MULE: Multimodal Universal Language Embedding

Donghyun Kim, Kuniaki Saito, Kate Saenko, Stan Sclaroff, Bryan A. Plummer
Boston University
{donhk, keisaito, saenko, sclaroff, bplum}@bu.edu

Abstract

Existing vision-language methods typically support two languages at a time at most. In this paper, we present a modular approach which can easily be incorporated into existing vision-language methods in order to support many languages. We accomplish this by learning a single shared *Multimodal Universal Language Embedding* (MULE) which has been visually-semantically aligned across all languages. Then we learn to relate MULE to visual data as if it were a single language. Our method is not architecture specific, unlike prior work which typically learned separate branches for each language, enabling our approach to easily be adapted to many vision-language methods and tasks. Since MULE learns a single language branch in the multimodal model, we can also scale to support many languages, and *languages with fewer annotations* can take advantage of the good representation learned from other (more abundant) language data. We demonstrate the effectiveness of our embeddings on the bidirectional image-sentence retrieval task, supporting up to four languages in a single model. In addition, we show that Machine Translation can be used for data augmentation in multilingual learning, which, combined with MULE, improves mean recall by up to 20.2% on a single language compared to prior work, with the most significant gains seen on languages with relatively few annotations. Our code is publicly available[1].

Introduction

Vision-language understanding has been an active area of research addressing many tasks such as image captioning (Fang et al. 2015; Gu et al. 2018), visual question answering (Antol et al. 2015; Goyal et al. 2017), image-sentence retrieval (Wang et al. 2019; Nam, Ha, and Kim 2017), and phrase grounding (Plummer et al. 2015; Hu et al. 2016). Recently there has been some attention paid to expanding beyond developing monolingual (typically English-only) methods by also supporting a second language in the same model (*e.g*., (Gella et al. 2017; Hitschler, Schamoni, and Riezler 2016; Rajendran et al. 2015; Calixto, Liu, and Campbell 2017; Li et al. 2019; Lan, Li, and Dong 2017)). However, these methods often learn completely separate language representations to relate to visual data, resulting in many language-specific model parameters that grow linearly with the number of supported languages.

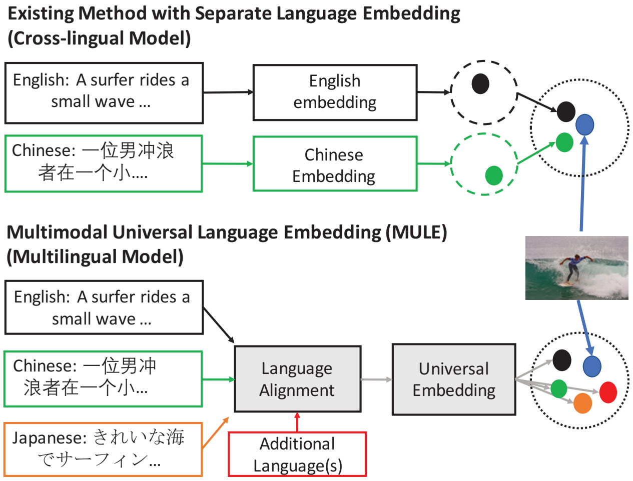

Existing Method with Separate Language Embedding
(Cross-lingual Model)
English: A surfer rides a small wave ...
English embedding
Chinese: 一位男冲浪者在一个小....
Chinese Embedding
Multimodal Universal Language Embedding (MULE)
(Multilingual Model)
English: A surfer rides a small wave ...
Chinese: 一位男冲浪者在一个小...
Japanese: きれいな海でサーフィン...
Language Alignment
Additional Language(s)
Universal Embedding

Figure 1: Most prior work on vision-language tasks supports up to two languages where each language is projected into a shared space with the visual features using its own language-specific model parameters (top). Instead, we propose MULE, a language embedding that is visually-semantically aligned across multiple languages (bottom). This enables us to share a single multimodal model, significantly decreasing the number of model parameters, while also performing better than prior work using separate language branches or multilingual embeddings which were aligned using only language data.

In this paper, we propose a Multimodal Universal Language Embedding (MULE), an embedding that has been visually-semantically aligned across many languages. Since each language is embedded into to a shared space, we can use a single task-specific multimodal model, enabling our approach to scale to support many languages. Most prior works use a vision-language model that supports at most

