École Polytechnique Fédérale de Lausanne

Improving multilingual zero-shot learning for vision-and-language models with minimum resources.

by Maxence Jouve

Master Thesis

Approved by the Examining Committee:

Prof. Nigel Collier, Prof. Karl Aberer
Thesis Advisor

Fangyu Liu
External Expert

Fangyu Liu, Rémi Lebret
Thesis Supervisor

EPFL IC IINFCOM HEXHIVE
BC 160 (Bâtiment BC)
Station 14
CH-1015 Lausanne

August 18, 2022

Words do not express thoughts very well. they always become a little different immediately they are expressed, a little distorted, a little foolish. And yet it also pleases me and seems right that what is of value and wisdom to one man seems nonsense to another.
— Hermann Hesse, Siddhartha

Acknowledgments

This research would not have been possible without the help of Fangyu Liu and Dr Rémi Lebret. They gave me weekly feedback and valuable assistance throughout the course of my thesis. I would also like to thank Professor Nigel Collier and everyone at the Language Technology Lab in Cambridge for welcoming me when I arrived in the UK, as well as for their input and help. In addition, I am thankful to the Theoretical and Applied Linguistics office personnel who made it feasible for me to visit Cambridge and made my stay enjoyable.

Finally, I am grateful for all the people who accompanied me during my Master's at EPFL. It represents the conclusion of a significant stage in my life and would not have been nearly as joyful without my family and friends.

Lausanne, August 18, 2022 Maxence Jouve

Abstract

Large-scale multimodal models such as LXMERT, UNITER, or ViLBERT have excelled at a variety of visio-linguistics tasks, including visual question answering, image-text retrieval, and visual commonsense reasoning. More recently, thanks to new pretraining objectives, those models were enhanced to handle multilingual text input resulting in innovative work such as M^3P or UC^2. However, the recent advent of multilingual visio-linguistic benchmarks such as IGLUE, has demonstrated that the performances in English do not transfer well to other languages in a zero-shot setting. Not only do visio-linguistic models have to deal with different languages, but they also have to interact with multicultural concepts (both visual and texual) that are often missing from Western-centered visual datasets. In this thesis, we will use the MaRVL dataset (part of the IGLUE benchmark) as a test bed for improving the performance of visio-linguistic models using minimum resources. In particular, we will focus on mUNITER and xUNITER, which are two multilingual variants of UNITER. First, we will investigate the benefits of using the recent large-scale WIT dataset. It is a massive multilingual multimodal dataset that has not been used for pretraining by previous state-of-the-art models. Consequently, we will use this new dataset to improve the pretraining phase of mUNITER and xUNITER. Second, we will study the advantages of utilising the code-switching framework for training on reasoning tasks like MaRVL. As a result, we will highlight how it is possible to obtain an 3.8% and 2.2% average accuracy boost (across languages) for mUNITER and xUNITER, respectively, using minimal resources both in terms of accessability (open-source and ready-use) and computing power required. Our experiments will show how visio-linguistic models can extract further knowledge from multilingual resources.

Contents

Chapter 1

Introduction

1.1 Motivations

The fields of computer vision and natural language processing have witnessed significant advancements in the previous decade thanks to Deep Learning techniques. On the one hand, convolutional neural networks (CNNs) (Krizhevsky et al., 2012) combined with large labelled datasets such as ImageNet (Russakovsky et al., 2014) have played a significant role in a number of vision tasks, including classification, object detection, and segmentation. On the other hand, the transformer architecture (Vaswani et al., 2017), which uses a self-attention mechanism, has become the backbone of many successful language models that can perform tasks such as translation, natural language understanding, and text-generation, in either a transfer learning or zero-shot/few-shot learning fashion. The transformer design was also successfully implemented for image recognition, demonstrating comparable performance to CNNs but requiring fewer computational resources (Dosovitskiy et al., 2020).

Among the numerous Transformer language models, BERT (Devlin et al., 2018) architecture is the most widely used, and is shown to be extremely effective on a variety of NLP downstream tasks (evaluated using the GLUE benchmark (Wang et al., 2018), for example). In addition, BERT can be made into a multilingual language model (multilingual BERT) by incorporating multilingual Wikipedia data in the pre-training phase. It led to the possibility of zero-shot cross-language transfer (Pires et al., 2019), (Wu and Dredze, 2019b). A model that has been finetuned for a task in a high-resource language can be utilised to do the same task in another language, i.e., without using a single example from the target language. With the emergence of even larger language models, zero or few-shot transfers have become increasingly prevalent. GPT-3 (Brown et al., 2020), an autoregressive language model with 175B parameters, is a major milestone. Without fine-tuning, the model is able to generate outcomes comparable to state-of-the-art finetuned models. By communicating with the model via a text prompt, the task may be directly specified.

However, an AI agent's ultimate goal should be to interact with a multimodal world. The initial multimodal models were visio-linguistic since vision and language are two of the most prevalent modalities and their respective fields have achieved substantial advancements. Multimodal tasks involving vision and language can be grouped into three distinct categories: generation, classification, or retrieval (Uppal et al., 2022). In this study, we will concentrate on the Visual

Commonsense Reasoning (VCR) task, in which a model must infer understanding and expertise from a pair of visual inputs and textual assertions. It is possible for the output to be either a string or a boolean.

Earlier multimodal models handle each modality separately, resulting in task-specific architectures where representations from both modalities were fused using techniques such as Multimodal Compact Bilinear Pooling (MCB) (Fukui et al., 2016). In the meantime, following successes in NLP with machine translation and in computer vision with object detection, attention mechanisms were at the core of a new wave of visio-linguistics models such as Bilinear Attention Networks (Kim et al., 2018). Furthermore, the multimodal models were enhanced with the emergence of the transformer architecture, and especially BERT. Many visio-linguistic models employ the BERT framework, such as, VisualBERT (Li et al., 2019), ViLBERT (Lu et al., 2019), ImageBERT (Qi et al., 2020), or UNITER (Chen et al., 2019). After finetuning, these models often perform well with English and other languages, given sufficient resources, on a wide variety of downstream applications (e.g., image captioning, text-to-image synthesis, image or text retrieval from other modality, etc.). Nonetheless, very large-scale visio-linguistic models like as Flamingo (Alayrac et al., 2022) are emerging (80B parameters). This model can achieve new state-of-the-art performance on numerous downstream tasks with few-shot learning by simply interacting with the model via visio-linguistic prompts.

1.2 Main challenges

Despite the success of these pre-trained multimodal models, we identify two challenges that we are still facing when using them:

- **Multilinguality.**

  These multimodal models function well with resource-rich languages like English, but it is important to emphasise that the linguistic modality goes beyond English. In fact, there is still a significant performance gap between high-resource languages and low-resource languages for these models (Liu et al., 2021). One of the reasons is that the majority of multimodal datasets are currently only available in English and a few other languages with abundant resources such as Chinese, French, and German. Another reason is that multimodal datasets have more linguistic imbalances than text-only datasets. Indeed, the vast majority of vision datasets are collected from Western sources (Shankar et al., 2017). Consequently, multimodal datasets exclude concepts from other cultures and are unable to accurately represent a diverse and heterogeneous world.

- **Zero-shot learning.**

  Similar to how humans learn, it is desirable for a model to be able to adapt with zero or very few examples. While there has been a rich literature of studies on the zero/few-shot learning of English monomodal (text-based) models (e.g., BERT and GPT-3 have already demonstrated their success in tackling many NLU tasks with zero/few-shots or prompting techniques), how we can adapt multilingual multimodal models with zero or

few examples remains largely unexplored. It is important to note that zero-shot learning is a very significant application scenario for multilingual multimodal tasks, as labelled datasets in this field are extremely expensive and difficult to acquire. Also, the combined multimodality and multilinguality components will present new obstacles for the zero-shot learning environment, as out-of-distribution issues are anticipated to become more frequent and severe. This study focuses on multilingual zero-shot learning for vision-and-language models in an effort to address the aforementioned difficulties. We intend to use strategies to finetune pretrained multilingual multimodal models and to specialise them for downstream tasks that require no extra labels.

1.3 Thesis statement

The objective of this thesis is to enhance the zero-shot performance of multilingual multimodal models using minimal resources. We will rely on open-source, ready-to-use data, such as the WIT dataset (Srinivasan et al., 2021) or bilingual dictionaries from Panlex[1], and utilise as minimal computing power as feasible.

1.4 Contributions

To alleviate those challenges, we will attempt to improve the performances of top-performing multilingual multimodal models on the MaRVL benchmark (Liu et al., 2021). This dataset expands the NLVR2 challenge (Suhr et al., 2018) (visual commonsense reasoning task) to five languages: Indonesian, Mandarin Chinese, Swahili, Tamil, and Turkish. This selection covers different language families in terms of typography, genealogy, and geography. It also includes both high-resource and low-resource languages. Given a pair of photographs and a grounded statement about the pair, the goal is to predict whether or not the statement is correct. It is especially difficult because the statements are in several languages and the concepts covered are multicultural. Indeed, the images were chosen and annotated by a native speaker. As a result, top-performing multilingual multimodal models such as M^{3}P (Ni et al., 2020), UC^2 (Zhou et al., 2021), mUNITER, and xUNITER (Liu et al., 2021) show relatively poor zero-shot (after fine-tuning on NLVR2) scores on the MaRVL dataset (respectively 56.28% , 57.28% , 53.72% , and 54.59% accuracy). More importantly, the performances of mUNITER, xUNITER, and M^{3}P are closer to the random frontier (50%) than their performance on the translated version of MaRVL (in English). It indicates that the gap between cross-lingual transfer and English performances is still huge. Those models rely on Conceptual Captions (Sharma et al., 2018) as a pretraining dataset. As Conceptual Captions is a monolingual (English) dataset, its captions may have a Western bias and multicultural concepts may be lacking. To make it multilingual, M^{3}P and UC^2 opt for different strategies. While M^{3}P uses code-switching to better align word representation across languages (top 50 languages on Panlex), UC^2 uses an augmented version of the dataset that includes captions machine-translated into five languages. Neither mUNITER nor xUNITER,

[1] https://panlex.org/

in contrast, utilised a modified version of Conceptual Captions. The models alternate between a monolingual multimodal objective on Conceptual Captions and a monolingual multilingual masked language modelling training using a multilingual Wikipedia dump. This thesis will focus on mUNITER and xUNITER.

The initial goal is to investigate the advantages of further pretraining the models using a multilingual multimodal dataset: Wikipedia Image Text (WIT) (Srinivasan et al., 2021). It is essential to investigate the advantages of utilising large-scale multilingual multimodal datasets encompassing many more languages (108), and cultural concepts. Integrating multicultural data from WIT could be a crucial step towards minimising the distribution shift observed by visio-linguistic models on the MaRVL benchmark. In addition, it is cost-effective for research to rely on such datasets because they do not require expensive approaches such as machine translation to expand language coverage of existing monolingual multimodal datasets. We construct very modest training datasets ($18,500$ entries) and further specialise mUNITER for each language (resulting in one model per language). Using this strategy, we will be able to increase accuracy across languages by 1.3%. On top of that, we will conduct a detailed analysis of the factors that influence zero-shot cross-lingual transfer on MaRVL. First, we discovered that the quality of target language multimodal data is not crucial for better performance. Then, we demonstrate that the amount of English finetuning data can be decreased to prevent overfitting and thus increase accuracy on the MaRVL task. Finally, we find that covering MaRVL (test set) concepts during pretraining does not result in higher accuracy on the downstream task. It will be discussed in greater detail in chapter 5.

The second objective is to apply a code-switching training technique to mUNITER and xUNITER and investigate its advantages on a visual commonsense reasoning challenge, such as MaRVL. M^{3}P utilised code-switching to broaden the multimodal capabilities to a multilingual setting. Ni et al. (2020) showed the benefits of code-switching for pretraining and finetuning (for an image-text retrieval task). As it utilises bilingual dictionaries, it is, in fact, a low-resource method for matching cross-linguistic representations of the same item. Given the prevalence of bilingual dictionaries, exploiting their data could be a crucial step in expanding multimodal models to multilingual contexts in the future. As a result, we will introduce a new code-switching framework. On the one hand, it uses information from part-of-speech tags to avoid noisy translations. On the other hand, it does translation from target languages to English in order to better match representations in both directions. Using this innovative code-switching method, we will increase the accuracy of mUNITER and xUNITER by 3.8% and 2.2%, respectively, beating both M^{3}P and UC^2 on MaRVL. Additionally, we conduct an extensive analysis of code-switching. First, we will code-switch the MaRVL test sets to determine the knowledge models can gain by having access to English concept representations. Then, we will demonstrate which part-of-speech tags produce the greatest gains. Finally, by examining word embeddings, we demonstrate that code-switching helps align representations of the same concept across languages. This topic will be covered in chapter 6.

Summary of contributions.

(1) Examining the benefits of pretraining visio-linguistics models with the large-scale WIT dataset, as well as studying a number of important aspects of multilingual multimodal pretraining.

(2) Comprehensive analysis of code-switching and cross-lingual gaps between English and other languages that exist in current vision-language models. It will improve the performances of mUNITER and xUNITER on the MaRVL benchmark by 3.8% and 2.2% assuming the availability of just bilingual dictionaries.

Chapter 2

Background

2.1 Multimodal (visio-linguistic) models

The initial visio-linguistic models addressed each modality independently before combining the two representations using diverse strategies. It led to architecture tailored for various tasks. Using standalone vision and language encoders, they first model visual and textual inputs. On the one hand, traditional vision models, such as FasterRCNN (Ren et al., 2015) or YOLO (Redmon et al., 2015), were used to process visual inputs and extract crucial semantic information, such as the presence of objects and their bounding boxes. On the other hand, context-free language models (word embeddings), such as word2vec (Mikolov et al., 2013), or contextual models (capable of capturing relationships inside a sentence), such as LSTM (Hochreiter and Schmidhuber, 1997) or GRU (Cho et al., 2014), were utilised to extract text embeddings. The representations from both modalities are then fused using techniques such as Multimodal Compact Bilinear Pooling (MCB) (Fukui et al., 2016). When it was released, this model obtained state-of-the-art performance on the VQA challenge (Agrawal et al., 2015).

Visio-linguistic models were then enhanced to include attention mechanisms. It enables the models to pay greater "attention" to specific visual and linguistic inputs, resulting in a more accurate depiction of images and sentences. Using Bilinear Attention Networks, Kim et al. (2018) achieved state-of-the-art performance on the VQA challenge. Following attention mechanisms, the transformer architecture was also used by visio-linguistic models. LXMERT (Tan and Bansal, 2019) relies on three encoders: one to model objects' relationships; one for language; and the last one to model cross-modality.

The BERT framework, which expands the transformer architecture, is also at the foundation of numerous large-scale visio-linguistic models like VisualBERT (Li et al., 2019), ViLBERT (Lu et al., 2019), ImageBERT (Qi et al., 2020), or UNITER (Chen et al., 2019). Thus, they can simultaneously model text and visual inputs. These models can be categorised as dual-stream (LXMERT and ViLBERT) or single-stream (VisualBERT, ImageBERT, and UNITER), although Bugliarello et al. (2020) demonstrated that they can be unified under a single framework and had comparable performance when trained using the same procedure. Following some finetuning on a specific task, those large-scale visio-linguistic models can perform well on a wide variety of tasks in English or other high-resource languages. Aside from Visual Commonsense Reasoning (NLVR2

and MaRVL), other popular and related multimodal tasks involving vision and language are Visual Question Answering (given a visual input and a question, a model must produce a textual response), Visual Captioning (a model must describe a visual input), and Visual Generation (given a textual input, a model must produce a visual output).

2.2 Zero-shot cross-lingual transfer

It refers to a model that was trained and selected (for a specific task) in a source language (usually one with abundant resources) and whose knowledge is transferable to another target language without additional training. For a given task, it is especially advantageous to transfer information acquired in a language with abundant resources to one with limited resources. Thanks to cross-lingual word embedding spaces, the zero-shot transfer scheme was made feasible. Ruder et al. (2019) surveyed various techniques used to obtain multilingual word embedding models. They demonstrated that the majority of these models are often trained for the same objective. However, they differ with regard to the data employed, the source monolingual corpus, and various hyperparameters. These multilingual word embedding models form the foundation of cross-lingual transfer for various tasks. At first, the architectures were task-specific. For instance, Xie et al. (2018) leveraged those embeddings for Name Entity Recognition (NER) and achieved competitive performance across languages without relying on a large amount of target language resources.

The introduction of BERT was a turning point for zero-shot cross-lingual transfer. Indeed, Wu and Dredze (2019a) showed that mBERT, which was trained on 104 languages, learns a cross-lingual space to represent words. Due to its large-scale architecture, the model can be fine-tuned for a variety of tasks, including document classification, natural language inference, NER, part-of-speech tagging, and dependency parsing. They demonstrated that mBERT can obtain strong or even state-of-the-art performance on these five tasks without supervision in the target language.

Chapter 3

Datasets

3.1 Pretraining datasets

3.1.1 Conceptual Captions

The Conceptual Captions (Sharma et al., 2018) dataset played a key part in training recent large-scale visual-linguistic models, including ImageBERT, ViLBERT, UNITER, M^3P, UC^2, and $CCLM^3$. When it was introduced, it was an order of magnitude bigger than the previous largest dataset, COCO. Conceptuals Captions comprises 3.3M (image + caption) pairs, which span a wide range of topics. On top of that, it gathers high-quality captions. Sharma et al. considered a set of 5B images and their alt-texts from 1B English websites. Each image-text pair was processeded using a two-step pipeline consisting of a filter (quality of image, content, and determining whether caption tokens and detected items in the image match), followed by a transformation (a mapping between name entities and common nouns using a knowledge graph; date, duration, and location are removed, etc.).

3.1.2 Plaintext Wikipedia Dump

The Plaintext Wikipedia Dump (Rosa, 2018) contains the raw text from articles in 297 distinct languages.

