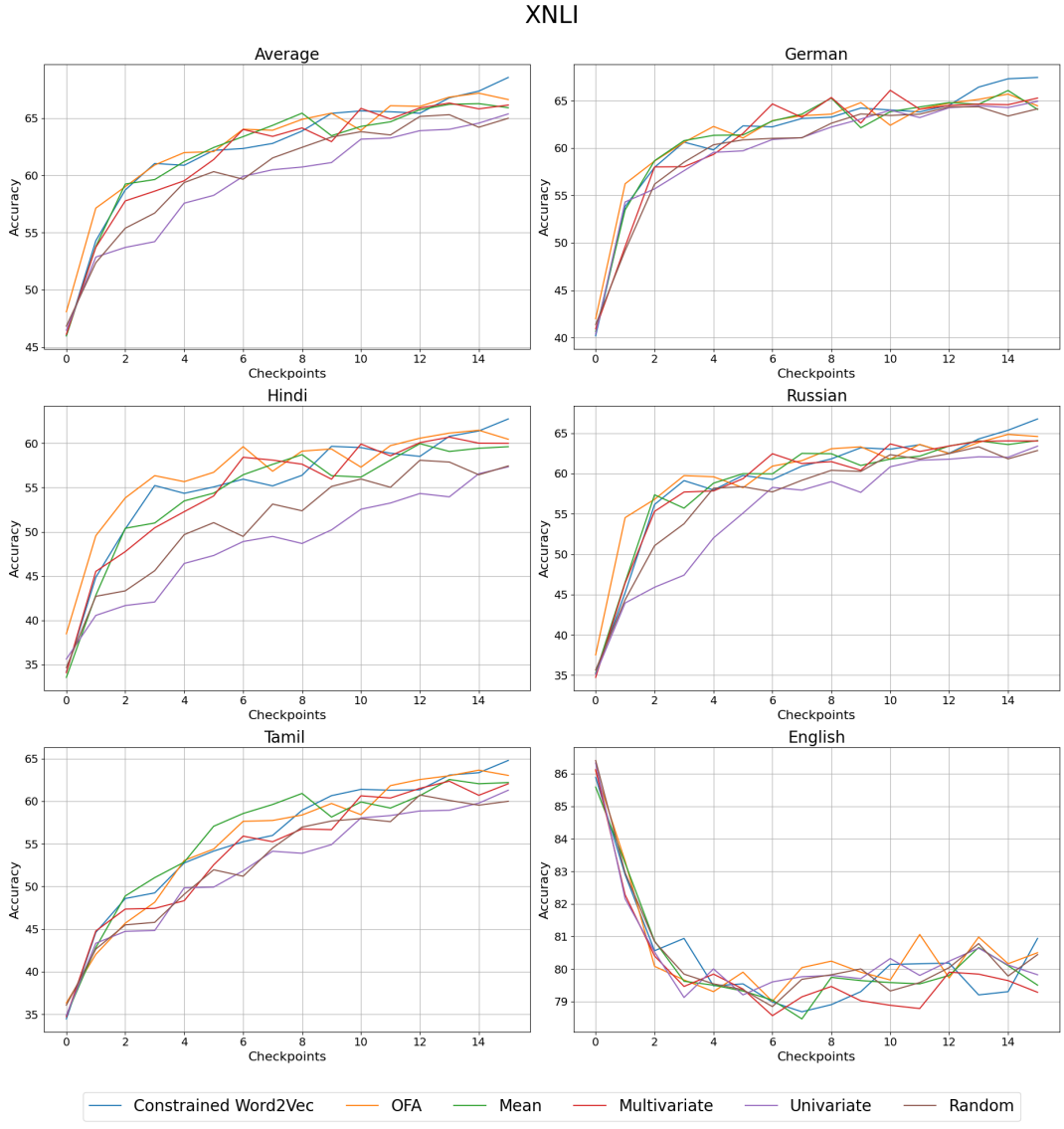

Figure 6: XNLI evaluation of expanded RoBERTa models