[1] http://cs-people.bu.edu/donhk/research/MULE.html

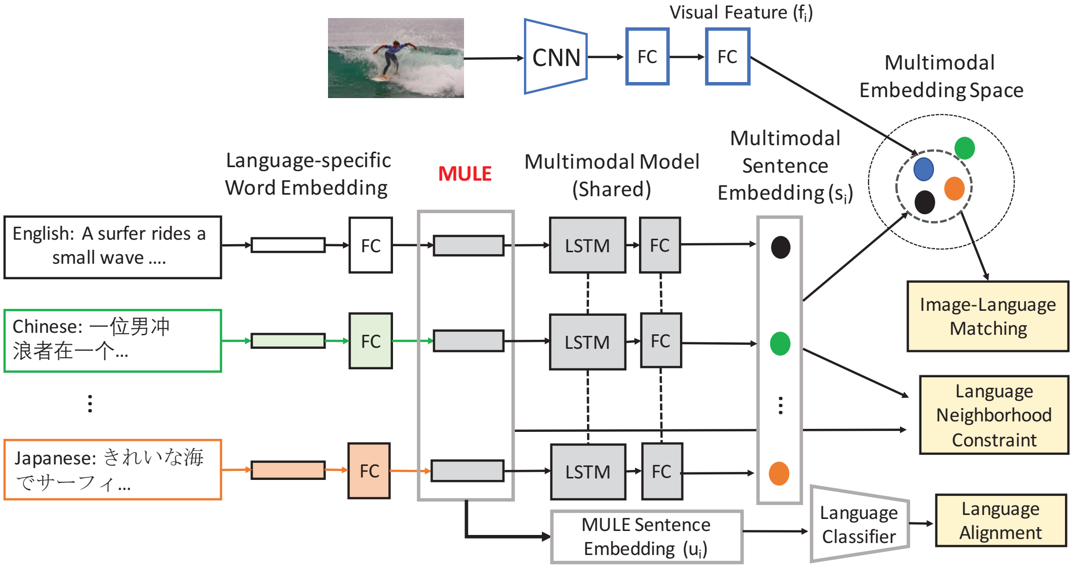

Visual Feature (f_i)
CNN
FC
FC
Multimodal Embedding Space
Language-specific Word Embedding
MULE
Multimodal Model (Shared)
Multimodal Sentence Embedding (s_i)
English: A surfer rides a small wave ....
Chinese: 一位男冲浪者在一个...
⋮
Japanese: きれいな海でサーフィ...
FC
LSTM
Image-Language Matching
Language Neighborhood Constraint
MULE Sentence Embedding (u_i)
Language Classifier
Language Alignment

Figure 2: An overview of the architecture used to train our multimodal universal language embedding (MULE). Training MULE consists of three components: neighborhood constraints which semantically aligns sentences across languages, an adversarial language classifier which encourages features from different languages to have similar distributions, and a multimodal model which helps MULE learn the visual-semantic meaning of words across languages by performing image-sentence matching.

two languages with separate language branches (*e.g.* (Gella et al. 2017)), significantly increasing the number of parameters compared to our work (see Fig. 1 for a visualization). A significant challenge of multilingual embedding learning is the considerable disparity in the availability of annotations between different languages. For English, there are many large-scale vision-language datasets to train a model such as MSCOCO (Lin et al. 2014) and Flickr30K (Young et al. 2014), but there are few datasets available in other languages, and some contain limited annotations (see Table 1 for a comparison of the multilingual datasets used to train MULE). One could simply use Neural Machine Translation (*e.g.* (Bahdanau, Cho, and Bengio 2014; Sutskever, Vinyals, and Le 2014)) to convert the sentence from the original language to a language with a trained model, but this has two significant limitations. First, machine translations are not perfect and introduce some noise, making vision-language reasoning more difficult. Second, even with a perfect translation, some information is lost going between languages. For example,"她们" is used to refer to a group of women in Chinese. However, it is translated to "they" in English, losing all gender information that could be helpful in a downstream task. Instead of fully relying on translations, we introduce a scalable approach that supports queries from many languages in a single model.

An overview of the architecture we use to train MULE is provided in Fig. 2. For each language we use a single fully-connected layer on top of each word embedding to project it into an embedding space shared between all languages, *i.e.*, our MULE features. Training our embedding consists of three components. First, we use an adversarial language classifier in order to align feature distributions between languages. Second, motivated by the sentence-level supervision used to train language embeddings (Devlin et al. 2018; Kiela et al. 2018; Lu et al. 2019), we incorporate visual-semantic information by learning how to match image-sentence pairs using a multimodal network similar to (Wang et al. 2019). Third, we ensure semantically similar sentences are embedded close to each other (referred to as neighborhood constraints in Fig. 2). Since MULE does not require changes to the architecture of the multimodal model like prior work (*e.g.*, (Gella et al. 2017)), our approach can easily be incorporated to other multimodal models.

Dataset	Language	# images	# descriptions
Multi30K	English	29K	145K
	German	29K	145K
	Czech	29K	29K
	French	29K	29K
MSCOCO	English	121K	606K
	Japanese	24K	122K
	Chinese	18K	20K

Table 1: Available data for each language during training.

Despite being trained to align languages using additional large text corpora across each supported language, our experiments will show recent multilingual embeddings like MUSE (Conneau et al. 2018) perform significantly worse on tasks like multilingual image-sentence matching than our approach. In addition, sharing all the parameters of the multimodal component of our network enables languages with fewer annotations to take advantage of the stronger representation learned using more data. Thus, as our experiments will show, MULE obtains its largest performance gains on languages with less training data. This gain is boosted further by using Neural Machine Translation as a data augmentation technique to increase the available vision-language training data.

We summarize our contributions as follows:

- We propose MULE, a multilingual text representation for vision-language tasks that can transfer and learn textual representations for low-resourced languages from label-rich languages, such as English.
- We demonstrate MULE's effectiveness on a multilingual image-sentence retrieval task, where we outperform extensions of prior work by up to 20.2% on a single language while also using fewer model parameters.
- We show that using Machine Translation is a beneficial data augmentation technique for training multilingual embeddings for vision-language tasks.