3.1.3 Wikipedia Image Text Dataset (WIT)

WIT (Srinivasan et al., 2021) is the first massive multilingual multimodal dataset. It contains 37.6M image-text pairs (11.5M unique images) in 108 languages. As a result, it encompasses a broad range of concepts from diverse cultures, making it more inclusive than previous datasets compiled primarily from Western sources. WIT could become a viable pretraining dataset for future large-scale multilingual multimodal models. It was assembled from Wikipedia articles, leveraging the quality checks already implemented by the platform. In addition, qualitative entries were selected using further filtering (image size, some generic images such as flags or maps, questionable content, etc.). Each WIT entry includes an image and several text fields (some may be missing):

Language	# (image, text) pairs
ID	132275
SW	19695
TA	42950
TR	114734
ZH	125060

Table 3.1: The number of entries per language remaining from WIT after filtering.

- The associated Wikipedia article title.
- The Wikipedia article main text (introduction).
- The corresponding section text (if the image is specific to a section).
- The reference description (in various languages, directly describing the image).
- The attribution description (the description of the image in Wikimedia, which is common to all occurrences of the image).
- The alt-text description.

This study will concentrate on the languages included in the MaRVL dataset: Indonesian (ID), Swahili (SW), Tamil (TA), Turkish (TR), and Mandarin Chinese (ZH). We will also only retain entries with a reference description (closest to a caption) and discard all images in GIF or SVG format. Table 3.1 summarises the number of remaining entries per language after filtering. We took a random sample of one-third of all Mandarin Chinese entries for storage-related reasons.

3.2 Finetuning dataset

3.2.1 NLVR2

NLVR2 (Natural Language Visual Reasoning for Real) (Suhr et al., 2018) is a dataset for joint reasoning about language and images. The objective of this challenge is to determine whether a given assertion regarding a pair of images is true. The version we are utilising for this project contains 86373 entries, covering a large spectrum of concepts. It was built to be semantically dense. NLVR2 is challenging since a model must simultaneously reason about a statement and two images. It entails dealing with quantities, object properties, comparisons, and spatial relations. To create the dataset, Suhr et al. asked workers to provide two "true" statements and two "false" ones given four pairs of images. As a result, one statement yields four unique entries in the dataset, ensuring that the model cannot predict the answer based just on the sentence and resulting in a balanced dataset. Figure 3.1 depicts an instance from NLVR2.

Figure 3.1: Example of a NLVR2 entry. Sentence: *"Each image shows a full round pizza."*, Label: *TRUE*.

3.3 Test dataset

3.3.1 MaRVL

MaRVL (Multicultural Reasoning over Vision and Language) (Liu et al., 2021) and NLVR2 share the same objective. However, to accommodate global and multicultural themes, Liu et al. developed a new methodology for constructing an ImageNet-like hierarchy. First, they chose diverse languages, including Indonesian, Mandarin Chinese, Swahili, Tamil, and Turkish. These languages belong to distinct families (Austron, Sino-Tibetan, Niger-Congo, Dravidian, and Turkic, respectively). Consequently, they encompass different language typologies (differences in grammatical structure), genealogies (historical criteria) and areas (geographical criteria). Moreover, their alphabets differ (Latin, Tamil script, and Hanzi), and they are both low-resource and high-resource languages. Then, they requested that native speakers (of each language) select concepts and images. This stage is essential because, unlike in NLVR2, where concepts are automatically scrapped, it permits a multicultural and diverse perspective. Indeed, the majority of datasets rely on English resources, resulting in Western-centric views of the world. Overall, 21.1% of the selected concepts are absent from ImageNet. Lastly, they asked workers to annotate the images similarly to NLVR2, resulting in four entries per statement. Overall, the diversity of languages and multiculturalism of concepts make MaRVL extremely challenging for visio-linguistic models in a zero-shot cross-lingual transfer setting. Figure 3.2 shows a MaRVL entry in Swahili. Table 3.2 displays the number of entries and unique concepts for each language.

Figure 3.2: Example of a MaRVL entry. Sentence: "Picha moja inaonyesha mtu akipiga filimbi na picha nyingine inaonyesha filimbi zimewekwa chini", Machine-translated sentence: "One picture shows a person blowing a whistle and another picture shows the whistles set down.", Label: *FALSE*.

Language	**# entries**	**# concepts**
ID	1128	95
SW	1108	78
TA	1242	83
TR	1180	79
ZH	1012	94

Table 3.2: Number of unique concepts and total entries for each language in MaRVL.

Chapter 4

Models

4.1 UNITER

4.1.1 Architecture

UNITER (UNiversal Image-TExt Representation) (Chen et al., 2019) is a large-scale visio-linguistic model. It can serve as a backbone architecture for a variety of multimodal tasks, such as visual question answering or visual commonsense reasoning. Two separate embedders are utilised to process an image-text pair input. On the one hand, visual features are retrieved from pictures using an image embedder for each of their regions. Additionally, the positions of each region are encoded. Then, visual features and locations are sent to a fully connected layer, summed up and normalised using a normalisation layer. The text embedder, on the other hand, employs BERT to turn each token into WordPieces. The representation of each token is then obtained by adding its sub-word embeddings to its position embeddings, followed by a normalisation layer. After obtaining the picture and text representations, UNITER fuses the representations using a multi-layer transformer and learns a cross-modal embedding for each input. Figure 4.1 displays the UNITER architecture.

4.1.2 Pretraining objectives

Chen et al. (2019) relied on multimodal training objectives. We denote $\mathbf{v} = v_1, ..., v_V$, the set of visual features, $\mathbf{w} = w_1, ..., w_W$, the set of words, θ the model's parameters, D the training set.

- **Mask Language Modeling (MLM)**. The mask indices is called $\mathbf{m}$ and the corresponding masked words are denoted $\mathbf{w_m}$, non-masked token $\mathbf{w_{\setminus m}}$. Random words are masked with a probability of 0.15, and replaced by a $[\mathbf{MASK}]$ token. The goal is to predict the masked tokens $\mathbf{w_m}$ given the non-masked ones and the visual features $\mathbf{v}$, minimizing the following negative log-likelihood:

$$L_{MLM}(\theta) = E_{(\mathbf{w},\mathbf{v})\sim D} \log P_\theta(\mathbf{w_m} \mid \mathbf{w_{\setminus m}}, \mathbf{v}) \tag{4.1}$$

- **Image-Text Matching (ITM)**. A $[\mathbf{CLS}]$ representing the combined representation of an

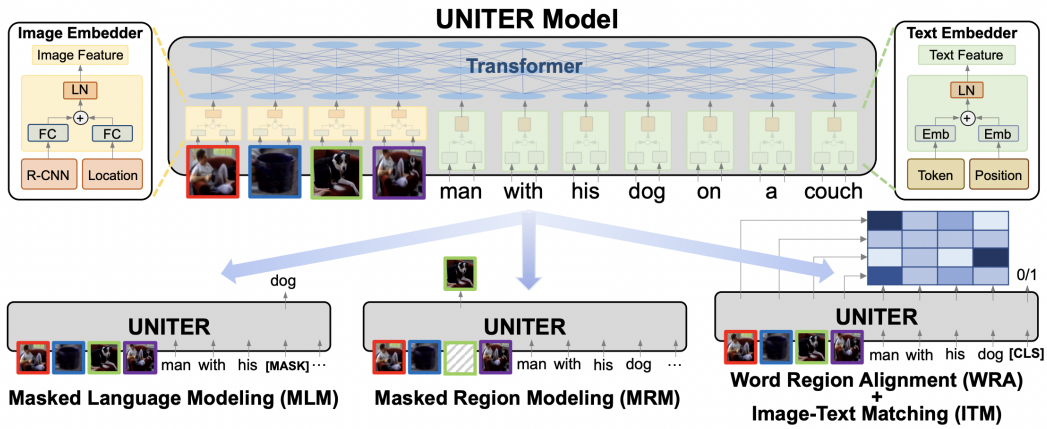

Figure 4.1: UNITER model architecture & training objectives. The picture was taken from the UNITER paper (Chen et al., 2019).

image-text pair is used for this task. The model aims at predicting whether a sentence and a set of image regions correspond to a real pair in the dataset. A sigmoid $s_\theta(\mathbf{w}, \mathbf{v})$ function is used to score a pair between 0 and 1. The following function, a binary cross-entropy loss, is utilised as objective:

$$L_{ITM}(\theta) = -E_{(\mathbf{w},\mathbf{v})\sim D}[y \log s_\theta(\mathbf{w}, \mathbf{v}) + (1-y) \log (1 - s_\theta(\mathbf{w}, \mathbf{v}))] \tag{4.2}$$

- **Word-Region Alignment (WRA)**. It relies on an Optimal Transport (Peyré and Cuturi, 2018), (Chen et al., 2020). This optimal transport is used to learn an alignment between $\mathbf{w}$ and $\mathbf{v}$ thanks to a transport plan $T \in R^{W \times V}$. We can then obtain the following alignment loss between the words and image regions, using $c(x_i, y_i)$ as our distance function, in this case, the cosine distance[1].

$$L_{WRA}(\theta) = \min_{\mathbf{T}} \sum_{i=1}^{W} \sum_{j=1}^{V} \mathbf{T}_{ij}.d(w_i, v_j) \tag{4.3}$$

- **Masked Region Modeling (MRM)**. It is the equivalent of the MLM objective for visual inputs. In this case, some visual regions are masked (replaced by 0), denoted as $\mathbf{v_m}$, and the model should reconstruct them using the non-masked vision regions $\mathbf{v}_{\backslash \mathbf{m}}$ and the textual features $\mathbf{w}$. The objective is as follow:

$$L_{MRM}(\theta) = E_{(\mathbf{w},\mathbf{v})\sim D} f_\theta(\mathbf{v_m} \mid \mathbf{v}_{\backslash \mathbf{m}}, \mathbf{w}) \tag{4.4}$$

[1] $d(x, y) = 1 - \frac{x.y}{\|x\|^2.\|y\|^2}$

4.2 Multilingual UNITER: mUNITER & xUNITER

4.2.1 Architecture

Liu et al. (2021) introduced the mUNITER and xUNITER models. They adhere to the same architectural design as UNITER. The original model, however, is initialised with mBERT (Devlin et al., 2018) for mUNITER and XLM-R$_{BASE}$ (Conneau et al., 2019) for xUNITER.

The visual input consists of 36 visual features extracted by Faster-RCNN (Ren et al., 2015) with a ResNet-101 backbone (He et al., 2015). On top of that, an special token [**IMG**] representing the whole image is added. A visual input is thus $\mathbf{v} = [[\mathbf{IMG}], v_1, ..., v_{36}]$. Besides, the textual input is enclosed by two special tokens, [CLS] and [SEP]. The first one represents the whole sequence and the other one marks the end of it, $\mathbf{w} = [[\mathbf{CLS}], w_1, ..., w_W, [\mathbf{SEP}]]$. Finally, to yield an embedding for an image-text pair, the two tokens representing the visual input ([**IMG**]) and the textual input ([**IMG**]) are element-wise multiplied.

4.2.2 Pretraining objectives

For pretraining Liu et al. (2021) alternate between a multilingual monomodal phase and multimodal monolingual (in English) phase. The parameters are updated after each phase. The training procedure follows a controlled setup developed by Bugliarello et al. (2020) to compare the performance of various visio-linguistic models.

- **Multilingual monomodal phase**. The model is trained using the same MLM procedure as in BERT (Devlin et al., 2018). It uses the Wikipedia Dump (Rosa, 2018) data in 104 languages. To achieve a balanced training across languages, a multinomial distribution sample is employed. For mUNITER, the parameter of the distribution is $\alpha = 0.70$ and for xUNITER, $\alpha = 0.30$.

- **Monolingual multimodal phase**. The model is trained using three objectives mentioned for UNITER. MLM (equation 4.1), ITM (equation 4.2) and MRM (equation 4.4). For MRM, KL-divergence is used. It means that labels from the object detectors are used as supervision signals for the masked regions (for a region v_i, the distribution of object classes is $c(v_i)$). In this case, the f_θ function introduced in equation 4.4 is $f_\theta(\mathbf{v_m} \mid \mathbf{v}_{\setminus \mathbf{m}}, \mathbf{w}) = \sum_{i=1}^{M} D_{KL}[(c(v_m^{(i)})||g_\theta(v_m^{(i)})]$ with M the number of masked regions, and g_θ a prediction of the V classes (one per region) from the masked region. Only the Conceptual Captions dataset is used during the pretraining phase (rather than a combination of multimodal datasets). After pruning some examples present in their downstream task test sets, and accounting for broken links, Bugliarello et al. (2020) were left with 2.77M entries.

4.2.3 Finetuning

The finetuning is performed on the NLVR2 dataset. Two images (I_1 and I_2) and one sentence S are given as input. The model will obtain the unique representation of the image-text pair for both (I_1, S) and (I_2, S). The task is therefore viewed as a classification problem in which S is

Hyperparameters		mUNITER	xUNITER
Pretraining	Batch size (cross-modal training)	256	256
	Batch size (text training)	256	256
	Gradient accumulation step	4	4
	Max (cross-modal) seq. length	66	66
	Max (textual) seq. length	66	66
	Learning rate	10^{-4}	10^{-4}
	Adam optimizer epsilon	10^{-6}	10^{-6}
	Adam optimizer betas	0.9, 0.999	0.9, 0.999
	Weight decay	10^{-2}	10^{-2}
	Warmup proportion (linear learning rate)	10%	10%
	Number of epochs	10	10
Finetuning	Gradient accumulation step	2	4
	Adam optimizer epsilon	10^{-6}	10^{-6}
	Adam optimizer betas	0.9, 0.999	0.9, 0.999
	Weight decay	10^{-4}	10^{-4}
	Warmup proportion (linear learning rate)	10%	10%
	Number of epochs	20	20

Table 4.1: Hyperparameters used to pretrain and finetune mUNITER and xUNITER.

true if both (I_1, S) and (I_2, S) are true. The representation of the two pairs are passed through a 2-layer MLP with GeLU activation and a final softmax on two classes (true or false).

4.2.4 Hyperparameters

The various hyperparameters used for pretraining and finetuning for both mUNITER and xUNITER are summarised in table 4.1. The gradient accumulation step parameter means that we will accumulate the gradient for x steps before updating the model parameters. The Adam optimizer (Kingma and Ba, 2014) is used to optimize stochastic loss function and is initialised with an epsilon and two betas parameters.

Chapter 5

Multilingual multimodal pretraining

5.1 Motivations

In this chapter, we will examine the implications of using a multilingual multimodal dataset such as WIT (Srinivasan et al., 2021) for pretraining large-scale multilingual visio-linguistic models. In fact, top-performing models on MaRVL (Liu et al., 2021) like M^{3}P (Ni et al., 2020) and UC^2 (Zhou et al., 2021) did not rely on WIT for their pretraining. They rely on a machine-translated version of Conceptual Captions (Sharma et al., 2018) as an alternative. Even though it produces high-quality image-text pairs, scaling visio-linguistic models to a large number of other languages may be difficult if additional computational resources are required. Consequently, we will measure the benefits we can get from further pretraining mUNITER using the WIT dataset. In addition, we will limit the number of entries per language to $18,500$ in order to keep it as lean as feasible.

After introducing related works (section 5.2), we will further pretrain mUNITER and compare two setups, one with random data and the other with selected data (section 5.3). Then, in section 5.4, we will attempt to mitigate the overfitting that occurs during NLVR2 finetuning (in English). Finally, we will study how concepts covered during pretraining may affect performances on the downstream task, in section 5.5.

5.2 Related works

By proposing new training objectives, multimodal models have been adapted to a multilingual context in recent years. M^{3}P (Ni et al., 2020), is one of the first large-scale examples. It implemented multimodal code-switched training to obtain universal representations for multilingual textual and visual input. Multimodal code-switched training can be further divided into three distinct objectives: Multimodal Code-switched Masked Language Modeling (MC-MLM) (given visual and textual code-switched tokens, the model should predict the masked textual token), Multimodal Code-switched Masked Region Modeling (MC-MRM) (the model should reconstruct a visual region given other visual and textual code-switched tokens),

and Multimodal Code-switched Visual-Linguistic Matching (MC-VLM) (image-text matching with code-switched caption). M^{3}P obtained state-of-the-art performances for non-English image-text retrieval on the MSCOCO (Lin et al., 2014) and Multi30K benchmarks (Elliott et al., 2016). To summarise, M^{3}P utilised English to transition from other languages to the visual modality.

UC^2 (Zhou et al., 2021) is a new framework for training multilingual multimodal models. It relies on machine-translation to augment a monomodal multimodal datasets (Conceptual Caption 3M). To enable multilingual inputs, they adapted the standard Masked Language Modeling and Image-Text Matching training objectives. They then introduced two new objectives: Masked Region-to-Token Modeling (MRTM) (Given a visual input with a masked region, the model predicts its associated object label based on the other regions' labels and the caption) and Visual Translation Language Modeling (VTLM) (Given an image and a pair of captions in different languages, the model predicts some masked textual token in the two languages). The UC^2 framework-trained model achieved state-of-the-art performance in non-English languages for image-text retrieval (MSCOCO and Flickr30K (Young et al., 2014) benchmarks) and for visual question answering (VQA v2 benchmark) while matching English state-of-the-art performance. In conclusion, the UC^2 system uses visual inputs as a pivot to connect multilingual textual sources.

Finally, Zeng et al. (2022) presented CCLM^3, which was trained with the cross-view language modelling framework. It takes advantage of the fact that cross-lingual and cross-modal training objectives share a similar goal: aligning two distinct views of the same object inside a unique semantic space. They rely on a vision encoder, a cross-lingual encoder, and a fusion model. The input, whether it be a (text and image) pair or a (text and translation) pair, is always viewed as two distinct perspectives of the same item. Using a contrastive loss, a matching loss, or a conditional masked language modelling loss, the fusion model then aligns their respective representations. This model is employed for both translingual and transmodal fusion. CCLM^3 achieved state-of-the-art performances on IGLUE (Bugliarello et al., 2022) (multilingual multimodal benchmark testing, visual question answering, cross-modal retrieval, grounded reasoning, and grounded entailment) and image-text retrieval benchmarks (MSCOCO and Flickr30K). More importantly, it is the first model to achieve higher results on the target language test sets than state-of-the-art English multimodal models on the translate-test version of those test sets.

5.3 Impact of further pretraining on WIT

5.3.1 Experimental setup

We will leverage the WIT dataset to further pretrain the mUNITER model. As a reminder, we begin with a model that has already been trained for 10 epochs (2.77 million entries) on the Conceptual Captions dataset (Sharma et al., 2018) (monolingual multimodal) and the Plaintext Wikipedia Dump (multilingual monomodal) using all the languages. This model will be referred

to as $baseline\_pretrained$.