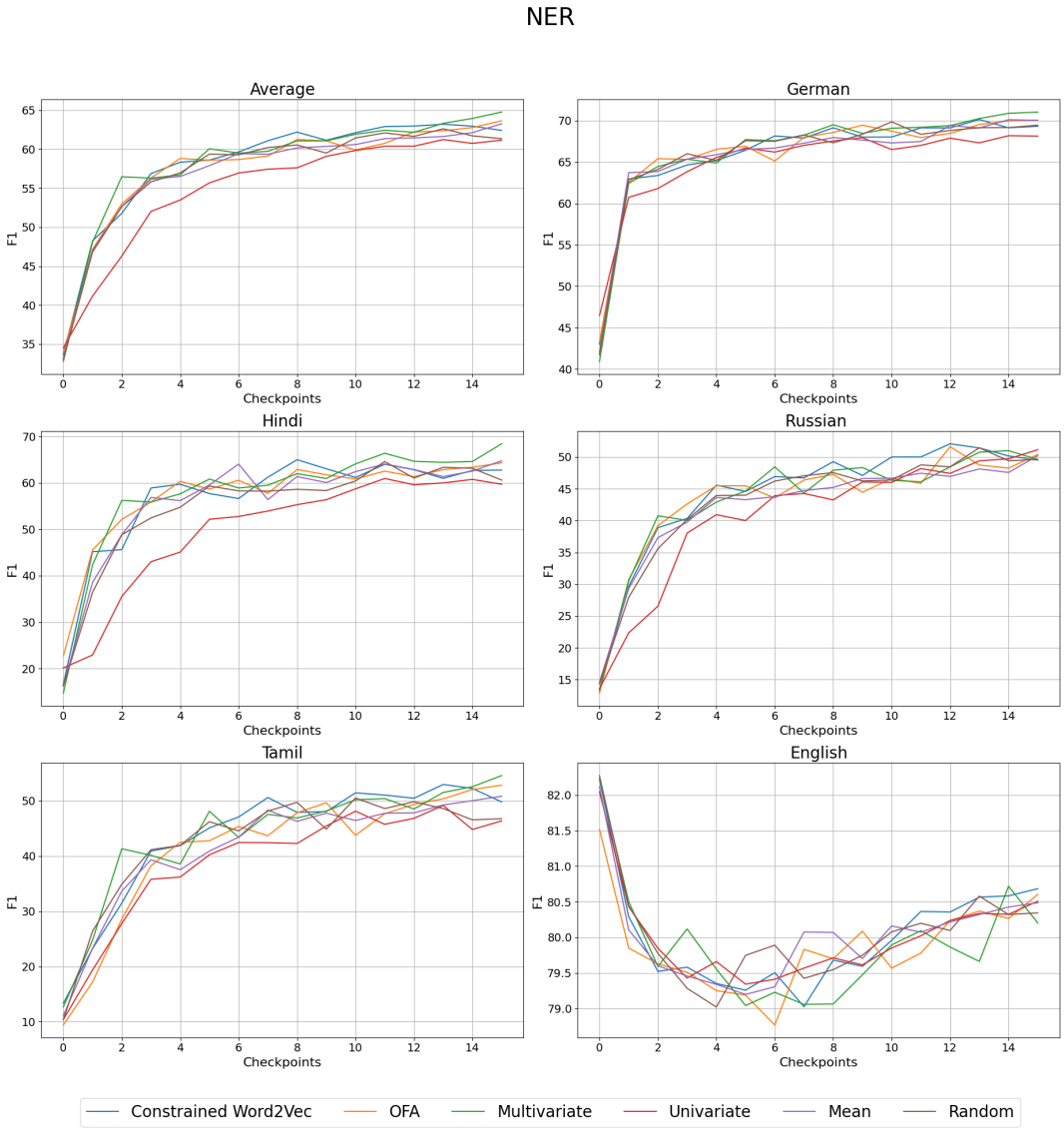

Figure 7: NER evaluation of expanded RoBERTa models