Related Work

Language Representation Learning. Word embeddings, such as Word2Vec (Mikolov, Yih, and Zweig 2013) and FastText (Bojanowski et al. 2017), play an important role in vision-language tasks. These word embeddings provide a mapping function from a word to an n-dimensional vector where semantically similar words are embedded close to each other and are typically trained using language-only data. However, recent work has demonstrated a disconnect between how these embeddings are evaluated and the needs of vision-language tasks (Burns et al. 2019). Thus, several recent methods have obtained significant performance gains across many tasks over language-only trained counterparts by learning the visual-semantic meaning of words specifically for use in vision-language problems (Kottur et al. 2016; Kiela et al. 2018; Burns et al. 2019; Lu et al. 2019; Gupta, Schwing, and Hoiem 2019; Nguyen and Okatani 2019; Tan and Bansal 2019). All these methods have addressed embedding learning only in the monolingual (English-only) setting, however, and none of the methods that align representations across many languages were designed specifically for vision-language tasks (*e.g.* (Conneau et al. 2018; Rajendran et al. 2015; Calixto, Liu, and Campbell 2017)). Thus, just as in the monolingual setting, and verified in our experiments, these multilingual, language-only trained embeddings do not generalize as well to vision-language tasks as the visually-semantically aligned multilingual embeddings in our approach.

Image-Sentence Retrieval. The goal of this task is to retrieve relevant images given a sentence query and vice versa. Although there has been considerable attention given to this task, nearly all have focused on supporting queries in a single language, which is nearly always English (*e.g.* (Nam, Ha, and Kim 2017; Wang et al. 2019)). These models tend to either learn an embedding between image and text features (*e.g.* (Plummer et al. 2015; Wang et al. 2019; Lee et al. 2018; Huang, Wu, and Wang 2018)) or sometimes directly learn a similarity function (*e.g.* (Wang et al. 2019)). Most relevant to our work is (Gella et al. 2017) who propose a cross-lingual model, which uses an image as a pivot and enforce the sentence representations from English and German to be similar to the pivot image representation, similar to the structure-preserving constraints of (Wang et al. 2019). However, in (Gella et al. 2017) each language is modeled with a completely separate language model. While this may be acceptable for modeling one or two languages, it would not scale well for representing many languages as the number of parameters would grow too large. (Wehrmann et al. 2019) proposes a character-level encoding for a cross-lingual model, which effectively reduces the size of the word embedding for languages. However, this approach shows a significant drop in performance when training for just two languages.

In this work we explore multiple languages with *underrepresented and low-resourced languages* (up to 4 languages). We learn a shared representation between all languages, enabling us to scale to many languages with few additional parameters. This enables feature sharing with low-resourced languages, resulting in significantly improved performance, even when learning many languages.

Neural Machine Translation. In Neural Machine Translation (NMT) the goal is to translate text from one language to another language with parallel text corpora (Bahdanau, Cho, and Bengio 2014; Sutskever, Vinyals, and Le 2014; Johnson et al. 2017). (Johnson et al. 2017) proposed a multilingual NMT model, which uses a single model with an encoder-decoder architecture. They observed that translation quality on low-resourced languages can be improved when trained with label-rich languages. As discussed in the Introduction, and verified in our experiments, directly using NMT for vision-language tasks has some limitations in its usefulness for vision-language tasks, but it can provide additional benefits combined with our method.

Visual-Semantic Multilingual Alignment

In this section we describe how we train MULE, a lightweight multilingual embedding which is visually-semantically aligned across many languages and can easily be incorporated into many vision-language tasks and models. Each word in some language input is encoded using a continuous vector representation, which is then projected to the shared language embedding (MULE) using a language-specific fully connected layer. In our experiments, we initialize our word embeddings from 300-dimensional monolingual FastText embeddings (Bojanowski et al. 2017). The word embeddings and these fully connected layers are the only language-specific parameters in our network. Due to their compact size, they can easily scale to a large vocabulary encompassing many languages.

To train MULE, we use paired and unpaired sentences between the languages from annotated vision-language datasets. We find that we get the best performance by first pretraining MULE with paired sentences before fine-tuning using the multimodal layers with the multi-layer neighboring constraints described in (Eq. 1) and the adversarial language classifier described below. While our experiments focus solely on utilizing multimodal data, one could also try to integrate large text corpora with annotated language pairs (*e.g*. (Conneau et al. 2018)). However, as our experiments will show, only using generic language pairs for this alignment (*i.e*., not sentences related to images) results in some loss of information that is important for vision-language reasoning. We will now discuss the three major components of our loss used to train our embedding as shown in Fig 2.

Multi-Layer Neighborhood Constraints

During training we assume we have paired sentences obtained from the vision-language annotations, *i.e*., sentences that describe the same image. These sentences are typically independently generated, so they may not refer to the same entities in the image, and when they do describe the same object they may be referenced in different ways (*e.g*., *a black dog* vs. *a Rottweiler*). However, we assume they convey the same general sentiment since they describe the same image. Thus, the multi-layer neighborhood constraints try to encourage sentences from the same image to embed near each other. These constraints are analogous to those proposed in related work on image-sentence matching (Gella et al. 2017; Wang et al. 2019), except that we apply the constraints at multiple layers of our network. Namely, we use the neighborhood constraints on the MULE layer as well as the multimodal embedding layer as done in prior work.