To compare outcomes across languages, we will employ the same number of training entries for each. As we only have $19,695$ entries for Swahili, we will use $18,500$ pretraining items and keep $1,000$ for validation. We will therefore start from the $baseline\_pretrained$ model and further pretrain it for each language for 25 epochs. It will give us five different models.

The additional pretraining consists of alternating between:

- A monolingual monomodal language training with MLM on the target language's Wikipedia Plaintext
- A multimodal monolingual training with image-text (in the target language) pairs from WIT.

Lastly, to gain a better understanding of the significance of data quality, we will evaluate two distinct situations:

- With data drawn at random from the set of potential entries (for each language).
- With selected data. For each target language concept (see table 3.2 for the number), the closest WIT entries will be chosen. For this, we will use fastText (Grave et al., 2018) (for each language using its corresponding model[1]) to compute the Wikipedia article title embedding for each WIT entry and for each MaRVL concept. We decided to compute embedding of the article's title as it is less likely to contain noisy information. Finally, we will compute the cosine similarity[2] between each concept and the article title and select the closest k. k depends on the language and is set so that the generated training set has close to $18,500$ entries. The final training datasets have $18,522$ entries ($k = 460$) for Indonesian, $18,502$ entries ($k = 799$) for Mandarin Chinese, $18,508$ entries ($k = 634$) for Tamil, and $18,507$ entries ($k = 678$) for Turkish. As the number of Swahili entries is already close to $18,500$, the selected training data would not differ significantly from random data. Thus, we will not apply this configuration for Swahili.

Finally, all pretrained models will be finetuned on NLVR2 for 10 epochs and tested on their respective MaRVL test set (which is unique for each language) in a zero-shot manner. $baseline\_pretrained$ will also have the same finetuning procedure, and the resulting model will be referred to as $baseline$. The $baseline$ model will be evaluated on all languages.

For each language and setup, there will be three runs. For the random data setup, we will assemble three distinct training set. In contrast, for the selected data setup, we will use the same training set with different training seeds.

[1] https://fasttext.cc/docs/en/crawl-vectors.html

[2] $c(x,y) = \frac{x.y}{\|x\|^2.\|y\|^2}$

		ID	SW	TA	TR	ZH	avg.
mUNITER	baseline	55.2	52.1	52.0	54.1	**57.4**	54.2
	w/ random WIT data	54.6	**54.2**	**53.5**	55.9	54.5	54.5
	w/ selected WIT data	**56.5**	**54.2**	53.3	**56.9**	54.0	**55.0**

Table 5.1: mUNITER performance (accuracy) on MaRVL. Best scores are put in bold, but do not imply statistical significance. The random data and selected data setups for Swahili are the same.

5.3.2 Results & analyses

We retained the models with the highest performance on the NLVR2 validation set. Then, for each language, we take the average of the three runs' outcomes. The results are displayed in table 5.1.

For the reasons stated previously, random and selected data configurations for Swahili are deemed equivalent. Overall, it appears that further pretraining with data from the WIT dataset enhances performance. At least one of the two setups increases accuracy for four out of the five languages. However, for Mandarin Chinese, this extra pretraining step appears to degrade performance. If English and Mandarin Chinese concept embeddings were already well aligned following the first pretraining, a second pretraining conducted exclusively in Mandarin Chinese could cause them to diverge.

To acquire a better grasp of the progress of training, we will now display the accuracy vs. the epoch number for each language, as well as the average across all languages. Figure 5.1 depicts the resulting plot for the mean across all languages. Individual language plots are available in the appendix. Figure A.1 for Indonesian , Figure A.2 for Swahili, Figure A.3 for Tamil, Figure A.4 for Turkish, and Figure A.5 for Mandarin Chinese.

After two epochs, the average performance reaches its peak ($+2.5\%$ accuracy relative to the baseline model) and begins to decline after four epochs. It may be the result of overfitting. During the extra training step, each model was trained solely on target language data. As a result, the gain in target language understanding may diminish the longer the model is finetuned using English-only data.

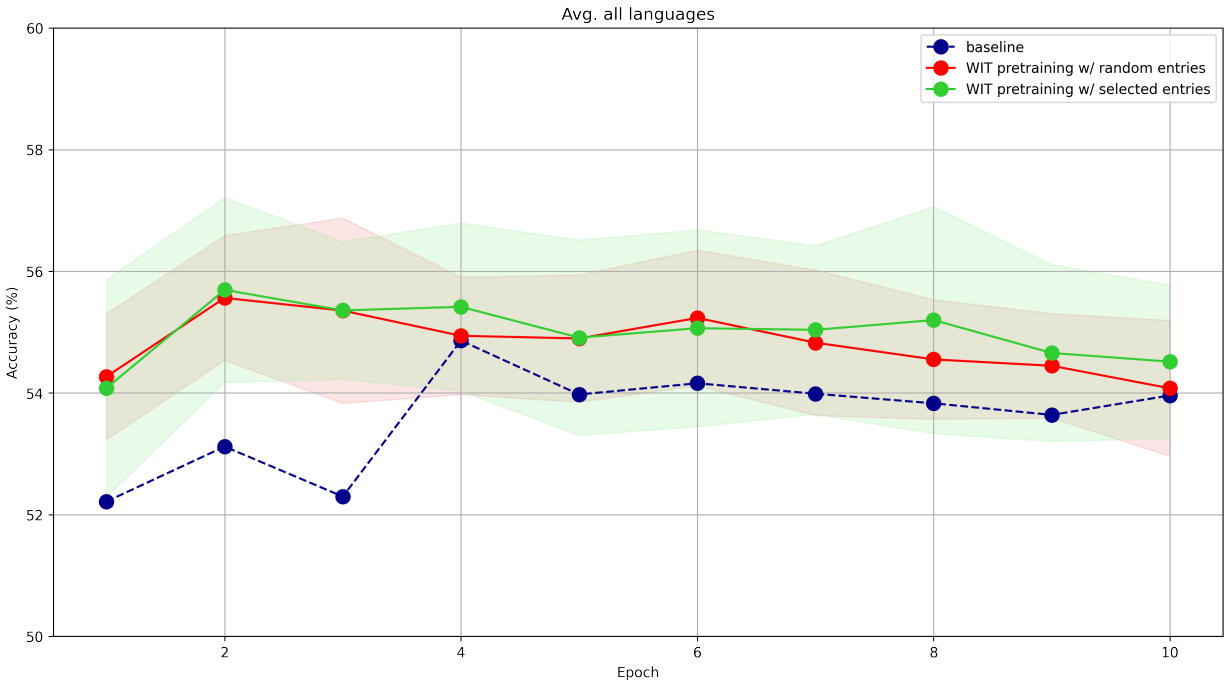

Figure 5.1: Average performance (accuracy) across languages on the MaRVL test set vs. number of training epochs on the NLVR2 dataset. The baseline is a single model that has been evaluated for each language. The other two configurations are different models (one for each language) whose results were averaged.

5.4 Impact of the amount of finetuning data

5.4.1 Experimental setup

In the previous experiment, we realised that our models may be overfitting on English data during the finetuning step on NLVR2. We will attempt to limit this phenomenon by adjusting the amount of entries used for finetuning. We will start with the models further pretrained on random WIT data and finetune them for 10 epochs on varying quantities of NLVR2 data ($5,000$, $25,000$, $45,000$, $65,000$, and the full dataset, which has around $86,000$ entries).

For each language and setup, we will do three runs: we will take the same pretrained model (the one with the best performance in the previous experiment, section 5.3) for every combination but will have different finetuning seeds.

5.4.2 Results & analyses

We keep the models achieving the best performance on NLVR2's validation set. Besides, the results of the three runs for each combination (language and setup) are averaged out and can be observed in table 5.2.

Additionally, the results averaged out across languages can be seen in figure 5.2. Individual language plots are available in the appendix. Figure B.1 for Indonesian , Figure B.2 for Swahili,

		ID	SW	TA	TR	ZH	avg.
mUNITER	baseline	55.2	52.1	52.0	54.1	**57.4**	54.2
	w/ 5,000 entries	55.1	54.3	53.7	55.8	55.7	54.9
	w/ 25,000 entries	**56.6**	**55.3**	**54.5**	**57.3**	55.2	**55.5**
	w/ 45,000 entries	54.9	**55.3**	54.3	55.7	55.4	55.0
	w/ 65,000 entries	55.0	54.9	53.6	55.4	54.8	54.8
	w/ 86,373 entries (full NLVR2 dataset)	55.3	54.8	54.0	55.8	55.3	54.9

Table 5.2: mUNITER performance (accuracy) on MaRVL when varying the amount of finetuning data (NLVR2). Best scores are put in bold, but do not imply statistical significance.

Figure B.3 for Tamil, Figure B.4 for Turkish, and Figure B.5 for Mandarin Chinese.

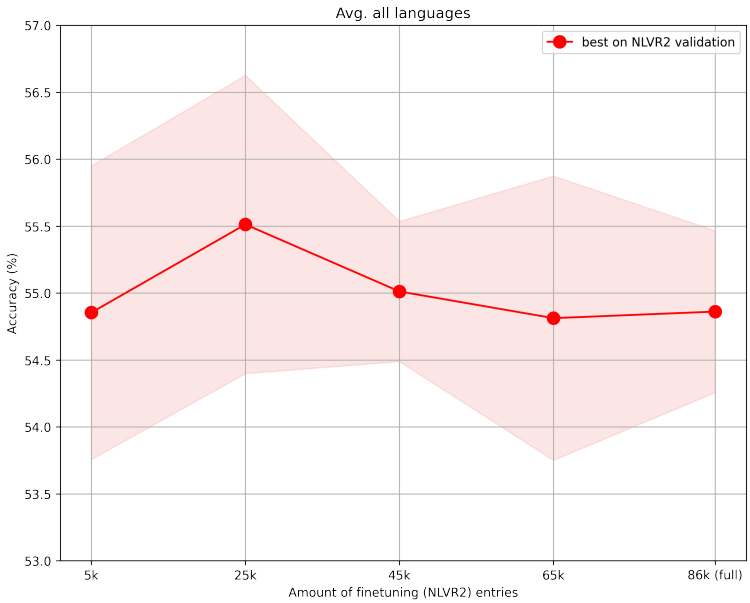

Figure 5.2: Average performance (accuracy) across languages on the MaRVL test set vs. amount of finetuning data (NLVR2) used.

Overall, for four out of five languages, the best performances are achieved when using only $25,000$ NLVR2 entries for finetuning. In the case of Turkish and Swahili, we even increased accuracy by 3% with this configuration. With fewer finetuning entries, both performances and variances are higher. For Mandarin Chinese, the scores are still lower than the baseline, but the performance is unaffected when using $25,000$ entries compared to other quantities.

Now, when we discuss finetuning, we will just consider $25,000$ NLVR2 entries (as opposed to the entire dataset) for 10 epochs. It will benefit both performance and required training time. In conclusion, we estimate that additional pretraining of the $baseline\_pretrained$ model using the

		ID	SW	TA	TR	ZH
title_concept_set	# entries	2,676	295	2,206	1,428	3,350
	# concepts	90	65	78	67	71
entry_concept_set	# entries	7,817	1032	7,815	3,016	8,797
	# concepts	91	71	78	73	87

Table 5.3: Number of entries and concepts covered for each set.

WIT dataset can result in a 1.3% gain in accuracy relative to the baseline.

5.5 Impact of concepts covered during pretraining

5.5.1 Experimental setup

We will attempt to quantify the benefit of pretraining models on image-text pairings containing the same concepts as those used in the downstream task, i.e. MaRVL concepts. The Turkish MaRVL test set, for instance, has the notion "keman" (violin). During pretraining on WIT data, if a picture of a violin and its Turkish caption were fed to the model, would it have an effect on its downstream performance?

To answer this question, we will first construct two separate sets for each language.

- A set of all WIT entries whose titles contain[3] a MaRVL concept. It shall be called the **title_concept_set**.
- A set of all WIT entries where a MaRVL concept appears[3] in the title, section title (if appropriate), or caption. It shall be called the **entry_concept_set**.

Table 5.3 provides a summary of the number of entries and concepts included in each of these sets by language.

Using these sets, we can now construct different training datasets (consisting of $18,500$ entries) for each language:

- The full coverage datasets. They include all entries from the **title_concept_set** (in their respective languages) plus random entries from the remaining possible set (no duplicates). We chose items from the **title_concept_set** because they are more likely to emphasise only the concepts of interest (Wikipedia articles were dedicated to these concepts).
- The zero coverage datasets. We first take all possible entries from WIT in a given language and prune all the ones included in the **entry_concept_set**. It assures that no resultant

[3] We consider a sentence (title, section title, or caption) to contain a MaRVL concept if: it contains an exact word match for languages using the Latin alphabet (i.e., if the concept is a substring of a word, it will not be considered); and if the string is present in the sentence for Mandarin Chinese and Tamil.

		ID	SW	TA	TR	ZH	avg.
mUNITER	zero coverage	**55.7**	54.4	**55.5**	**57.2**	**56.2**	**55.8**
	full coverage	55.2	**54.8**	55.4	56.2	55.5	55.4

Table 5.4: mUNITER performance (accuracy) on MaRVL zero vs. full coverage of concepts. Best scores are put in bold, but do not imply statistical significance.

entry contains a MaRVL concept. Then, we randomly select entries from this pruned collection.

We will do three runs per configuration (language and setup). Each run will utilise a different pretraining dataset, and will then be finetuned on $25,000$ NLVR2 entries.

5.5.2 Results & analyses

Again, we retain the models with the greatest performance on the validation set for NLVR2. In addition, the average results of the three runs for each combination (language and setup) are displayed in table 5.4. Figure 5.3 presents a summary of the two setups' accuracy by language. On top of that, individual language's results (accuracy and standard deviation) are presented in the appendix. Figure C.1 for Indonesian , Figure C.2 for Swahili, Figure C.3 for Tamil, Figure C.4 for Turkish, and Figure C.5 for Mandarin Chinese.

The two distinct configurations, zero concept coverage and full concept coverage, have no meaningful effect on the outcomes. Surprisingly, it appears that the zero concept coverage setup achieves superior accuracy for four languages. However, the difference with the full concept coverage setup is always less than or equal to 1%, which is insufficient to draw a definitive conclusion.

Therefore, we will not consider selecting entries with concepts coverage for the pretraining stage. From now on, when performing additional pretraining using the WIT dataset, we will use the random entries from the first experiment (section 5.3).

5.6 Conclusion

In this chapter, we have explored pretraining with WIT, a multimodal, multilingual dataset. Initially, we realised that by exploiting extra multimodal data in a target language, we may improve performance. In particular, preselecting pretraining entries might even increase accuracy on MaRVL. However, training a model in the target language and then finetuning it in English can result in overfitting. In the first few epochs, we proved that accuracy increases rapidly and outperforms the baseline by 2.5% on average. Unfortunately, after 10 epochs, accuracy returns to baseline levels. To address this issue, we have demonstrated that the amount

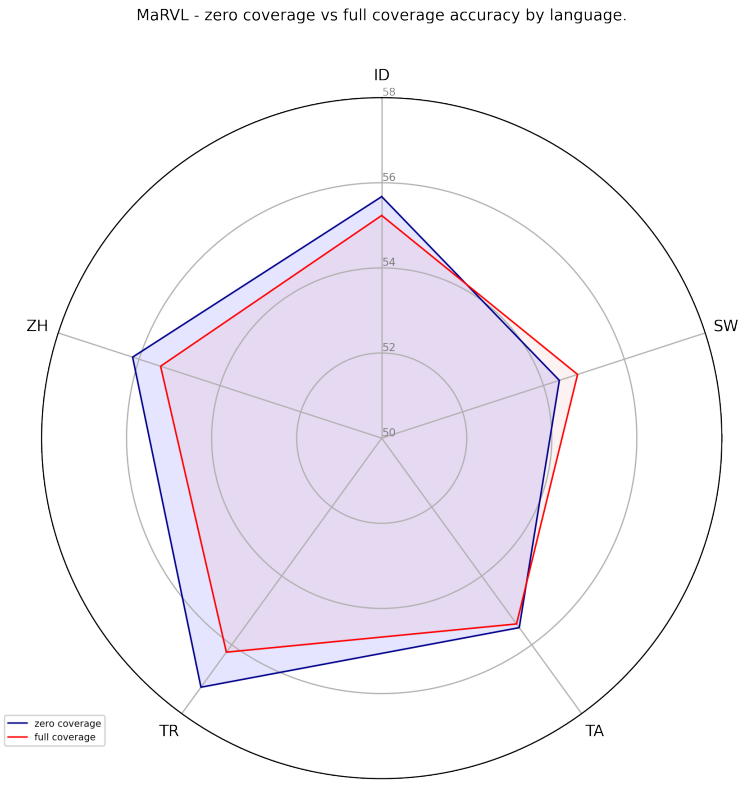

Figure 5.3: mUNITER performance (accuracy) on the MaRVL test set for zero and full coverage setups by language.

of English finetuning data (NLVR2) can be decreased. By using less than a third of the initial finetuning dataset, we can achieve more robust performance. Finally, we established that picking pretraining entries covering the same concepts as those in MaRVL does not always result in improved accuracy.

These results demonstrate the advantages of utilising WIT as a pretraining dataset for multilingual visio-linguistic models and shed light on several aspects of the pretraining data.

Chapter 6

Code-switching

6.1 Motivations

So far, the additional WIT pretraining has not provided a significant advantage over the baseline. In addition, we remain closer to the random frontier on average than the translate-test baseline. Figure 6.1 illustrates the performance of the baseline model on MaRVL and its translated version by language for mUNITER and xUNITER. In particular, the gap for Swahili, Tamil and Turkish is greater than 5% for these two configurations. This section will investigate the advantages of code-switching. It is a data agumentation technique where words in a target language are translated into English or vice versa. It has been proved that it improves the model's conceptual grounding across languages (Ni et al., 2020).