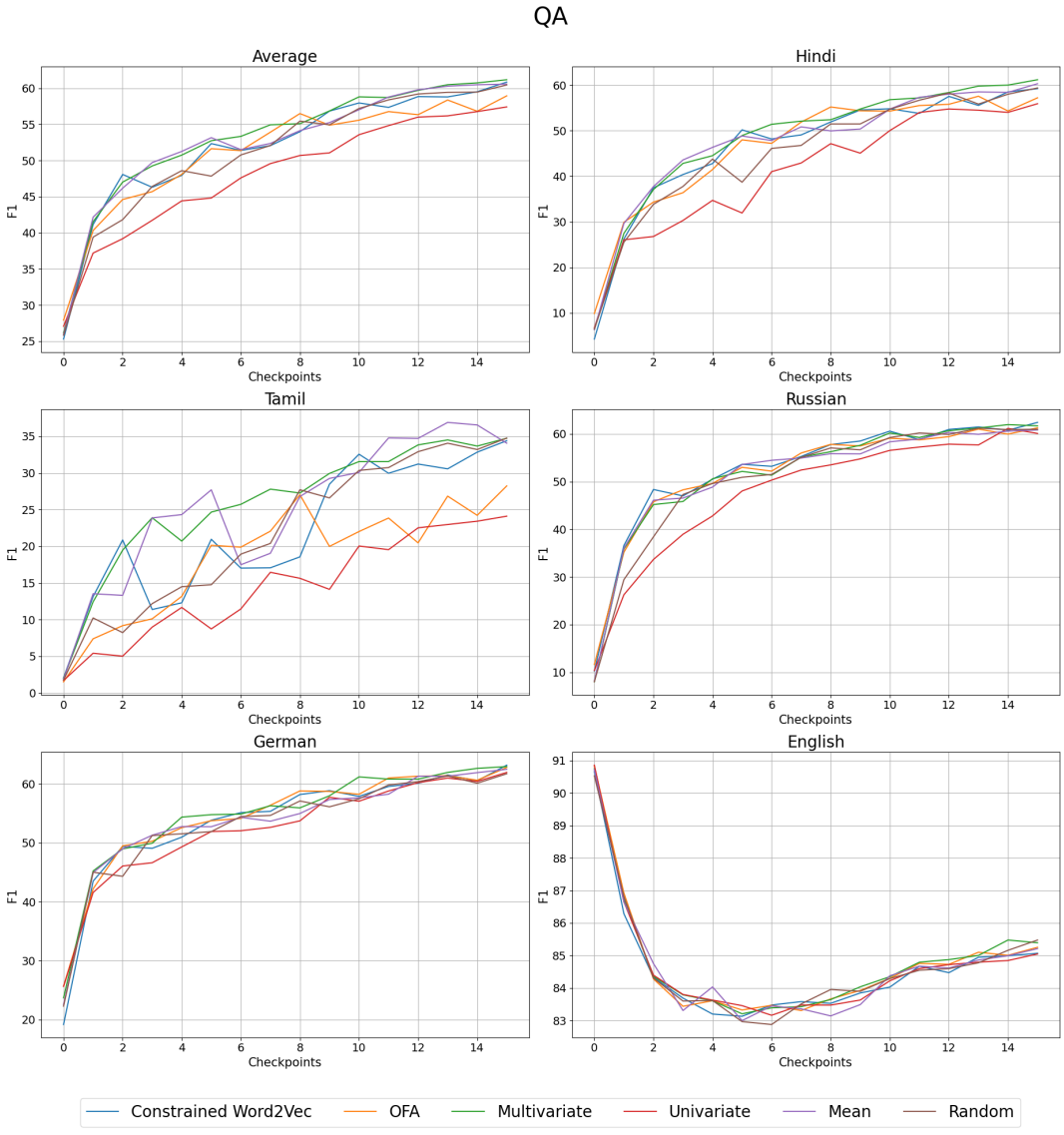

Figure 8: QA evaluation of expanded RoBERTa models