To obtain sentence representations in the MULE space, we simply average the features of each word, which we found to perform better than using an LSTM while increasing model efficiency (an observation also made by (Burns et al. 2019; Wang et al. 2019)). For the multimodal embedding space, we use the same features for a multimodal sentence representation that is used to relate to the image features. We denote the averaged representations in the MULE space (*i.e*. MULE sentence embeddings) as u_i and multimodal sentence embeddings as s_i as shown in Fig. 2.

The neighborhood constraints are enforced using a triplet loss function. For some specific sentence embedding s_i, where s_{i^+} and s_{i^-} denote a positive and negative pair for s_i, respectively. We use the same notation for positive and negative pairs u_{i^+} and u_{i^-}. Positive and negative pairs may be from any language. So, for example, German and Czech sentences describing the same image are all positive pairs, while any pair of sentences from different images we assume are negatives (analogous assumptions were made in (Gella et al. 2017; Wang et al. 2019)). Given a cosine distance function d, the margin-based triplet loss is to minimize with a margin m:

$$\begin{aligned}\mathcal{L}_{LM} = &\max(0, d(s_i, s_{i^+}) - d(s_i, s_{i^-}) + m) \\ &+ \max(0, d(u_i, u_{i^+}) - d(u_i, u_{i^-}) + m).\end{aligned} \quad (1)$$

Following (Wang et al. 2019), we enumerate all positive and negative pairs in a minibatch and use the top K most violated constraints, where $K = 10$ in our experiments.

Language Domain Alignment

Inspired by the domain adaptation approach of (Ganin and Lempitsky 2014; Tzeng et al. 2014), we use an adversarial language classifier (LC) to align the feature distributions of the different languages supported by our model. The goal is to project each language domain into a single shared domain, so that the model transfers knowledge between languages. This classifier does not require paired language data. We use a single fully connected layer for the LC denoted by W_{lc}. Given a MULE sentence representation u_i presented in l-th language, we first minimize the objective function w.r.t the language classifier W_{lc}:

$$\mathcal{L}_{LC}(W_{lc}, u_i, l) = CrossEntropy(W_{lc}u_i, l) \quad (2)$$

Then, in order to align the language domain, we learn language-specific parameters to maximize the loss function.

Image-Language Matching

To directly learn the visual meaning of words we also use a multimodal model to relate sentences to images which is trained along with our MULE embedding. To accomplish this, we use a two-branch network similar to that of (Wang et al. 2019), except we use the last hidden state of an LSTM to obtain a final multimodal sentence representation (s_i in Fig. 2). Although (Burns et al. 2019; Wang et al. 2019) found mean-pooled features followed by a pair of fully connected layers often perform better, we found using an LSTM to be more stable in our experiments. We also kept image representation fixed, and only the two fully connected layers after the CNN in Fig. 2 were trained.

Let f_i denote the image representation and s_i denote the sentence representation in the multimodal embedding space for the i-th image x_i. We construct a minibatch that contains positive image-sentence pairs from different images. In the batch, we get (f_i, s_i) from the image-sentence pair (x_i, y_i). It should be noted that sentences can be presented in multiple languages. We sample triplets to have negative pairs and positive pairs for image representations and sentence representations. To be specific, given f_i, we sample corresponding positive sentence representation s_{i^+} and a negative sentence representation s_{i^-} represented in the same language. Equivalently, given a y_i, we sample the positive image representation f_{i^+} and a negative image representation f_{i^-}. Then, our margin-based objective function for matching is to minimize with a margin m and a cosine distance function d:

$$\begin{aligned}\mathcal{L}_{triplet} = &\max(0, d(f_{i^+}, s_{i^+}) - d(f_{i^+}, s_{i^-}) + m) \\ &+ \max(0, d(s_{i^+}, f_{i^+}) - d(s_{i^+}, f_{i^-}) + m).\end{aligned} \quad (3)$$

As with the neighborhood constraints, the loss is computed over the $K = 10$ most violated constraints. Finally, our overall objective function is to find:

$$\begin{aligned}\hat{\theta} &= \underset{\theta}{\operatorname{argmin}}\, \lambda_1\mathcal{L}_{LM} - \lambda_2\mathcal{L}_{LC} + \lambda_3\mathcal{L}_{triplet} \\ \hat{W_{lc}} &= \underset{W_{lc}}{\operatorname{argmin}}\, \lambda_2\mathcal{L}_{LC}\end{aligned} \quad (4)$$

where θ includes all parameters in our network except for the language classifier, W_{lc} contains the parameters of the language classifier, and λ determines weights on each loss.

Experiments

Datasets

Multi30K (Elliott et al. 2016; 2017; Barrault et al. 2018). The Multi30K dataset augments Flickr30K (Young et al.

2014) with image descriptions in German, French, and Czech. Flickr30K contains 31,783 images where each image is paired with five English descriptions. There are also five sentences provided per image in German, but only one sentence per image is provided for French and Czech. French and Czech sentences are translations of their English counterparts, but German sentences were independently generated. We use the dataset's provided splits which uses 29K/1K/1K images for training/test/validation.