We will first start by introducing related works in section 6.2. Then, we will present how we constructed the bilingual dictionaries (section 6.3) to perform our code-switching pipeline (section 6.4). Furthermore, this data augmentation technique will be studied in a variety of contexts, including at test time (by translating some MaRVL test set words into English) in section 6.5, and for both pretraining and finetuning in section 6.6. Additionally, we will investigate which parts-of-speech tags provide the greatest boost when translated in section 6.7. Using this methodology, we will determine whether embeddings of the same concept across languages become more similar (section 6.8).

6.2 Related works

Code-switching is a data augmentation technique in which tokens from a given language's textual input are translated into another language using a bilingual dictionary. This approach was initially implemented in a monomodal NLP context. Bilingual dictionaries cover many more languages than standard multilingual labelled datasets, making this an attractive solution. Wang et al. (2022) indicated that employing bilingual dictionaries permits the expansion of current NLP systems to thousands of languages, significantly more than the 104 languages covered by mBERT, for instance. CoSDA-ML (Qin et al., 2020) is one of the first works to

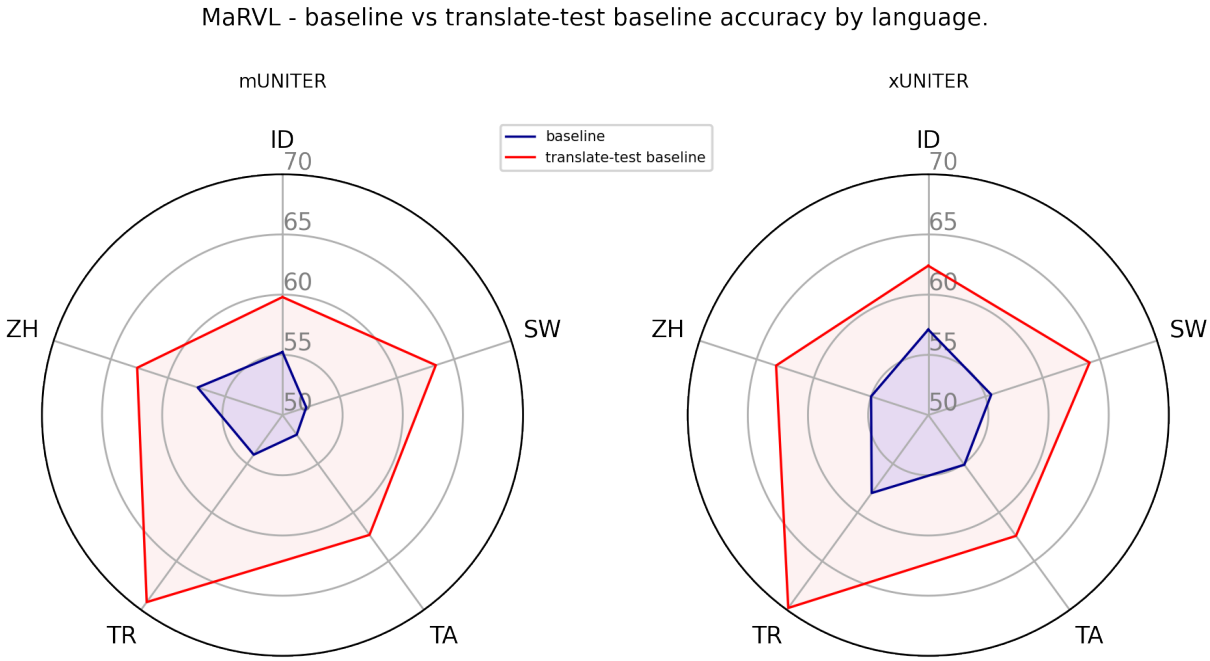

Figure 6.1: Performance (accuracy) of the baseline mUNITER and xUNITER on MaRVL and its translated-test version by language.

demonstrate how code-switched training can enhance zero-shot cross-lingual transfers. Using the framework to fine-tune mBERT for five tasks and 19 languages yielded an average performance increase of 2.9%. Additionally, code-switching was expanded to pretraining (rather than only finetuning) and to other tasks. Yang et al. (2020) pretrained a Neural Translation System with an encoder-decoder architecture using a code-switching framework. During pre-training, a code-switched sentence was fed to the encoder. The decoder should then retrieve the non-translated version of the code-switched tokens.

Finally, the code-switching framework was extended to support Visio-Linguistic model training. M^{3}P (Ni et al., 2020), a ground-breaking work for multilingual multimodal models, exploited this framework to augment a monolingual multimodal (image-caption pairs in English) data stream during pretraining. They achieved state-of-the-art performances on multilingual image-text retrieval tasks. Moreover, they examined the advantages of code-switching when it is only used for finetuning.

6.3 Constructing bilingual dictionaries

To implement code-switching, we must first construct bilingual dictionaries. In an effort to use as few computational resources as possible, we will attempt to utilise existing dictionaries.

We choose to employ Panlex, the largest lexical database in the world, which has $5,700$ languages, $25,000,000 words$, and $1,300,000,000$ translations. To extract translations, we relied

	ID	SW	TA	TR	ZH
EN → Target language	20,574	12,839	14,615	22,691	28,586
Target language → EN	11,251	8,347	8,889	13,649	29,637

Table 6.1: Number of bilingual pairs for each dictionary. It corresponds to the number of pairs for which the source word is non-composite, i.e., expression containing two or more words are excluded.

on the plexy repository[1]. It accepts as input a list of words in a source language and produces a bilingual dictionary.

For our list of English words, we began using WordNet (3.0) (Miller, 1995). It offers two benefits:

- All bilingual dictionaries will be comparable and overlapping (albeit some translations may be absent for some language pairs), making it easy to compare performance gains across languages.
- The WordNet provides information on univeral part-of-speech tags[2] that can be used to minimise the noise in word-to-word translation. Instead of translating one word into another, we can translate a pair (word and part-of-speech tag) into another pair. WordNet is a collection of nouns, verbs, adjectives, and adverbs.

There are $41,185$ unique (word, part-of-speech tag) pairs in WordNet.It is equivalent to $32,002$ distinct words. Indeed, some words can have different part-of-speech tags, resulting in different meanings. For instance, "green" can be both a noun (a piece of grassy land) and an adjective (a color). We will retrieve translations for the $32,002$ unique English words in each MaRVL language. If a word has two x distinct parts-of-speech tags, we will add x entries, one per part-of-speech tag, to our bilingual dictionary. We cannot anticipate a word's most frequent grammatical category in a given language. Thus, we add every possibility to our dictionary.

Table 6.1 provides a summary of the number of entries (pairs of words and part-of-speech tag) in each bilingual dictionary.

In addition, we provide smaller bilingual dictionaries for MaRVL test sets. Using Google Translation, we construct dictionaries for each language that map test-set words to English. These dictionaries do not include part-of-speech tags information.

[1] https://github.com/fdschmidt93/plexy
[2] https://universaldependencies.org/u/pos/

6.4 Code-switching training pipeline

We will now introduce a new online code-switching procedure for multimodal training. It is online, meaning that data will be modified at each training epoch. It is multimodal in the sense that words from sentences paired with images are code-switched. We are utilising two novel code-switching mechanisms:

- First, it uses part-of-speech tags to decrease the noise in word-to-word translations. We will use the same tags as the ones in WordNet: NOUN, VERB, ADJ (adjective), and ADV (adverb).
- Second, it is reciprocal. It implies that we can code-switch words from English to a target language and vice versa (depending on whether we are at the pretraining with WIT or finetuning on NLVR2 step).

To leverage part-of-speech tags, we will rely on two additional repositories. MultiCOMBO[3] is a multilingual part-of-speech tagger that uses mBERT as its core language model. It accepts sentences as input and outputs the universal part-of-speech tags for each of the terms in each sentence. For Mandarin Chinese, we employ the jieba[4] tool, which performs word segmentation and part-of-speech tagging. However, it generates part-of-speech tags with the ictclas[5] labelling system. The ictclas tags were manually mapped to universal part-of-speech tags. Each caption's (WIT) and each sentence's (NLVR2) part-of-speech tags will be precomputed.

Besides, we will also code-switch target language words to English during the pretraining phase with the WIT dataset. This poses additional difficulties in comparison to English. The first step is segmentation. With the exception of Mandarin Chinese, we shall divide words at every space character. We will use the jieba repository once again for Mandarin Chinese word segmentation. The second step is lemmatization, which maps a word to one of its root forms. Only Turkish and Tamil, which are particularly agglutinative languages, will undergo this process. We will try to select the longest lemmatized version of each unique word in the set of WIT captions that is not already in the bilingual dictionary. To lemmatize Turkish terms, we shall rely on the zeyrek repository[6]. For Tamil, we will utilise the IndoWordNet API provided through pyiwn[7].

Lastly, if a pair (word, part-of-speech tag) exists in the bilingual dictionary, we translate the word with a 0.5 probability, such that depending on the epoch, the model sees a word in either the target language or English.

[3] https://github.com/KoichiYasuoka/MultiCOMBO
[4] https://github.com/fxsjy/jieba
[5] https://www.lancaster.ac.uk/fass/projects/corpus/ZCTC/annotation.htm
[6] https://github.com/obulat/zeyrek/
[7] https://github.com/cfiltnlp/pyiwn

Figure 6.2 displays two instances that illustrate how the code-switching technique operates. The metrics regarding the code-switching pipeline are presented in Table 6.2.

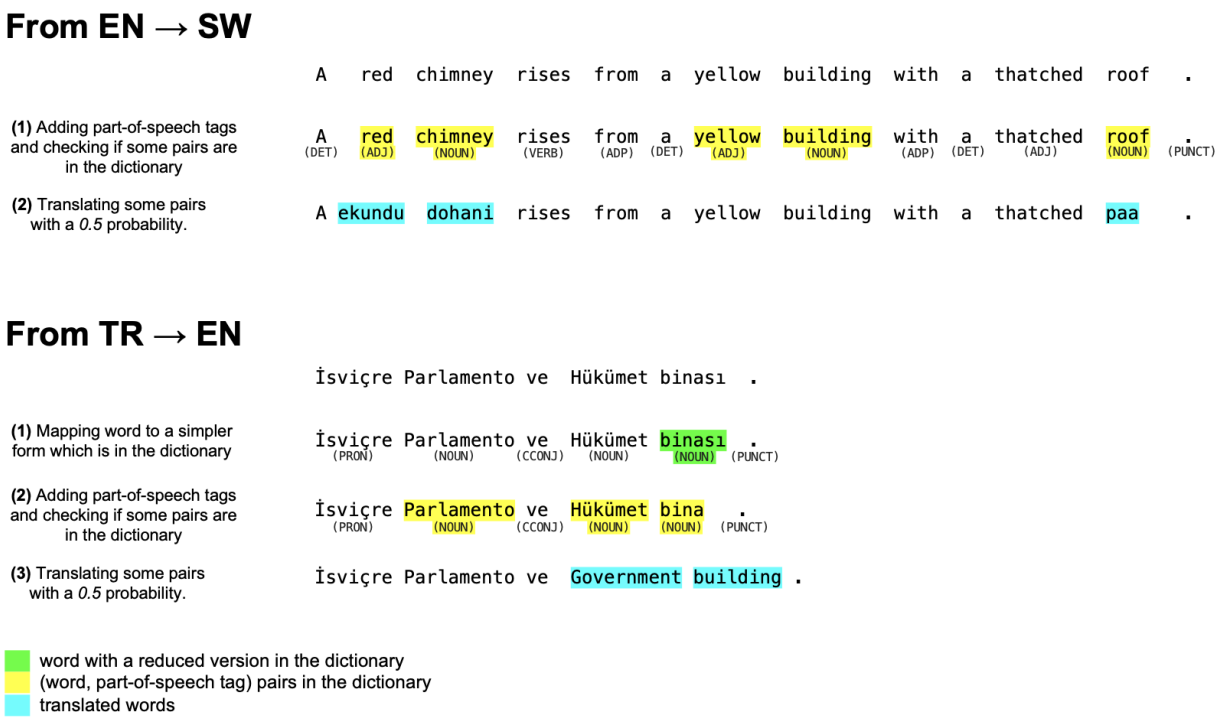

Figure 6.2: Examples of the code-switching pipeline. The first scenario illustrates what occurs during the English to Swahili finetuning process. The second example illustrates the Turkish to English translation example used during pretraining.

6.5 Code-switched MaRVL test set

6.5.1 Experimental setup

To gain a preliminary understanding of the advantages of code-switching, we will first measure the baseline performances of mUNITER and xUNITER on a code-switched version of the MaRVL test sets (one per language).

We will employ the MaRVL dictionaries we constructed. They translated every word in the test sets into English. First, words in the target language will be randomly translated into English with a probability of x. The range of x will be from 0 to 100 with 12.5 increments.

In addition, we will develop two test sets per language that correspond to:

- a test set containing code-switched MaRVL concepts. For example, if a MaRVL entry is about the concept "mbwa" (Swahili for "dog"), then if its sentence contains the word, it will be code-switched to "dog."

		ID	SW	TA	TR	ZH
EN → Language (NLVR2 finetuning)	total code-switched words (%)	29.7	28.6	29.2	30.1	30.6
	(%) dictionary used	12.4	16.8	15.3	11.3	9.4
	unique words code-switched	2,542	2,158	2,236	2,572	2,701
Language → EN (WIT pretraining)	total code-switched words (%)	20.4	19.3	15.0	30.7	22.1
	(%) dictionary used	51.4	24.0	29.6	48.3	29.0
	unique words code-switched	5,787	2,004	2,635	6,597	8,597

Table 6.2: Code-switching metrics for each language and each direction (target language to English or vice-versa). It indicates the total number of words that can be code-switched (in %) relative to the total number of words in the corpus (pretraining captions or NLVR2 sentences), the number of distinct words that were code-switched and the corresponding dictionary usage, i.e., the proportion of dictionary words that were ultimately code-switched (in %).

- A test set constructed using the same pipeline described in section 6.4 and the Panlex dictionaries. It will serve as a benchmark for the next code-switched training experiment.

For all obtained languages and test sets, the performance of mUNITER and xUNITER will be evaluated. The outcomes of each model are plotted by language.

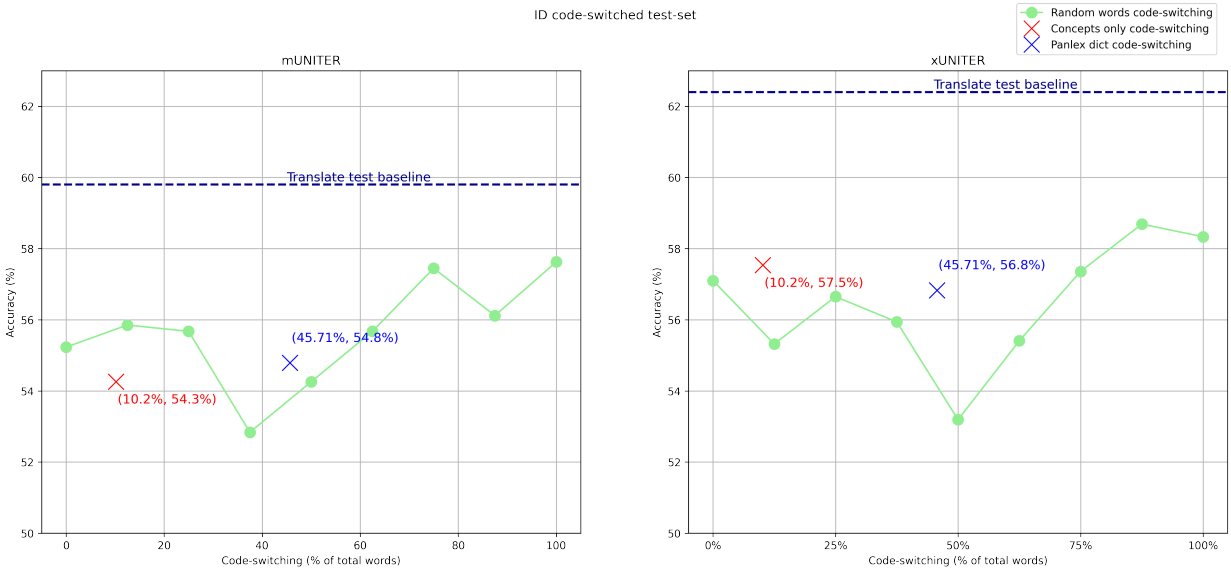

Figure 6.3: mUNITER and xUNITER baseline models evaluated on code-switched versions of the Indonesian MaRVL test set.

6.5.2 Results & analyses

Figure 6.3 depicts the results for Indonesian, Figure 6.4 for Swahili, Figure 6.5 for Tamil, Figure 6.6 for Turkish, and Figure 6.7 for Mandarin Chinese.

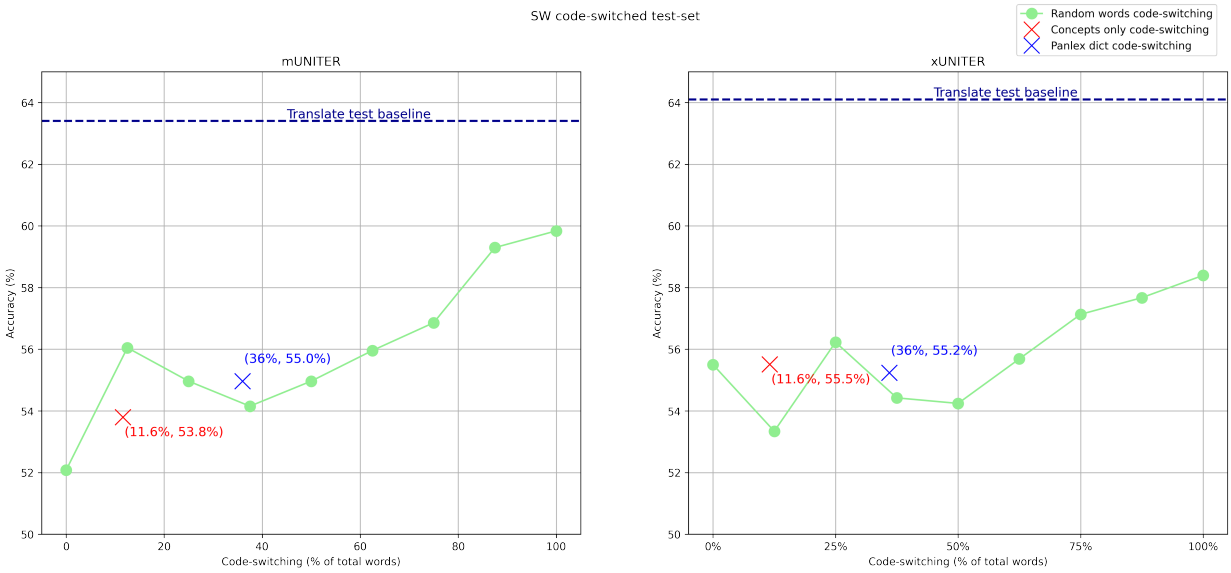

Figure 6.4: mUNITER and xUNITER baseline models evaluated on code-switched versions of the Swahili MaRVL test set..