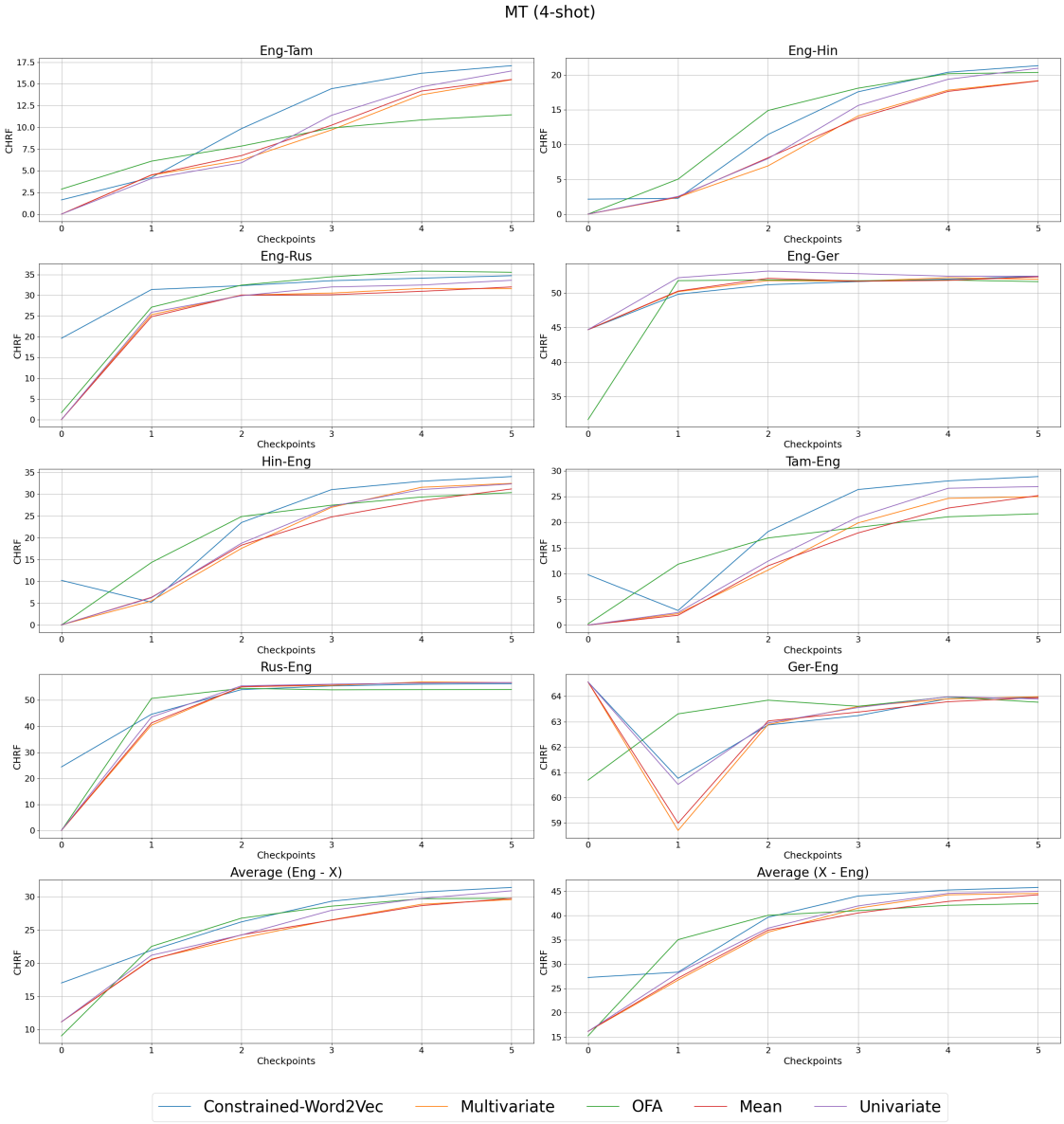

Figure 9: MT 4-shot evaluation of expanded LLaMA2 models