MSCOCO (Lin et al. 2014). MSCOCO is a large-scale dataset which contains 123,287 images and each image is paired with 5 English sentences. Although this accounts for a much larger English training set compared with Multi30K, but there are fewer annotated sentences in other languages. (Miyazaki and Shimizu 2016) released the YJ Captions 26K dataset which contains about 26K images in MSCOCO where each image is paired with independent 5 Japanese descriptions. (Li et al. 2019) provides 22,218 independent Chinese image descriptions for 20,341 images in MSCOCO. There are only about 4K image descriptions which are shared across the three languages. Thus, in this dataset, an additional challenge is the need to use unpaired language data. We randomly selected 1K images for the testing and validation sets from the images which contain descriptions across all three languages, for a total of 2K images, and used the rest for training. Since we use the different data split, it is not possible to compare directly with prior monolingual methods. We provide a fair comparison with our baseline and prior monolingual methods in the supplementary.

Machine Translations. As shown in Table 1, there is considerable disparity in the availability of annotations for training in different languages. As a way of augmenting these datasets, we use Google's online translator to generate sentences in other languages. Since the sentences in other languages are independently generated, their translations can provide additional variation in the training data. This also enables us to evaluate the effectiveness of NMT. In addition, we use these translated sentences to benchmark the performance translating languages from an unsupported language into one of the languages for which we have a trained model (*e.g.* translate a sentence from Chinese into English and perform the query using an English-trained model).

Image-Sentence Matching Results

Metrics. Performance on the image-sentence matching task is typically reported as Recall@$K = [1, 5, 10]$ for both image-to-sentence and sentence-to-image (*e.g.* as done in (Gella et al. 2017; Nam, Ha, and Kim 2017; Wang et al. 2019)), resulting in performance reported over six values per language. Results reporting performance over all the six values for each language can be found in the supplementary. In this paper, we average them to obtain an overall score (mR) for each compared method/language.

Model Architecture. We compare the following models:

- **EmbN** (Wang et al. 2019). As shown in (Burns et al. 2019), EmbN is the state-of-the-art image-sentence model when using image-level ResNet features and good language features. This model is the multimodal network in Fig. 2.
- **PARALLEL-EmbN.** This model borrows ideas from (Gella et al. 2017) to modify EmbN. Specifically, only a single image representation is trained, but it contains separate language branches.

Multi30K Discussion. We report performance on the Multi30K dataset in Table 2. The first line of Table 2(a) reports performance when training completely separate models (*i.e.* no shared parameters) for each language in the dataset. The significant discrepancy between the performance of English and German compared to Czech and French can be attributed to the differences in the number of sentences available for each language (Czech and French have 1/5th the sentences as seen in Table 1). Performance improves across all languages using the PARALLEL model in Table 2(a), demonstrating that the representation learned for the languages with more available annotations can still be leveraged to the benefit of other languages.

Table 2(b) and Table 2(c) show the the results of using multilingual embeddings, ML BERT (Devlin et al. 2018) and MUSE (Conneau et al. 2018) which learns a shared FastText-style embedding space for all supported languages. This enables us to compare against aligning languages using language-only data vs. our approach which performs a visual-semantic language alignment. Note that a single EmbN model is trained across all languages when using MUSE rather than training separate models since the embeddings are already aligned across languages. Comparing the numbers of Table 2(a) and Table 2(b), we observe that ML BERT which is a state-of-the-art method in NLP performs much worse than the monolingual FastText. In addition, we see in Table 2(c) that MUSE improves performance on low-resourced languages (*i.e.* French and Czech), but actually hurts performance on the language with more available annotations (*i.e.* English). These results indicate that some important visual-semantic knowledge is lost when relying solely on language-only data to align language embeddings and NLP method does not generalize well to the language-vision task.

Table 2(d) compares the effect that different components of MULE has on performance. Going from the last line of Table 2(a) to the first line of Table 2(d) demonstrates that using a single-shared language branch can significantly improve lower-resource language performance (*i.e.* French and Czech), with only a minor impact to performance on languages with more annotations. Comparing the last line of Table 2(c) which reports performance of our full model using MUSE embeddings, to the last line of Table 2(d), we see that using MUSE embeddings still hurts performance, which helps verify our earlier hypothesis that some important visual-semantic information is lost when aligning languages with only language data. This is also reminiscent of an observation in (Burns et al. 2019), *i.e.*, it is important to consider the visual-semantic meaning of words when learning a language embedding for vision-language tasks.