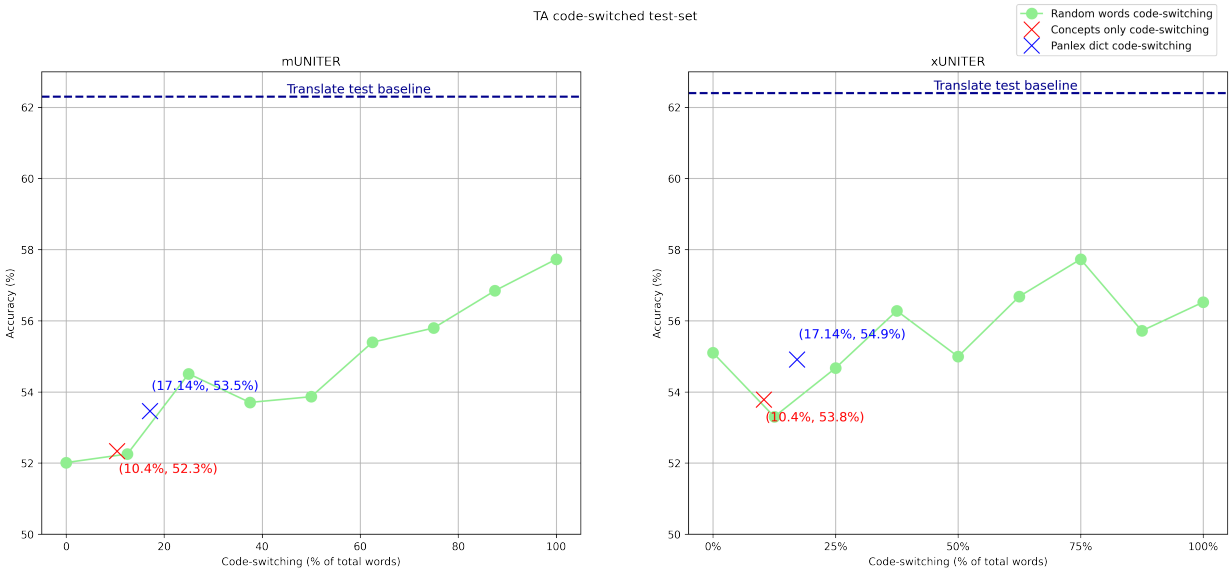

Figure 6.5: mUNITER and xUNITER baseline models evaluated on code-switched versions of the Tamil MaRVL test set..

We can make the following observations across languages and models (mUNITER and xUNITER):

- The greatest results are consistently obtained on 87.5% or 100% code-switched test sets. It means that the models improve when more English words are present.
- The best results on code-switched tests are still well behind the translate-test baseline (even

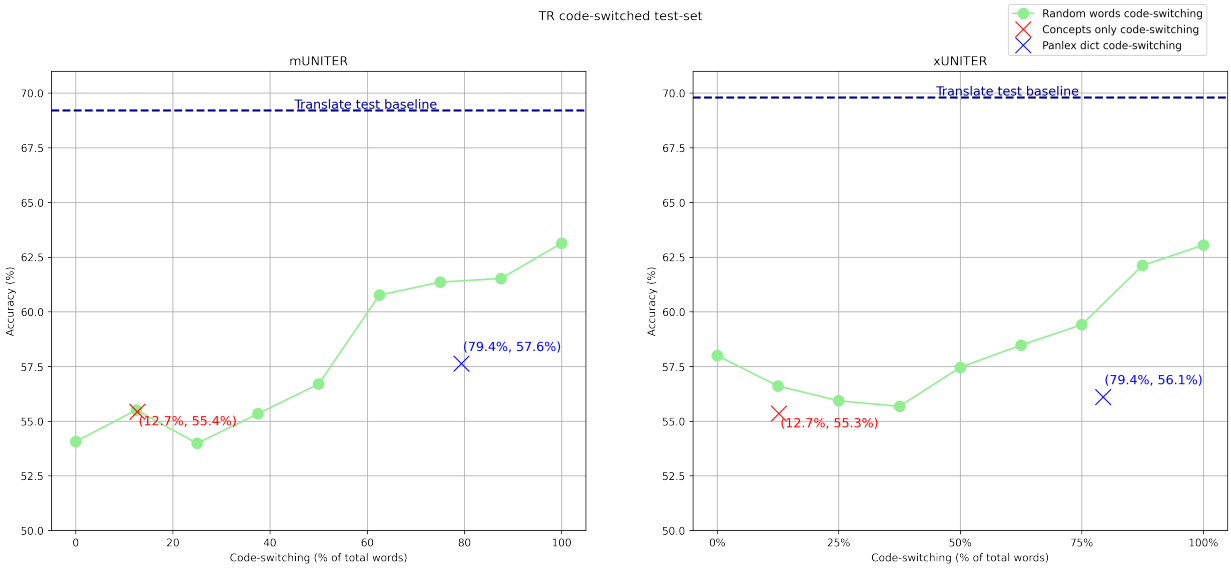

Figure 6.6: mUNITER and xUNITER baseline models evaluated on code-switched versions of the Turkish MaRVL test set..

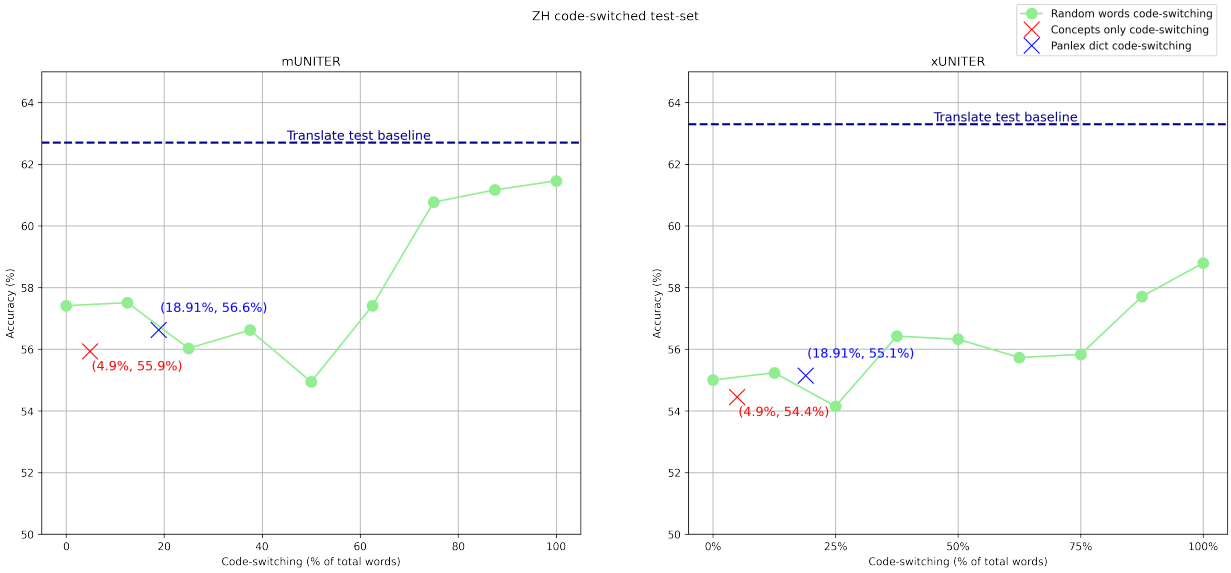

Figure 6.7: mUNITER and xUNITER baseline models evaluated on code-switched versions of the Mandarin Chinese MaRVL test set..

if we translate all words). Indeed, when performing word-by-word translations, we may produce poorly structured English sentences. However, a few key translations may be sufficient to achieve a significant performance gain over the original test set's baseline.

- Simply translating the MaRVL concepts does not improve performance. It could imply that the MaRVL concepts are not the most crucial elements for the models' performance improvement.

- Using the code-switching pipeline with Panlex dictionaries on the test sets does not result in optimal performance.
- After 50% code-switched words, there is a noticeable upward trend. It may suggest that the models need to see an important part of a sentence in English to really obtain higher accuracy. It indicates that models continue to struggle to connect textual concepts in a target language with visual features. This relationship appears clearer when the statement comprises more than 50% of English words.

6.6 Training with code-switching

6.6.1 Experimental setup

In this experiment, we will apply the code-switching training pipeline described in section 6.4.

First, to find the optimal setup, we will use code-switching only for the pretraining part on WIT (from a target language to English), solely for the finetuning part on NLVR2 (from English to a target language), and code-switching for both parts. For this part of the experiment, only mUNITER will be utilised (less costly to train). Once we determine the optimal configuration, we will utilise it to train xUNITER in a similar way.

As usual, each WIT pretraining set (for each language) has $18,500 entries$, while the finetuning set contains $25,000$ NLVR2 entries. We will do three runs for each configuration (each language, each code-switching configuration).

6.6.2 Results & analyses

In table 6.3, the average outcomes of the three trials are summarised. We demonstrate that the best results are achieved when the code-switching pipeline is used for both pretraining and fine-tuning. Consequently, we only use this configuration for xUNITER.

We realised that the code-switching pipeline applied just to pretraining does not significantly improve performance. However, when code-switching is used for fine-tuning, accuracy is considerably enhanced. Finally, the best performances are reached when the code-switching pipeline is applied for both pretraining and finetuning. Indeed, when we code-switch words during the finetuning phase, we give the models direct access to bilingual translations in a highly specific configuration, which is very similar to the one used later at test time on MaRVL.

Importantly, the achieved accuracies are greater than those obtained in the preceding section when we code-switched the test-set using the same pipeline and our Panlex dictionaries. It indicates that the benefits of code-switching training for models are more important than simply code-switching a test set using bilingual dictionaries.

Overall, we can achieve a 3.8% accuracy boost for mUNITER and 2.2% for xUNITER. The increases are not, however, uniform among languages.Those languages whose baseline score was less than 55% experienced the greatest gains. It is especially impressive for Swahili and

		ID	SW	TA	TR	ZH	avg.
mUNITER	baseline	55.2	52.1	52.0	54.1	**57.4**	54.2
	w/ code-switched pretraining	54.9	53.6	54.9	55.8	54.3	54.7
	w/ code-switched finetuning	57.3	57.3	**57.4**	59.6	**57.4**	57.8
	w/ both	**57.6**	**58.2**	56.7	**60.1**	**57.4**	**58.0**
xUNITER	baseline	57.1	55.5	55.1	58.0	55.0	56.1
	w/ both	**57.9**	**58.5**	**58.2**	**59.6**	**57.4**	**58.3**

Table 6.3: Performance (accuracy) by language for mUNITER and xUNITER on MaRVL with various code-switching configurations. Best scores are put in bold, but do not imply statistical significance.

Turkish, whose accuracy increases by 6.1% and 6.0%, respectively, when utilising mUNITER. When examining the code-switching metrics, it can be seen that the total number of words that can be code-switched for a language is not correlated with an increase in performance.

Finally, figure 6.8 depicts the baseline performances (both on the original MaRVL test sets and on their machine-translated equivalents) compared with our upgraded models. Even if we are still closer to the baseline performances than the translated-test ones, our models are now closer to the latter than to the random frontier (50%).

In appendix, Figure D.1 depicts the results with standard deviations for Indonesian, Figure D.2 for Swahili, Figure D.3 for Tamil, Figure D.4 for Turkish, and Figure D.5 for Mandarin Chinese.

6.7 Impact of different part-of-speech tags

6.7.1 Experimental setup

In this experiment, we will investigate the influence of various part-of-speech tags on code-switching. We will begin with the configuration and language for which we obtained the highest accuracy in the previous experiment (section 6.6), namely mUNITER for Turkish with code-switching for both pretraining and fine-tuning.

This time, however, rather than code-switching all nouns, verbs, adjectives, and adverbs, we will experiment with various combinations: verbs and nouns alone, and then only verbs and nouns. Table 6.5 provides a summary of the code-switching metrics for the various configurations.

We will conduct three runs (pretraining and finetuning) for each setting and then calculate the mean of the findings.

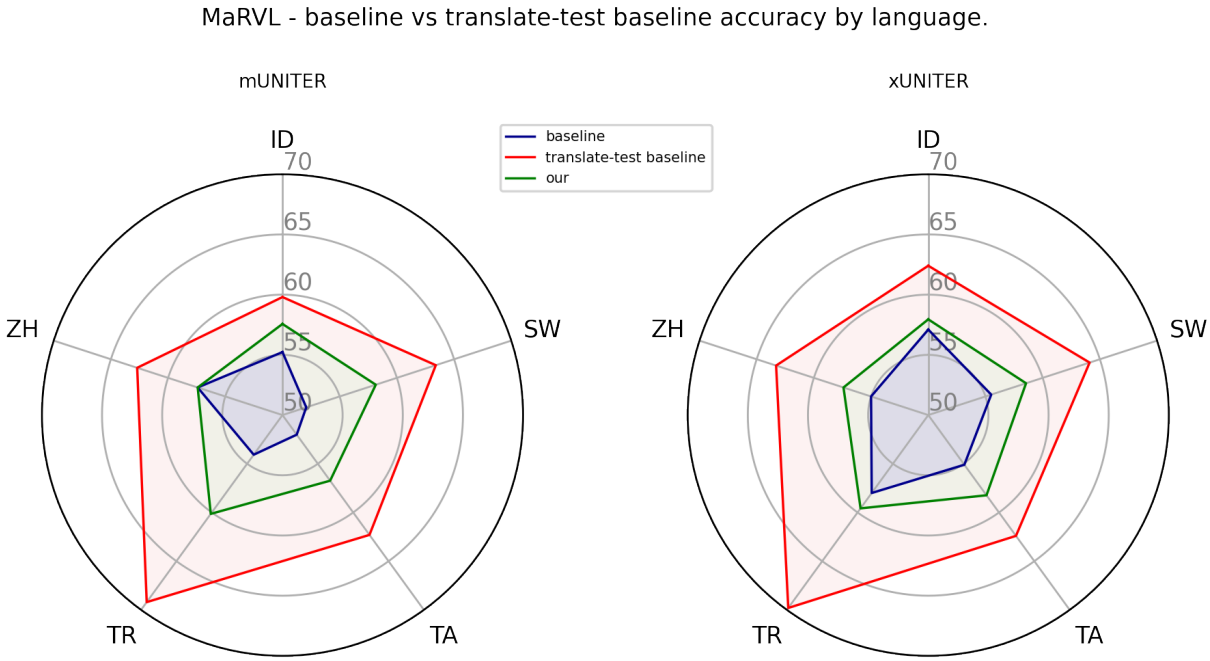

Figure 6.8: Performance (accuracy) by language of the baseline mUNITER and xUNITER on MaRVL, and on its translated-test version. In green, our best models obtained using code-switching.

6.7.2 Results & analyses

Table 6.4 and Figure 6.9 exhibit the averaged results.

We can see that code-switching verbs alone does not result in a significant performance increase. This can be explained by the fact that during pretraining and fine-tuning, only 3.8% and 0.7% of all words are translated, respectively.

Noun seems to be the part-of-speech tag with the greatest impact on performances. Indeed, when code-switching solely nouns, we obtain results that are just 0.3% lower than when considering all possible part-of-speech tags. On top of that, it required code-switching 7.8% and 10.4% fewer words during the pretraining and finetuning phases, respectively. When considering verbs and nouns together, we obtain similar results.

This experiment demonstrates that the "noun" is the most significant part-of-speech tag to consider for code-switching in order to increase accuracy. It operates similarly to configurations with more part-of-speech tags, but requires around 8% less code-switches. It could make it easier to create the necessary multilingual dictionaries in the future.

	part-of-speech tag(s)	TR
mUNITER	VERB	56.9
	NOUN	59.8
	VERB + NOUN	59.5
	ADJ + ADV + VERB + NOUN	**60.1**

Table 6.4: mUNITER performance (accuracy) on the Turkish MaRVL test set when considering different part-of-speech tags for code-switching. Best scores are put in bold, but do not imply statistical significance.

		TR → EN	EN → TR
VERB	total code-switched words (%)	3.8	0.7
	(%) dictionary used	6.2	0.9
	unique words code-switched	845	207
NOUN	total code-switched words (%)	22.9	19.7
	(%) dictionary used	34.7	7.7
	unique words code-switched	4,733	1,753
VERB + NOUN	total code-switched words (%)	26.8	20.4
	(%) dictionary used	40.9	8.6
	unique words code-switched	5,578	1,960
ADJ + ADV + VERB + NOUN	total code-switched words (%)	30.7	30.1
	(%) dictionary used	48.3	11.3
	unique words code-switched	6,597	2,572

Table 6.5: Code-switching metrics when including different part-of-speech tags in the Turkish code-switching pipeline. It indicates the total number of words that can be code-switched (in %) relative to the total number of words in the corpus (pretraining captions or NLVR2 sentences), the number of distinct words that were code-switched and the corresponding dictionary usage, i.e., the proportion of dictionary words that were ultimately code-switched (in %).