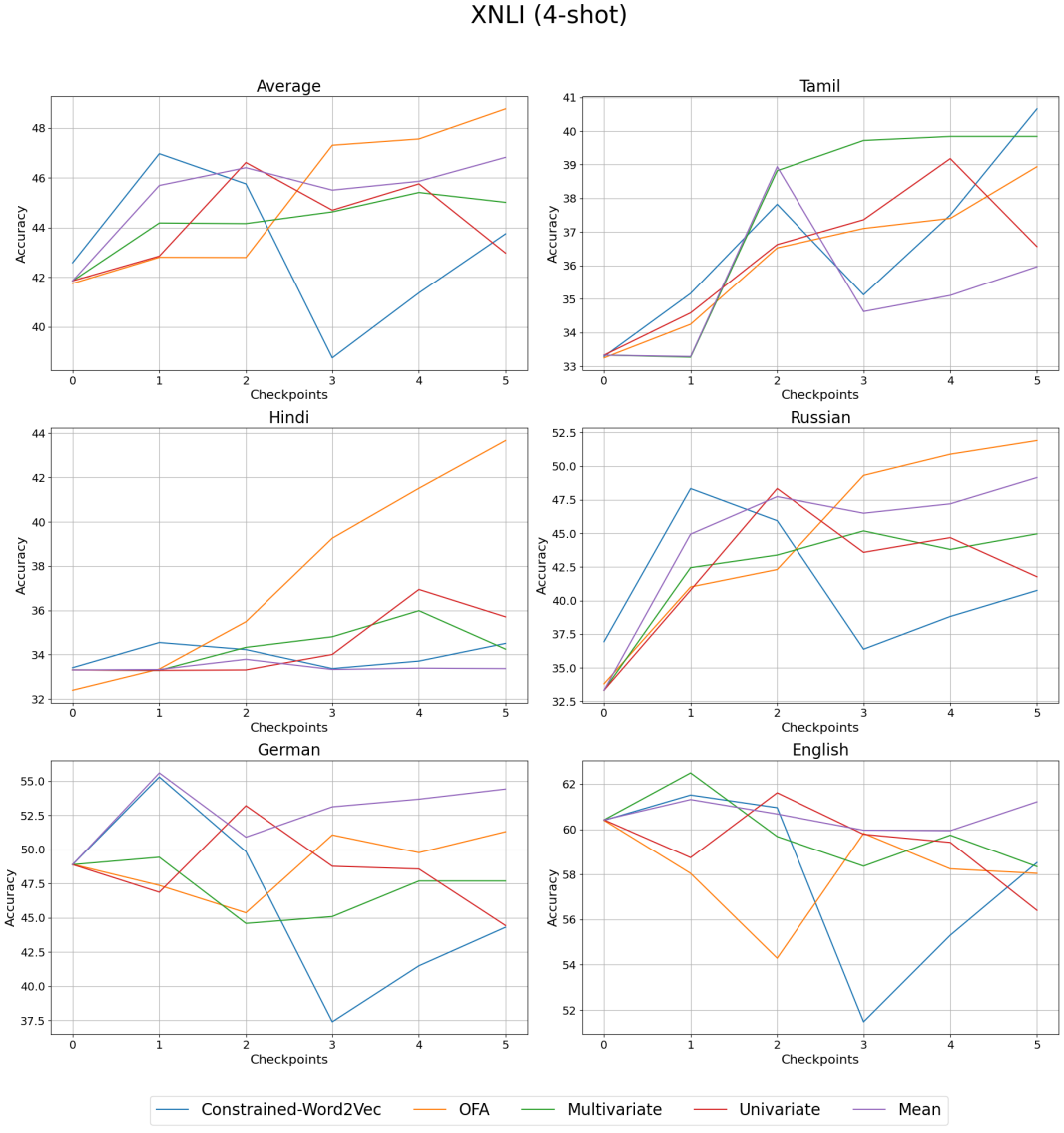

Figure 10: XNLI 4-shot evaluation of expanded LLaMA2 models