Breaking down the components of our model in the

Model	Single Model	Mean Recall			
		En	De	Fr	Cs
(a) FastText (Baseline)					
EmbN	N	**71.1**	57.9	43.4	33.4
PARALLEL-EmbN	Y	69.6	**61.6**	52.0	43.2
(b) ML BERT					
EmbN	Y	45.5	37.9	36.4	19.2
PARALLEL-EmbN	Y	60.4	51.1	42.0	29.8
(c) MUSE					
EmbN	Y	68.6	58.2	54.0	41.8
PARALLEL-EmbN	Y	69.5	59.0	51.6	40.7
EmbN+NC+LC+LP	Y	69.0	59.7	53.6	41.0
(d) MULE (Ours)					
EmbN	Y	67.5	59.2	52.5	43.9
EmbN+NC	Y	67.2	59.9	54.0	46.4
EmbN+NC+LC	Y	67.1	60.9	55.5	49.5
EmbN+NC+LC+LP (Full)	Y	68.0	61.4	**56.4**	**50.3**

Table 2: Performance comparison of different language embeddings on the image-sentence retrieval task on Multi30K. MUSE and MULE are multilingual FastText embeddings w/ and w/o visual-semantic alignment, respectively. We denote NC: multi-layer neighborhood constraints, LC: language classifier, and LP: pretraining MULE.

last three lines of Table 2(d), we show that including the multi-layer neighborhood constraints (NC), language classifier (LC), and pretraining MULE (LP) all provide significant performance improvements (a full ablation study can be found in the supplementary). In fact, they can make up for much of the lost performance on the high-resource languages when sharing a single language branch in the multimodal model, with German actually outperforming its separate language-branch counterpart. French and Czech perform even better, however, with a total improvement of 4.4% and 7.1% mean recall over our reproductions of prior work, respectively. Clearly, training multiple languages together in a single model, especially those with fewer annotations, can result in dramatic improvements to performance without having to sacrifice the performance of a single language as long as some care is taken to ensure the model learns a comparable representation between languages. Our method achieves the best performance on German, French, and Czech, while still being comparable for English.

MSCOCO Discussion. Table 3 reports results on MSCOCO. Here, the lower resource language is Chinese, while English and Japanese both have considerably more annotations (although, unlike German on Multi30K, English has considerably more annotations than Japanese on this dataset). For the most part we see similar behavior on the MSCOCO dataset that we saw on Multi30K - the lower resource languages (Chinese) performs worse overall compared to the higher resource languages, but most of the performance gap is reduced when using our full model. Overall, our formulation obtains a 5.9% improvement to mean recall over our baselines for Chinese, and also improves performance by 1.6% mean recall for Japanese. However, for English, we obtain a slight decrease in performance compared with the English-only model reported on the first line in Table 3(a).

Model	Single Model	Mean Recall		
		En	Cn	Ja
(a) FastText (Baseline)				
EmbN	N	**75.6**	55.7	69.4
PARALLEL-EmbN	Y	70.0	52.9	68.9
(b) ML BERT				
EmbN	Y	59.4	44.7	47.2
PARALLEL-EmbN	Y	57.6	57.5	62.1
(c) MULE (Ours)				
EmbN	Y	69.4	54.2	69.0
EmbN+NC	Y	69.8	56.6	69.5
EmbN+NC+LC	Y	71.3	57.9	70.3
EmbN+NC+LC+LP (Full)	Y	72.0	**58.8**	**70.5**

Table 3: Performance comparison of different language embeddings on the image-sentence retrieval task on MSCOCO.

The drop in performance on English could be due to the significant imbalance in the training data on this dataset, where more than 3/4 of the data contains only English captions. In our experiments we separated the data into three groups: English only, English-Japanese, and English-Chinese. We ensured each group was equally represented in the minibatch, which means some images containing Japanese or Chinese captions were sampled far more than many of the English-only images. This shift in the distribution of the training data may account for some of the loss of performance. We believe more sophisticated sampling strategies may help rectify these issues and re-gain the lost performance. That said, our model has significantly fewer parameters from learning a single language branch for all languages while also outperforming the PARALLEL model from prior work which learns separate language branches.

Leveraging Machine Translations

As mentioned in the introduction, an alternative for training a model to support every language would be to use Neural Machine Translation to convert a query sentence from an unsupported language into a language which there is a trained model available. We test this approach using an English-trained EmbN model whose performance is reported on the first lines of Table 2(a) and Table 3(a). For each non-English language, we use Google Translate to convert the sentence from the source language into English, then use an English EmbN model to retrieve the images in the test set.

The first row of Table 4(b) reports the results of translating non-English queries into English and using the English-only model. On the Multi30K test set we see this performs worse on each non-English language than our MULE approach, but it does outperform some of the baselines trained on human-generated captions. Similar behavior is seen on the MSCOCO data, with Chinese-translated sentences actually performing nearly as well as human-generated English sentences. In short, using translations performs better on low-resourced languages (French, Czech, and Chinese) than the baselines. These results suggest that these translated sentences are able to capture enough information from

				Multi30K				MSCOCO		
			Single	Mean Recall						
	Model	Training Data Source	Model	En	De	Fr	Cs	En	Cn	Ja
(a)	PARALLEL-EmbN	Human Generated Only (Tables 2&3)	Y	69.6	61.6	52.0	43.2	70.0	52.9	68.9
	MULE EmbN - Full	Human Generated Only (Tables 2&3)	Y	68.0	61.4	56.4	50.3	72.0	58.8	70.5
(b)	EmbN & Machine Translated Query	Human Generated English Only	Y	71.1	48.5	46.7	46.9	75.6	72.2	66.1
	EmbN	Human Generated + Machine Translations	N	**72.0**	60.3	54.8	46.3	**76.8**	71.4	73.2
	PARALLEL-EmbN	Human Generated + Machine Translations	Y	69.0	62.6	60.6	54.1	72.5	72.3	73.3
	MULE EmbN - Full	En → Others, Machine Translations Only	Y	69.3	62.1	61.5	55.5	70.1	71.6	73.7
	MULE EmbN - Full	Human Generated + Machine Translations	Y	70.3	**64.1**	**62.3**	**57.7**	73.5	**73.1**	**76.5**