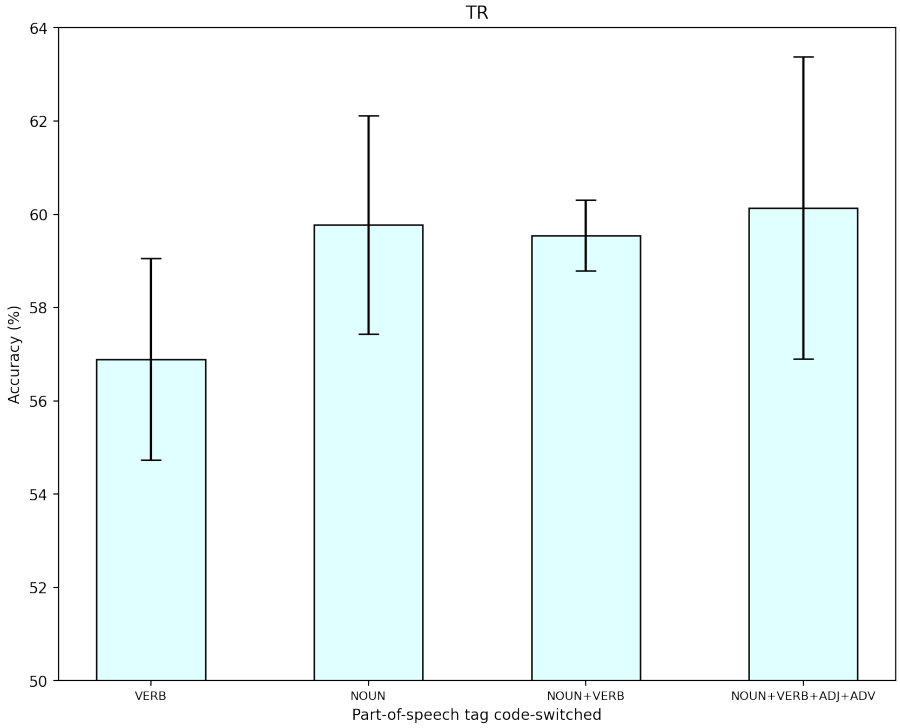

Figure 6.9: mUNITER performance (accuracy) on the Turkish MaRVL test set when including different part-of-speech tags in the code-switching pipeline.

6.8 Impact on word embeddings

6.8.1 Experimental setup

We will now have a look at word embeddings and how they evolve following the further pretraining we did and the code-switching pipeline we used.

We will use the tokenizer from mBERT as well as the word embedding layer of our models. Given a word, we first obtain its tokens' id using the tokenizer. For each token, we then get the corresponding embedding using the word embedding layer. Finally, for each word, we average the tokens' embedding to get a single vector per word.

Now that we can extract word embeddings using our models, we will try to see if the training we performed has an impact on word embeddings. For a given word and its translation into another language, we should expect a multilingual model to have close embeddings for the two words.

Consequently, we will now rely on the two kinds of dictionaries we built: the MaRVL dictionary (translating each test set's word into English) and the ones extracted from Panlex with part-of-speech tags. For each word in a target language and its translation into English, we

will compute their embeddings and then their cosine similarity[8]. We will average the cosine similarities for all pairs in a given dictionary. Then, by using the word embedding layer of different models, we can study the effects of our procedures on word embeddings.

For that, we will study three distinct versions of mUNITER: the baseline model, the one obtained after further pretraining on WIT (referred to as "w/ WIT pretraining"), and the final model obtained with further pretraining on WIT and code-switching for pretraining and finetuning (referred to as "w/ code-switching").

6.8.2 Results & analyses

Table 6.6 summarises the cosine similarity average for each language and model, and each dictionary.

We obtain a clear trend. Both the further pretrained model on WIT and the one obtained with code-switching have higher average cosine similarities for all dictionaries and part-of-speech tags. It means that the two steps help the models better align word embeddings of the same semantic concept. However, we can see that the average cosine similarities of MaRVL dictionaries' words are not correlated with performance. For example, Turkish has the second lowest average on MaRVL but is the language with the best results on the test set.

When looking at the Panlex dictionaries, we can see that "noun" is the part-of-speech tag with the highest average cosine similarities.

The key takeaway is that by using multimodal multilingual data from WIT to further pretrain mUNITER and then adding a code-switching procedure, we help the model better align words from a target language to English and vice versa.

6.9 Conclusion

This chapter demonstrated how the code-switching framework can improve the performance of multilingual visio-linguistic models. In fact, without this data augmentation technique, the additional pretraining on WIT did not effectively reduce the performance gap between the MaRVL baseline and its English translation. With code-switching, we were able to attain a 58% average accuracy across all languages for mUNITER and 58.3% for xUNITER, surpassing the 56.0% and 57.28% achieved by M^{3}P and UC^2, respectively. With these new accuracies, mUNITER and xUNITER are now closer to their baseline on the translated MaRVL (into English) of 62.3% and 62.4%, respectively, than to the random frontier of 50%.

In addition, we have provided key insights regarding code-switching for a visual commonsense reasoning task. First, we demonstrated that code-switching more than 50% of the test sets (MaRVL)can result in a significant performance increase. Besides, we demonstrate that

[8] $c(x,y) = \frac{x.y}{\|x\|^2.\|y\|^2}$

		MaRVL dict.	Panlex dict.				
			NOUN	VERB	ADJ	ADV	All
ID	baseline	0.263	0.296	0.191	0.255	0.192	0.271
	w/ WIT pretraining	0.265	0.298	0.193	0.256	0.190	0.272
	w/ code-switching	0.267	0.299	0.193	0.257	0.193	0.273
SW	baseline	0.184	0.208	0.150	0.168	0.133	0.189
	w/ WIT pretraining	0.186	0.208	0.151	0.168	0.133	0.190
	w/ code-switching	0.187	0.210	0.152	0.169	0.136	0.190
TA	baseline	0.152	0.208	0.173	0.199	0.176	0.200
	w/ WIT pretraining	0.155	0.209	0.175	0.201	0.178	0.201
	w/ code-switching	0.155	0.210	0.176	0.201	0.179	0.202
TR	baseline	0.189	0.266	0.195	0.238	0.180	0.249
	w/ WIT pretraining	0.191	0.267	0.196	0.239	0.179	0.250
	w/ code-switching	0.192	0.268	0.197	0.239	0.180	0.251
ZH	baseline	0.206	0.209	0.159	0.198	0.195	0.201
	w/ WIT pretraining	0.207	0.210	0.160	0.198	0.194	0.202
	w/ code-switching	0.209	0.210	0.161	0.198	0.195	0.202

Table 6.6: Average cosine similarity between translation pairs of the MaRVL and Panlex dictionaries for each language and three different mUNITER models. For the Panlex dictionary, we also segment the pairs by part-of-speech tag.

"noun" is the most essential part-of-speech tag for code-switching, and that just translating words with this grammatical property can provide a performance improvement comparable to when all part-of-speech tags are considered (nouns, verbs, adjectives, and adverbs). Finally, we establish that the further pretraining coupled with code-switching helps the model to better align concepts across languages.

Chapter 7

Conclusion

The objective of this thesis was to improve the zero-shot performance of multilingual multimodal models while utilising minimal resources. Using the recent MaRVL benchmark as a test bed, we increased the average performance across all languages of mUNITER and xUNITER by 3.8% and 2.2%, respectively.

In the first part, we investigated how the recent multilingual and multimodal WIT dataset could assist multilingual visio-linguistic pretraining. Top-performing models such as M^{3}P and UC^2 did not use it for pretraining and relied on machine-translated versions of Conceptual Captions. Nonetheless, if we wish to generalise visio-linguistic models to a large number of languages, it is critical to incorporate ready-to-use datasets that do not demand additional computational resources, such as machine translation. By further pretraining mUNITER on WIT data (resulting in a new model for each language), we were able to achieve an average improvement of 1.3% across all languages. In addition, we demonstrated that it is possible to achieve more robust and superior performance with less than a third of the total NLVR2 finetuning data. Finally, we evaluated several data configurations and determined that they do not significantly affect the results.

In the second part, we examined how applying code-switching during training can affect performances. This data augmentation technique improves the alignment of concepts across languages. Also, it could be a cost-effective solution to expand visio-linguistic models to multilingual contexts, as it relies on already-ubiquitous bilingual dictionaries, such as Panlex, which covers $5,700$ languages. Code-switching is an online framework that does not necessitate extra computations during pretraining. We started by showing that code-switching at least 50% of the words in the MaRVL test sets yielded better performances, when using mUNITER and xUNITER baseline models. Consequently, we designed a new code-switching procedure to pretrain visio-linguistic models. First, it leverages the part-of-speech tags information to provide less noisy bilingual translations. Second, it can be utilised in both directions: from a target language to English and vice versa. Therefore, it can be utilised not only for pretraining on multilingual data, but also for finetuning when the majority of available datasets are exclusively in English. Thanks to our dictionaries with part-of-speech tags,

we determined that "noun" is the most essential grammatical property for code-switching and results in identical performances to code-switching all nouns, verbs, adjectives, and adverbs. Importantly, mUNITER and xUNITER are now closer to the baseline performance on the translated version of MaRVL than to the random frontier (50%) and outperform M^3P and UC^2.

Finally, it would be interesting to start from the recent state-of-the art model $CCLM^3$ (Zeng et al., 2022). Indeed, the model achieves an average accuracy of 67.17% percent on MaRVL across languages. The model was pretrained on multilingual multimodal data obtained from the machine-translated version of Conceptual Captions (into five additional languages: German, French, Czech, Japanese, and Chinese). However, they did not rely on any code-switching frameworks for finetuning, and it could potentially boost the performance even more, by giving the model access to direct bilingual translations. Furthermore, we could expand our training procedure with WIT and code-switching to other tasks. Specifically, we could attempt to enhance the performance of models on the IGLUE benchmark (Bugliarello et al., 2022). On top of MaRVL, it includes other visio-linguistic tasks such as visual question answering and grounded natural language inference.

Bibliography

Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Dhruv Batra, and Devi Parikh. Vqa: Visual question answering, 2015. URL https://arxiv.org/abs/1505.00468.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a visual language model for few-shot learning, 2022. URL https://arxiv.org/abs/2204.14198.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020. URL https://arxiv.org/abs/2005.14165.

Emanuele Bugliarello, Ryan Cotterell, Naoaki Okazaki, and Desmond Elliott. Multimodal pretraining unmasked: A meta-analysis and a unified framework of vision-and-language berts, 2020. URL https://arxiv.org/abs/2011.15124.

Emanuele Bugliarello, Fangyu Liu, Jonas Pfeiffer, Siva Reddy, Desmond Elliott, Edoardo Maria Ponti, and Ivan Vulić. Iglue: A benchmark for transfer learning across modalities, tasks, and languages, 2022. URL https://arxiv.org/abs/2201.11732.

Liqun Chen, Zhe Gan, Yu Cheng, Linjie Li, Lawrence Carin, and Jingjing Liu. Graph optimal transport for cross-domain alignment. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1542–1553. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/chen20e.html.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning, 2019. URL https://arxiv.org/abs/1909.11740.

Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches, 2014. URL https://arxiv.org/abs/1409.1259.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale, 2019. URL https://arxiv.org/abs/1911.02116.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018. URL https://arxiv.org/abs/1810.04805.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2020. URL https://arxiv.org/abs/2010.11929.

Desmond Elliott, Stella Frank, Khalil Sima'an, and Lucia Specia. Multi30k: Multilingual english-german image descriptions, 2016. URL https://arxiv.org/abs/1605.00459.

Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding, 2016. URL https://arxiv.org/abs/1606.01847.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. URL https://arxiv.org/abs/1512.03385.

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

Jin-Hwa Kim, Jaehyun Jun, and Byoung-Tak Zhang. Bilinear attention networks, 2018. URL https://arxiv.org/abs/1805.07932.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014. URL https://arxiv.org/abs/1412.6980.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language, 2019. URL https://arxiv.org/abs/1908.03557.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2014. URL https://arxiv.org/abs/1405.0312.

Fangyu Liu, Emanuele Bugliarello, Edoardo Maria Ponti, Siva Reddy, Nigel Collier, and Desmond Elliott. Visually grounded reasoning across languages and cultures, 2021. URL https://arxiv.org/abs/2109.13238.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks, 2019. URL https://arxiv.org/abs/1908.02265.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013. URL https://arxiv.org/abs/1301.3781.

George A. Miller. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, nov 1995. ISSN 0001-0782. doi: 10.1145/219717.219748. URL https://doi.org/10.1145/219717.219748.

Minheng Ni, Haoyang Huang, Lin Su, Edward Cui, Taroon Bharti, Lijuan Wang, Jianfeng Gao, Dongdong Zhang, and Nan Duan. M3p: Learning universal representations via multitask multilingual multimodal pre-training, 2020. URL https://arxiv.org/abs/2006.02635.

Gabriel Peyré and Marco Cuturi. Computational optimal transport. 2018. doi: 10.48550/ARXIV.1803.00567. URL https://arxiv.org/abs/1803.00567.

Telmo Pires, Eva Schlinger, and Dan Garrette. How multilingual is multilingual bert?, 2019. URL https://arxiv.org/abs/1906.01502.

Di Qi, Lin Su, Jia Song, Edward Cui, Taroon Bharti, and Arun Sacheti. Imagebert: Cross-modal pre-training with large-scale weak-supervised image-text data, 2020. URL https://arxiv.org/abs/2001.07966.

Libo Qin, Minheng Ni, Yue Zhang, and Wanxiang Che. Cosda-ml: Multi-lingual code-switching data augmentation for zero-shot cross-lingual nlp, 2020. URL https://arxiv.org/abs/2006.06402.

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection, 2015. URL https://arxiv.org/abs/1506.02640.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks, 2015. URL https://arxiv.org/abs/1506.01497.

Rudolf Rosa. Plaintext wikipedia dump 2018, 2018. URL http://hdl.handle.net/11234/1-2735. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Sebastian Ruder, Ivan Vulić , and Anders Søgaard. A survey of cross-lingual word embedding models. *Journal of Artificial Intelligence Research*, 65:569–631, aug 2019. doi: 10.1613/jair.1.11640. URL https://doi.org/10.1613%2Fjair.1.11640.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge, 2014. URL https://arxiv.org/abs/1409.0575.

Shreya Shankar, Yoni Halpern, Eric Breck, James Atwood, Jimbo Wilson, and D. Sculley. No classification without representation: Assessing geodiversity issues in open data sets for the developing world, 2017. URL https://arxiv.org/abs/1711.08536.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1238. URL https://aclanthology.org/P18-1238.

Krishna Srinivasan, Karthik Raman, Jiecao Chen, Michael Bendersky, and Marc Najork. Wit: Wikipedia-based image text dataset for multimodal multilingual machine learning, 2021. URL https://arxiv.org/abs/2103.01913.

Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. A corpus for reasoning about natural language grounded in photographs, 2018. URL https://arxiv.org/abs/1811.00491.

Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers, 2019. URL https://arxiv.org/abs/1908.07490.

Shagun Uppal, Sarthak Bhagat, Devamanyu Hazarika, Navonil Majumder, Soujanya Poria, Roger Zimmermann, and Amir Zadeh. Multimodal research in vision and language: A review of current and emerging trends. *Information Fusion*, 77:149–171, 2022. ISSN 1566-2535. doi: https://doi.org/10.1016/j.inffus.2021.07.009. URL https://www.sciencedirect.com/science/article/pii/S1566253521001512.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017. URL https://arxiv.org/abs/1706.03762.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding, 2018. URL https://arxiv.org/abs/1804.07461.

Xinyi Wang, Sebastian Ruder, and Graham Neubig. Expanding pretrained models to thousands more languages via lexicon-based adaptation, 2022. URL https://arxiv.org/abs/2203.09435.

Shijie Wu and Mark Dredze. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 833–844, Hong Kong, China, November 2019a. Association for Computational Linguistics. doi: 10.18653/v1/D19-1077. URL https://aclanthology.org/D19-1077.

Shijie Wu and Mark Dredze. Beto, bentz, becas: The surprising cross-lingual effectiveness of bert, 2019b. URL https://arxiv.org/abs/1904.09077.

Jiateng Xie, Zhilin Yang, Graham Neubig, Noah A. Smith, and Jaime Carbonell. Neural cross-lingual named entity recognition with minimal resources, 2018. URL https://arxiv.org/abs/1808.09861.

Zhen Yang, Bojie Hu, Ambyera Han, Shen Huang, and Qi Ju. CSP:code-switching pre-training for neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2624–2636, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.208. URL https://aclanthology.org/2020.emnlp-main.208.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014. doi: 10.1162/tacl_a_00166. URL https://aclanthology.org/Q14-1006.

Yan Zeng, Wangchunshu Zhou, Ao Luo, and Xinsong Zhang. Cross-view language modeling: Towards unified cross-lingual cross-modal pre-training, 2022. URL https://arxiv.org/abs/2206.00621.

Mingyang Zhou, Luowei Zhou, Shuohang Wang, Yu Cheng, Linjie Li, Zhou Yu, and Jingjing Liu. Uc2: Universal cross-lingual cross-modal vision-and-language pre-training, 2021. URL https://arxiv.org/abs/2104.00332.

Appendix A

Impact of further pretraining on WIT

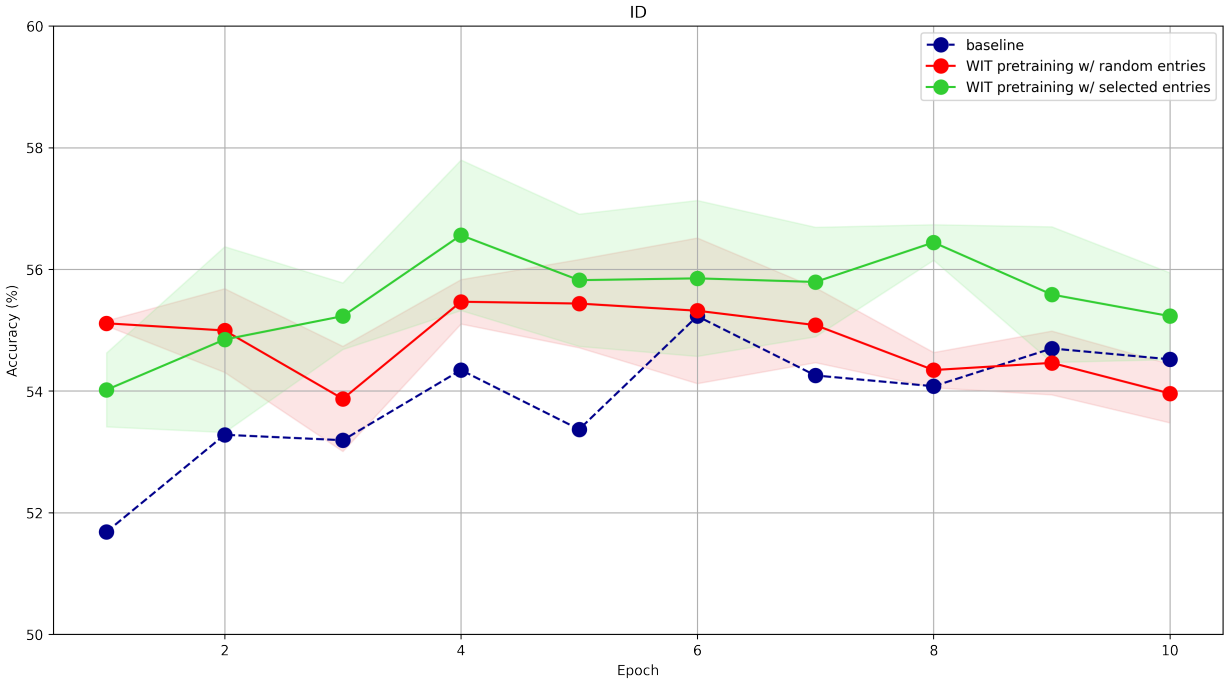

Figure A.1: mUNITER performance (accuracy) for Indonesian on the MaRVL test set vs. number of training epochs on the NLVR2 dataset.