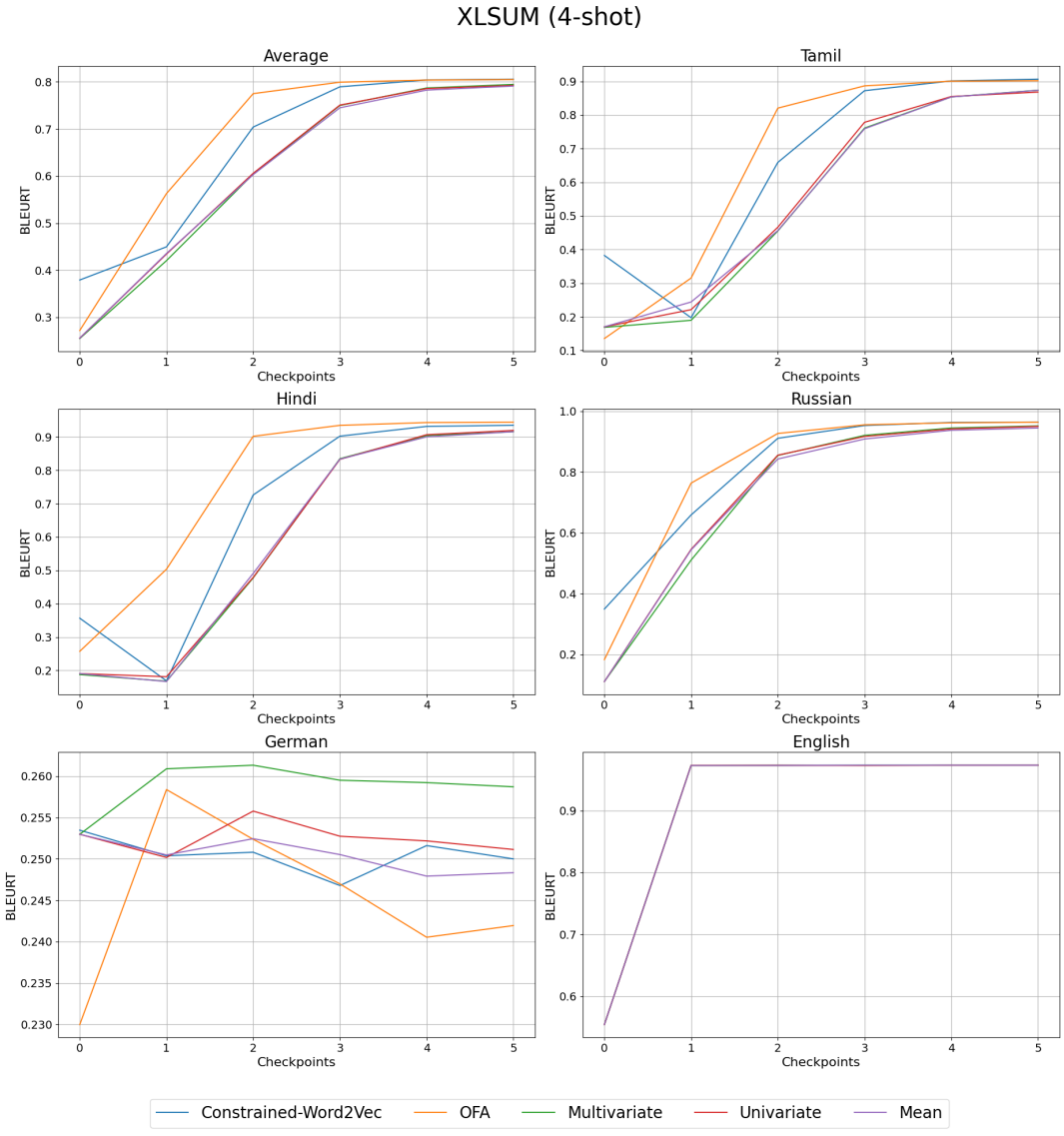

Figure 11: XLSUM 4-shot evaluation of expanded LLaMA2 models