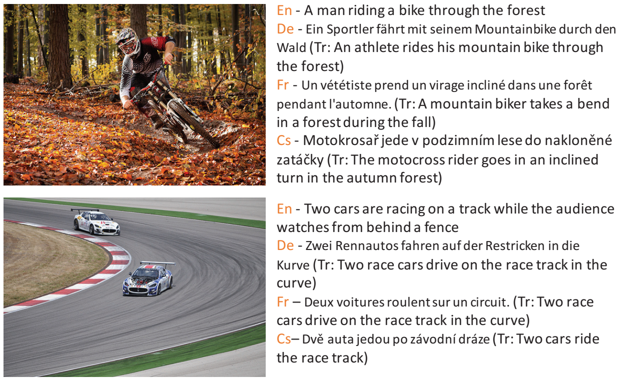
Table 4: Image-sentence matching results with Machine Translation data. We translate sentences between English and the other languages (*e.g.* En ⟷ Ja and En ⟷ Cn for MSCOCO) and augment our training set with these translations.

the original language to still provide a representation that is "good enough" to be useful.

Since translations provide a good representation for performing the retrieval task, they should also be useful in training a new model. This is especially true for any sentences that were independently generated, as they might provide a novel sentence after being translated into other languages. We report the performance of using these translated sentences to augment our training set for both datasets in Table 4(b), where our model obtains best overall performance. We observe that the models with the augmentation (*e.g.* last line of Table 4(b)) always outperform the corresponding models without the augmentation (*e.g.* last line of Table 4(a)) on all languages. On the second line of Table 4(b) we see that these translations are useful in providing more training examples even for a monolingual EmbN model. Comparing the fourth and last lines of Table 4(b) we see the difference between training the non-English languages using translated sentences alone and training with both human-generated and translated sentences. Even though the human-generated Chinese captions account for less than 5% of the total Chinese training data, we still see a significant performance improvement using them, with similar results on all other languages. This suggests that human-generated captions still provide better training data than machine translations. We also see comparing our full model to the PARALLEL-EmbN model and when using MUSE embeddings that using MULE provides performance benefits even when data is more plentiful.

Parameter Comparison

The language branch in our experiments contained 6.8M parameters. This results in 6.8M × 4 = 27.2M parameters for the PARALLEL-EmbN model proposed by (Gella et al. 2017) on Multi30K (a branch for each language). MULE uses a FC layer containing 1.7M parameters to project word features into the universal embedding, so an EmbN model for Multi30K that uses MULE would have $6.8\text{M} + 1.7\text{M} \times 4 = 13.6\text{M}$ parameters, *half the number used by (Gella et al. 2017)*. MULE also scales better with more languages than (Gella et al. 2017). ML BERT is much larger than MULE, consisting of 12 layers with ≈ 110M parameters.



En - A man riding a bike through the forest
De - Ein Sportler fährt mit seinem Mountainbike durch den Wald (Tr: An athlete rides his mountain bike through the forest)
Fr - Un vététiste prend un virage incliné dans une forêt pendant l'automne. (Tr: A mountain biker takes a bend in a forest during the fall)
Cs - Motokrosař jede v podzimním lese do nakloněné zatáčky (Tr: The motocross rider goes in an inclined turn in the autumn forest)

En - Two cars are racing on a track while the audience watches from behind a fence
De - Zwei Rennautos fahren auf der Restricken in die Kurve (Tr: Two race cars drive on the race track in the curve)
Fr – Deux voitures roulent sur un circuit. (Tr: Two race cars drive on the race track in the curve)
Cs– Dvě auta jedou po závodní dráze (Tr: Two cars ride the race track)

Figure 3: Examples of image-sentence matching results. Given an image, we pick the closest sentences on Multi30K.

Qualitative Results

Fig. 3 shows the qualitative results on our full model. We pick the two samples and retrieve the closest sentences given an image for each language on Multi30K. For other languages, we provide English translations using Google Translate. The top example shows the perfect matching between the languages. The bottom image shows that the model overestimates contextual information from the image in the English sentence. It captures not only the correct event (car racing) but also wrong objects not presented in the image (audience and fence). This sentence came from similar images with minor differences in the test set. However, the minor differences in images can be important for matching between similar images. Learning how to accurately capture the details of an image may improve the performance in future work. More results can be found in supplementary.

Conclusion

We investigated bidirectional image-sentence retrieval in a multilingual setting. We proposed MULE, which can handle multiple language queries with negligible language-specific parameters unlike prior work which learned completely distinct representations for each language. In addition to being more scalable, our method enables the model to transfer knowledge between languages, resulting in especially good performance on lower-resource languages. In addition, in order to overcome limited annotations, we show that leveraging Neural Machine Translation to augment a training dataset can significantly increase performance for training both a multilingual network as well as monolingual model. Although our work primarily focused on image-sentence retrieval, our approach is modular and can be easily incorporated into many other vision-language models and tasks.