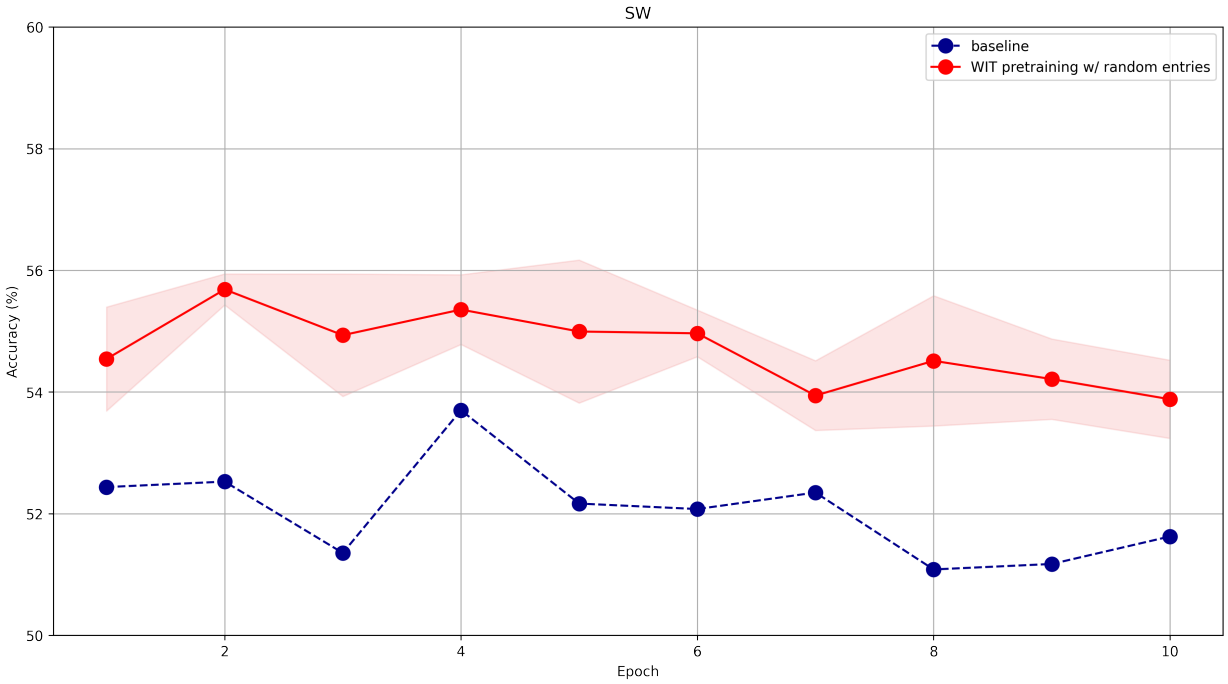

Figure A.2: mUNITER performance (accuracy) for Swahili on the MaRVL test set vs. number of training epochs on the NLVR2 dataset.

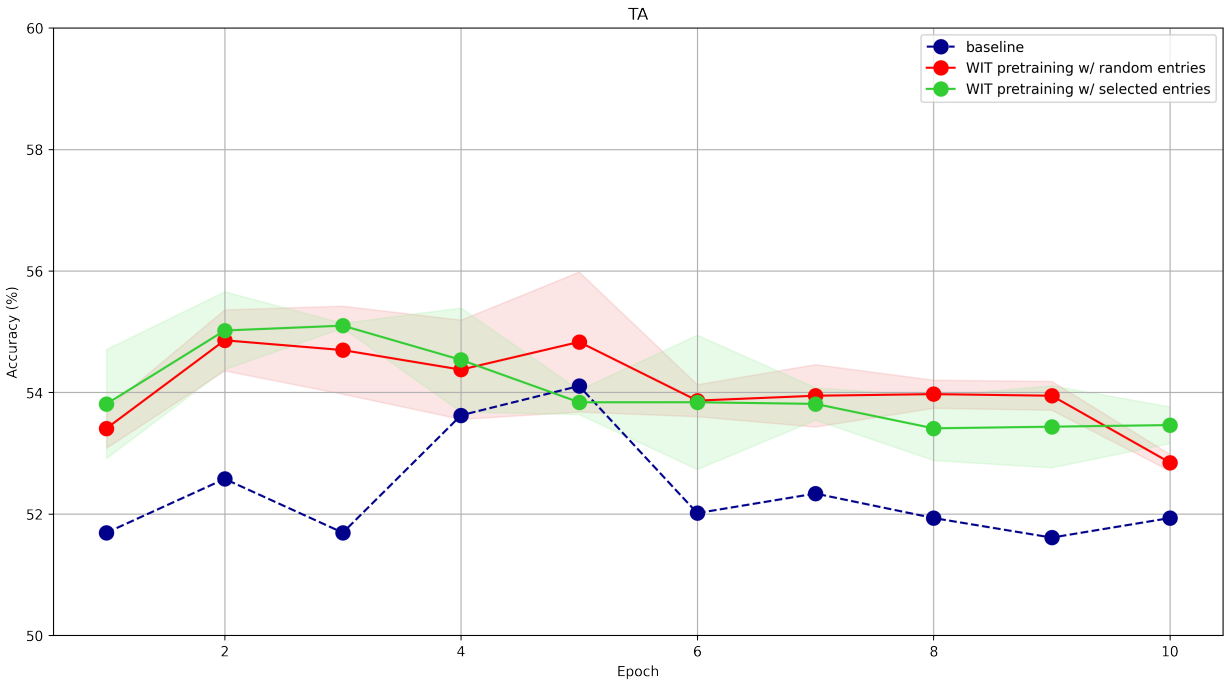

Figure A.3: mUNITER performance (accuracy) for Tamil on the MaRVL test set vs. number of training epochs on the NLVR2 dataset.

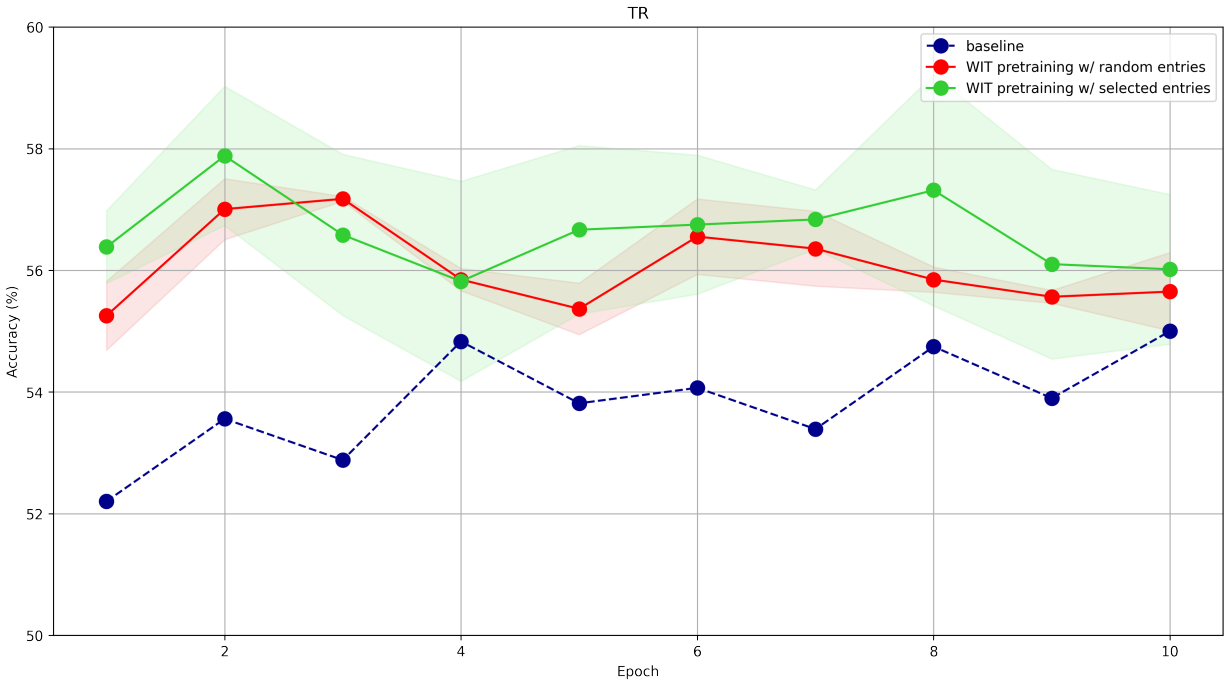

Figure A.4: mUNITER performance (accuracy) for Turkish on the MaRVL test set vs. number of training epochs on the NLVR2 dataset.

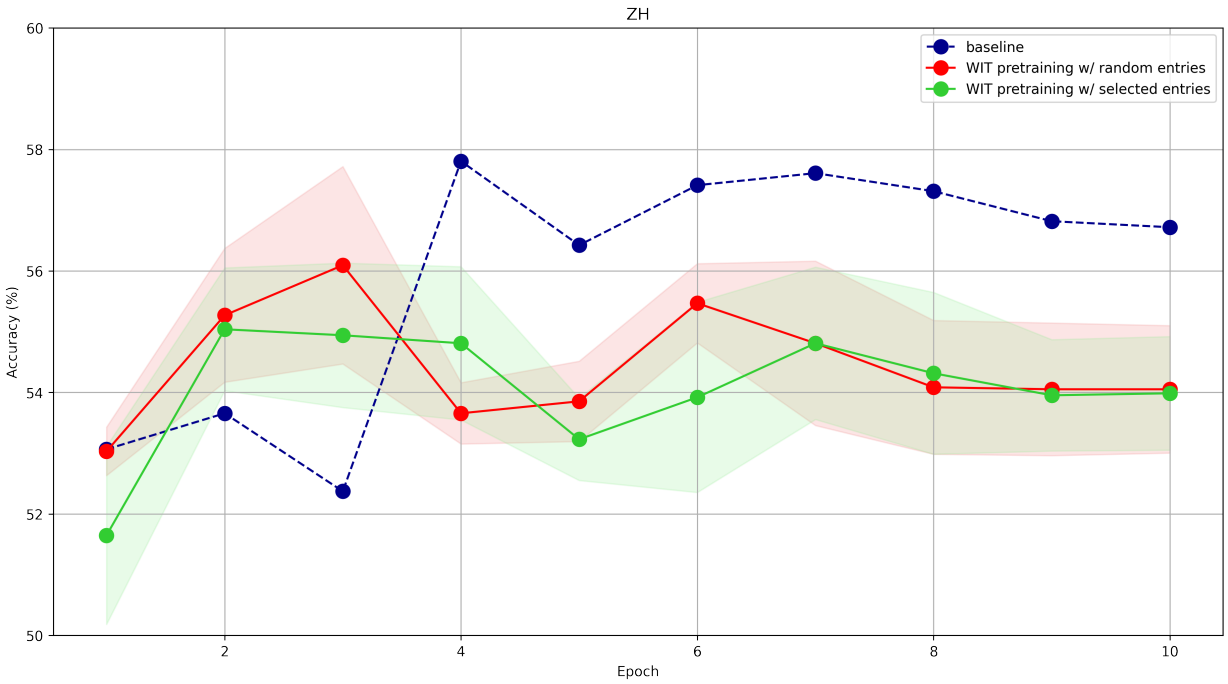

Figure A.5: mUNITER performance (accuracy) for Mandarin Chinese on the MaRVL test set vs. number of training epochs on the NLVR2 dataset.

Appendix B

Impact of the amount of finetuning data

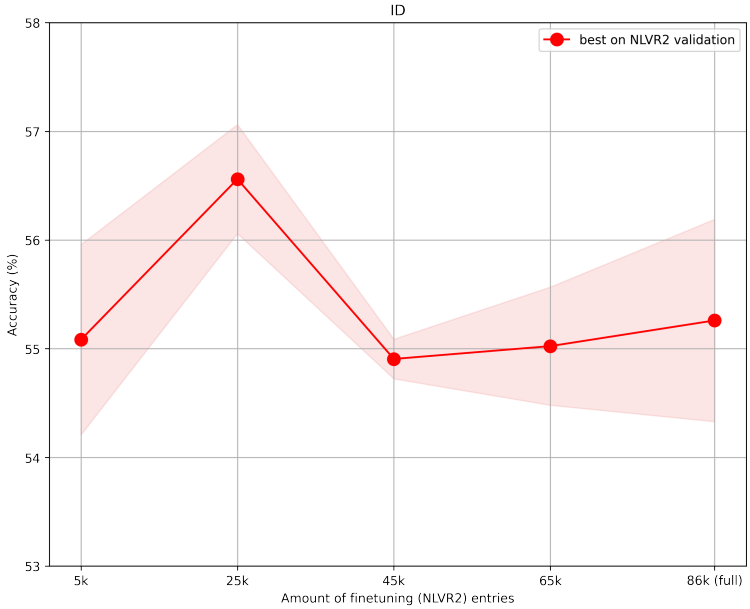

Figure B.1: mUNITER performance (accuracy) for Indonesian on the MaRVL test set vs. amount of finetuning data (NLVR2) used.

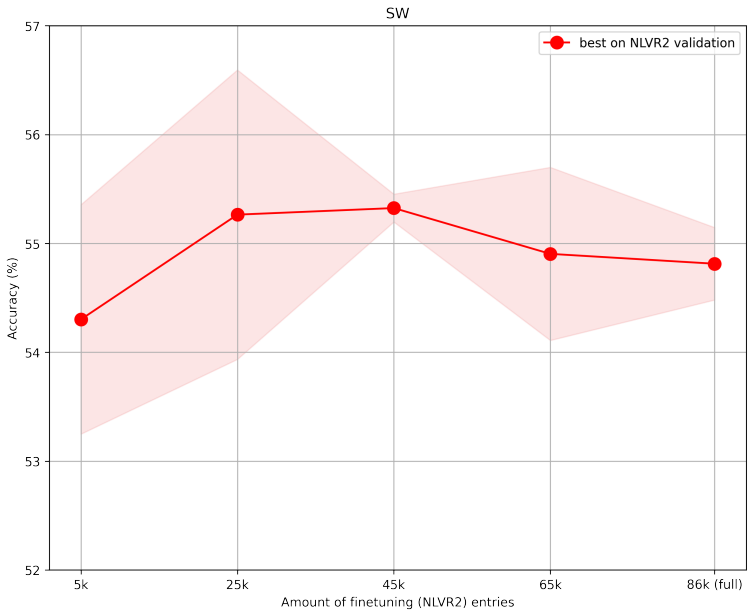

Figure B.2: mUNITER performance (accuracy) for Swahili on the MaRVL test set vs. amount of finetuning data (NLVR2) used.

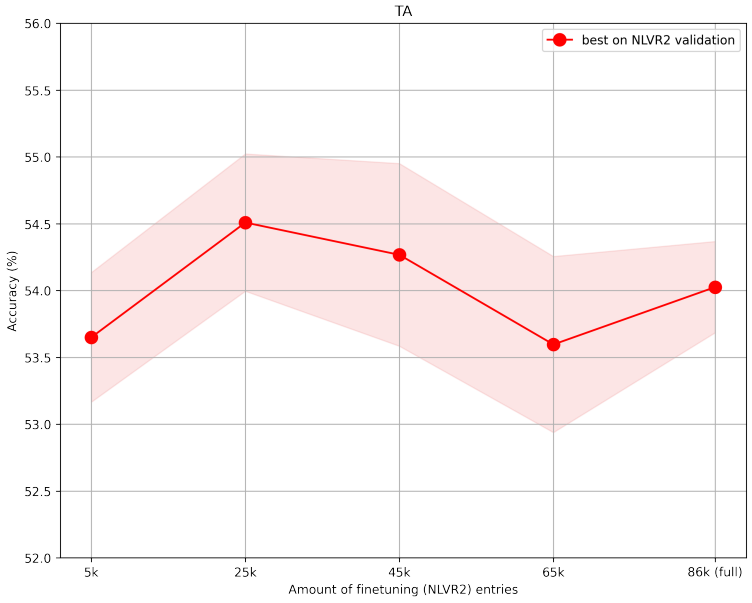

Figure B.3: mUNITER performance (accuracy) for Tamil on the MaRVL test set vs. amount of finetuning data (NLVR2) used.

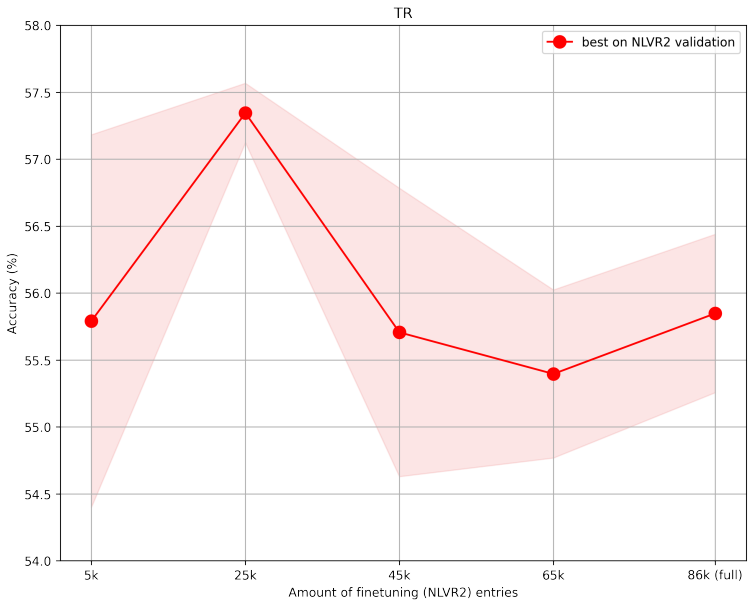

Figure B.4: mUNITER performance (accuracy) for Turkish on the MaRVL test set vs. amount of finetuning data (NLVR2) used.