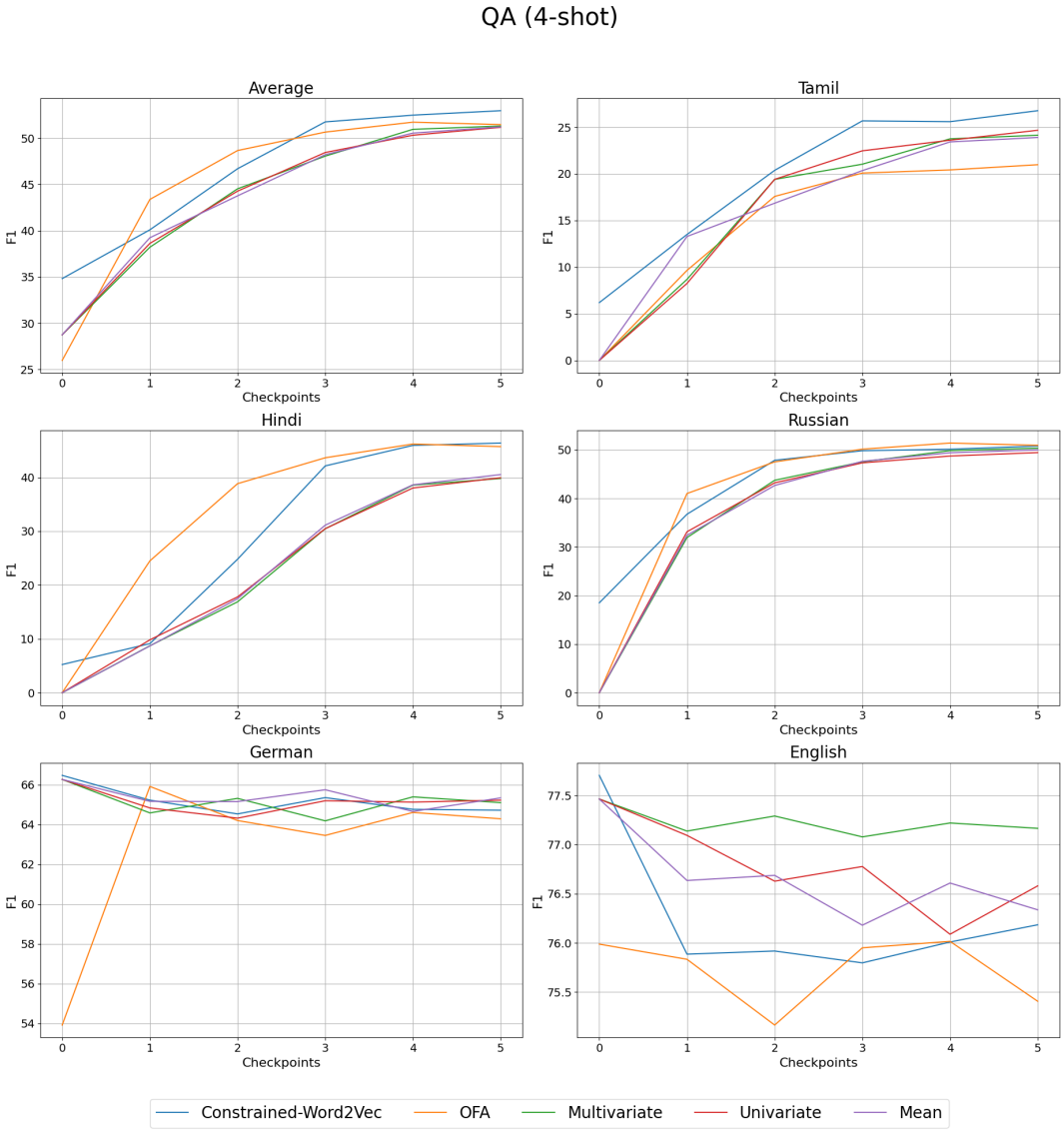

Figure 12: QA 4-shot evaluation of expanded LLaMA2 models