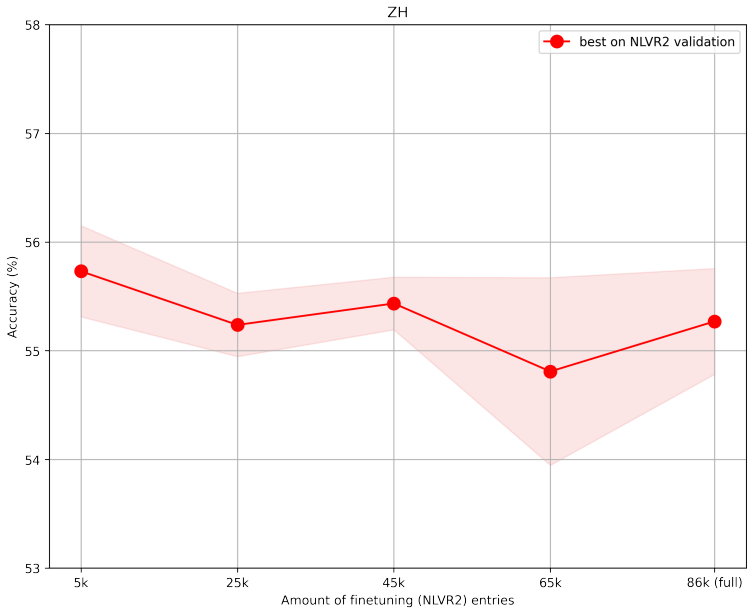

Figure B.5: mUNITER performance (accuracy) for Mandarin Chinese on the MaRVL test set vs. amount of finetuning data (NLVR2) used.

Appendix C

Impact of concepts covered during pretraining

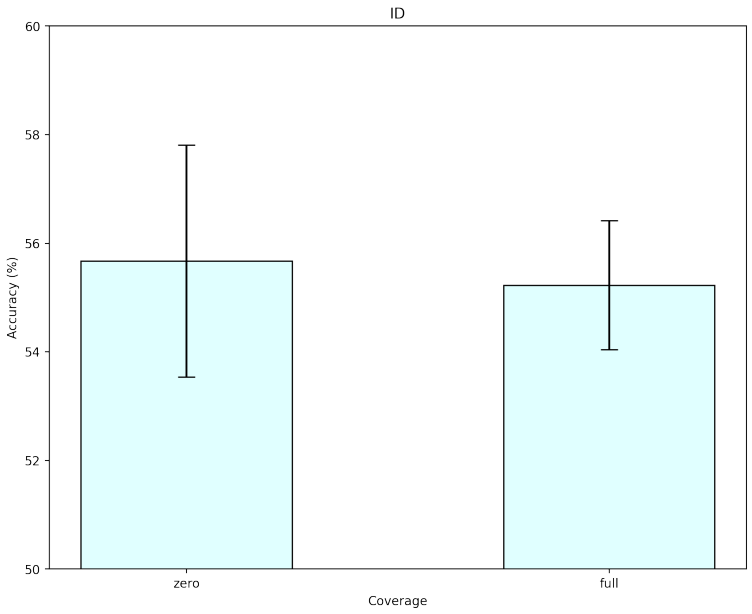

Figure C.1: mUNITER performance (accuracy) for Indonesian on the MaRVL test set for the zero and full coverage setups.

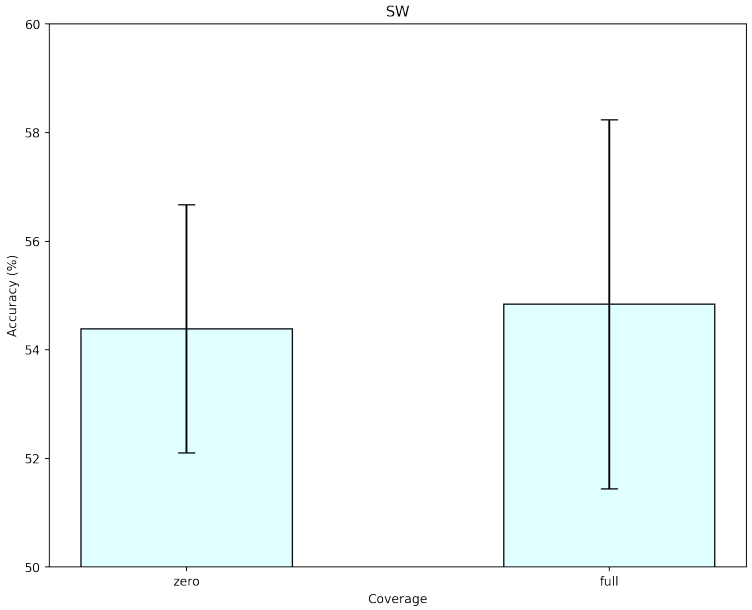

Figure C.2: mUNITER performance (accuracy) for Swahili on the MaRVL test set for the zero and full coverage setups.

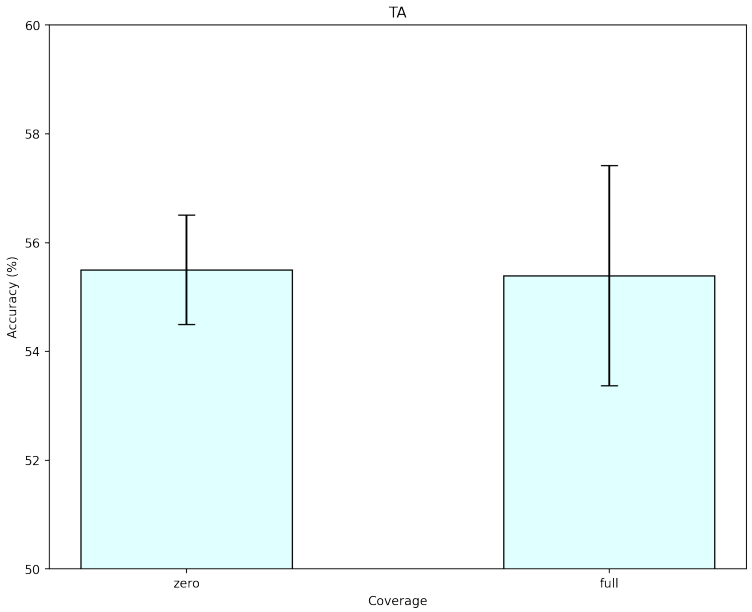

Figure C.3: mUNITER performance (accuracy) for Tamil on the MaRVL test set for the zero and full coverage setups.

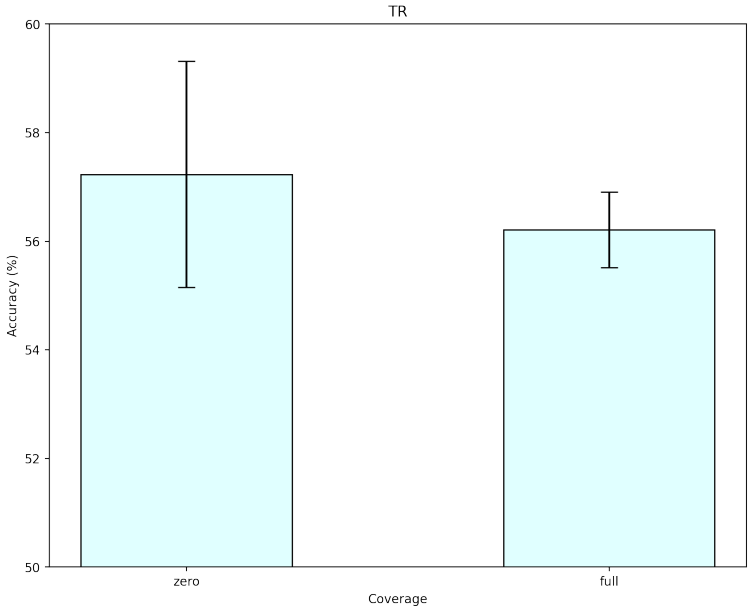

Figure C.4: mUNITER performance (accuracy) for Turkish on the MaRVL test set for the zero and full coverage setups.

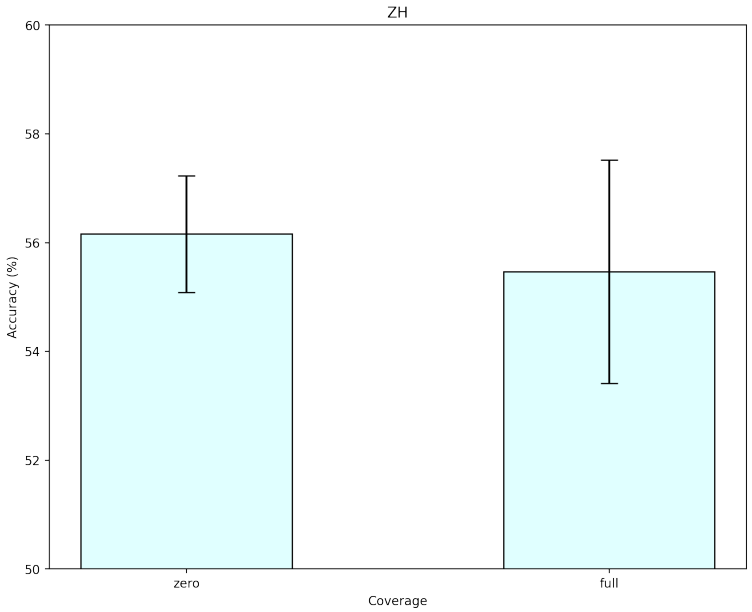

Figure C.5: mUNITER performance (accuracy) for Mandarin Chinese on the MaRVL test set for the zero and full coverage setups.

Appendix D

Code-switched training

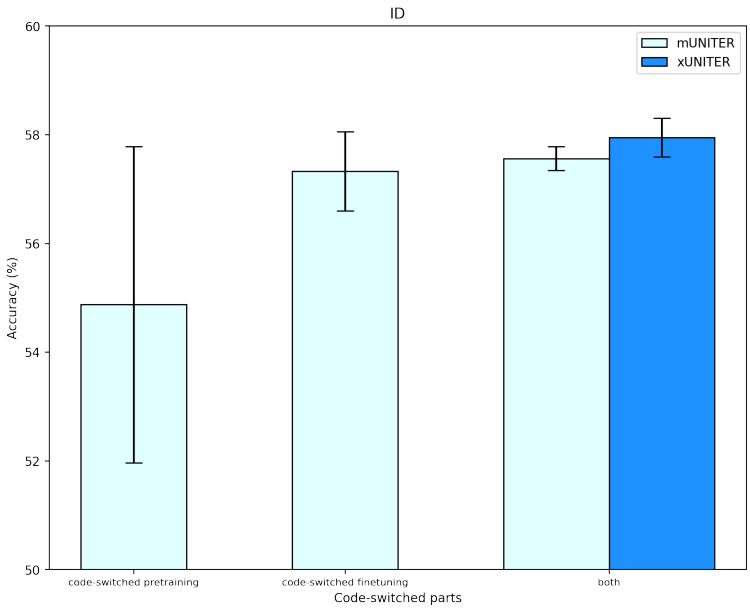

Figure D.1: Performance (accuracy) on the Indonesian MaRVL test set for mUNITER and xUNITER on MaRVL with various code-switching configurations.

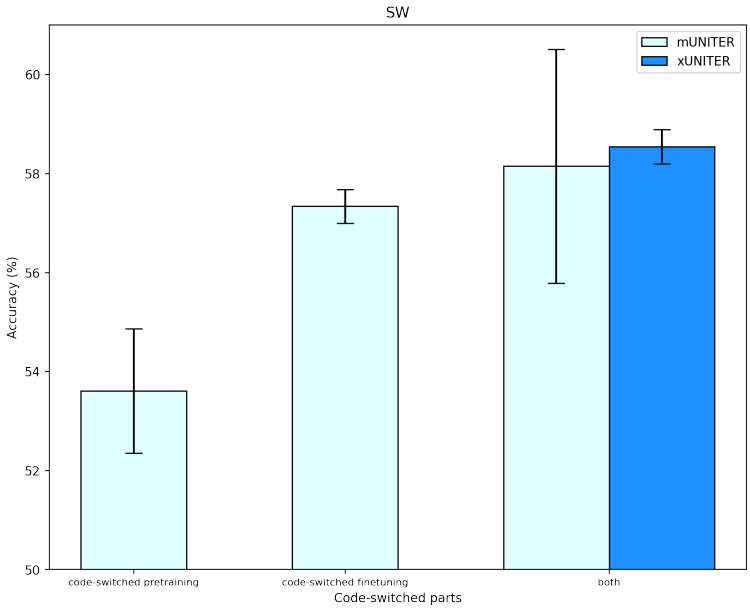

Figure D.2: Performance (accuracy) on the Swahili MaRVL test set for mUNITER and xUNITER on MaRVL with various code-switching configurations.

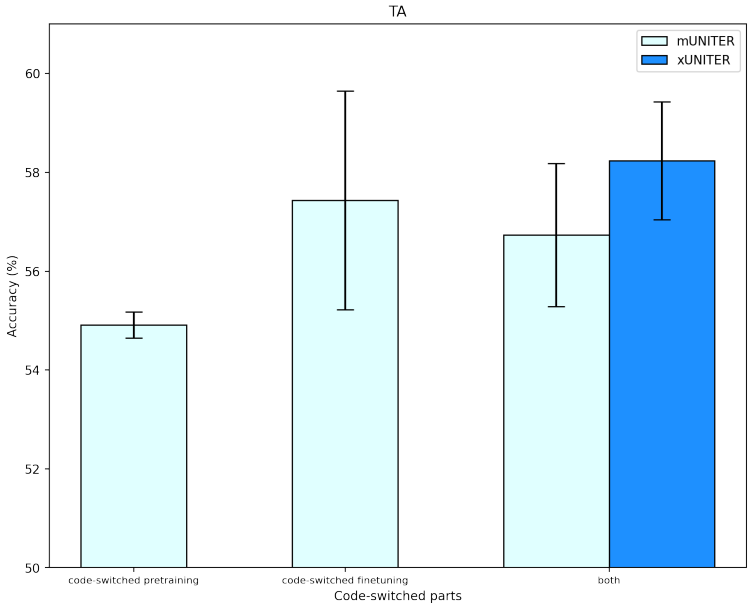

Figure D.3: Performance (accuracy) on the Tamil MaRVL test set for mUNITER and xUNITER on MaRVL with various code-switching configurations.

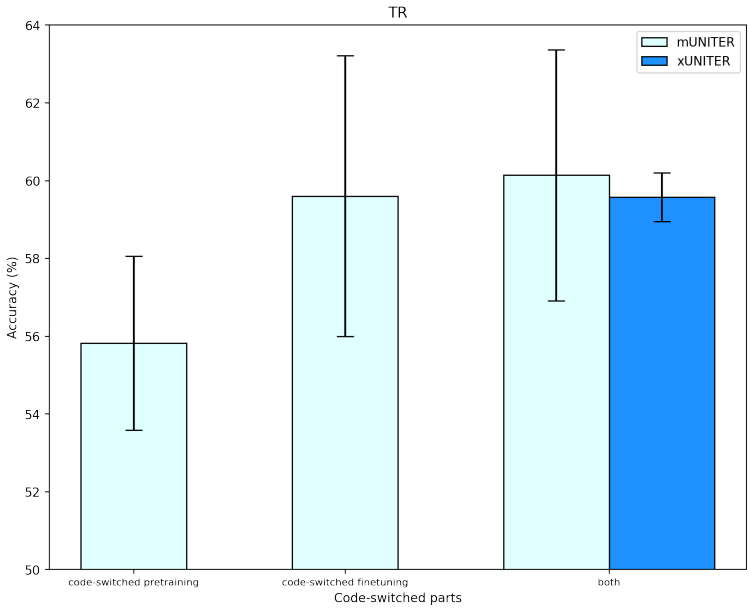

Figure D.4: Performance (accuracy) on the Turkish MaRVL test set for mUNITER and xUNITER on MaRVL with various code-switching configurations.

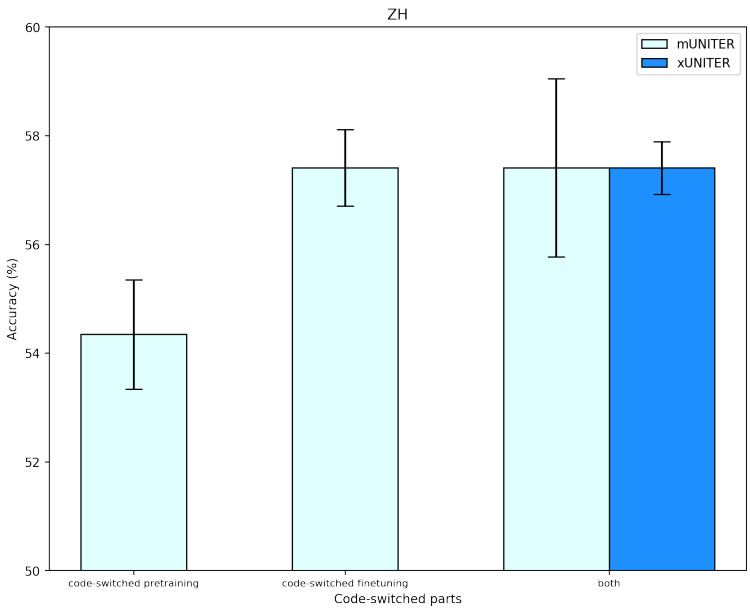

Figure D.5: Performance (accuracy) on the Mandarin Chinese MaRVL test set for mUNITER and xUNITER on MaRVL with various code-switching configurations.