Acknowledgements

This work is supported in part by Honda and by DARPA and NSF awards IIS-1724237, CNS-1629700, CCF-1723379.

References

Antol, S.; Agrawal, A.; Lu, J.; Mitchell, M.; Batra, D.; Zitnick, C. L.; and Parikh, D. 2015. VQA: Visual Question Answering. In *ICCV*.

Bahdanau, D.; Cho, K.; and Bengio, Y. 2014. Neural machine translation by jointly learning to align and translate. *arXiv:1409.0473*.

Barrault, L.; Bougares, F.; Specia, L.; Lala, C.; Elliott, D.; and Frank, S. 2018. Findings of the third shared task on multimodal machine translation. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*.

Bojanowski, P.; Grave, E.; Joulin, A.; and Mikolov, T. 2017. Enriching word vectors with subword information. *TACL* 5:135–146.

Burns, A.; Tan, R.; Saenko, K.; Sclaroff, S.; and Plummer, B. A. 2019. Language features matter: Effective language representations for vision-language tasks. In *ICCV*.

Calixto, I.; Liu, Q.; and Campbell, N. 2017. Multilingual multi-modal embeddings for natural language processing. *arXiv:1702.01101*.

Conneau, A.; Lample, G.; Ranzato, M.; Denoyer, L.; and Jégou, H. 2018. Word translation without parallel data. In *ICLR*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. In *arXiv:1810.04805v1*.

Elliott, D.; Frank, S.; Sima'an, K.; and Specia, L. 2016. Multi30k: Multilingual english-german image descriptions. *arXiv:1605.00459*.

Elliott, D.; Frank, S.; Barrault, L.; Bougares, F.; and Specia, L. 2017. Findings of the second shared task on multimodal machine translation and multilingual image description. *arXiv:1710.07177*.

Fang, H.; Gupta, S.; Iandola, F.; Srivastava, R. K.; Deng, L.; Dollár, P.; Gao, J.; He, X.; Mitchell, M.; Platt, J. C.; et al. 2015. From captions to visual concepts and back. In *CVPR*.

Ganin, Y., and Lempitsky, V. 2014. Unsupervised domain adaptation by backpropagation. *arXiv:1409.7495*.

Gella, S.; Sennrich, R.; Keller, F.; and Lapata, M. 2017. Image pivoting for learning multilingual multimodal representations. In *EMNLP*.

Goyal, Y.; Khot, T.; Summers-Stay, D.; Batra, D.; and Parikh, D. 2017. Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In *CVPR*.

Gu, J.; Joty, S.; Cai, J.; and Wang, G. 2018. Unpaired image captioning by language pivoting. In *ECCV*.

Gupta, T.; Schwing, A.; and Hoiem, D. 2019. Vico: Word embeddings from visual co-occurrences. In *ICCV*.

Hitschler, J.; Schamoni, S.; and Riezler, S. 2016. Multimodal pivots for image caption translation. In *ACL*.

Hu, R.; Xu, H.; Rohrbach, M.; Feng, J.; Saenko, K.; and Darrell, T. 2016. Natural language object retrieval. In *CVPR*.

Huang, Y.; Wu, Q.; and Wang, L. 2018. Learning semantic concepts and order for image and sentence matching. In *CVPR*.

Johnson, M.; Schuster, M.; Le, Q. V.; Krikun, M.; Wu, Y.; Chen, Z.; Thorat, N.; Viégas, F.; Wattenberg, M.; Corrado, G.; et al. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *TACL* 5:339–351.

Kiela, D.; Conneau, A.; Jabri, A.; and Nickel, M. 2018. Learning visually grounded sentence representations. In *NAACL-HLT*.

Kottur, S.; Vedantam, R.; Moura, J. M. F.; and Parikh, D. 2016. Visual word2vec (vis-w2v): Learning visually grounded word embeddings using abstract scenes. In *CVPR*.

Lan, W.; Li, X.; and Dong, J. 2017. Fluency-guided cross-lingual image captioning. In *ACM-MM*.

Lee, K.-H.; Chen, X.; Hua, G.; Hu, H.; and He, X. 2018. Stacked cross attention for image-text matching. In *ECCV*.

Li, X.; Xu, C.; Wang, X.; Lan, W.; Jia, Z.; Yang, G.; and Xu, J. 2019. Coco-cn for cross-lingual image tagging, captioning and retrieval. *Transactions on Multimedia*.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *ECCV*.

Lu, J.; Batra, D.; Parikh, D.; and Lee, S. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *arXiv:1908.02265*.

Mikolov, T.; Yih, W.-t.; and Zweig, G. 2013. Linguistic regularities in continuous space word representations. In *NAACL-HLT*.

Miyazaki, T., and Shimizu, N. 2016. Cross-lingual image caption generation. In *ACL*.

Nam, H.; Ha, J.-W.; and Kim, J. 2017. Dual attention networks for multimodal reasoning and matching. In *CVPR*.

Nguyen, D.-K., and Okatani, T. 2019. Multi-task learning of hierarchical vision-language representation. In *CVPR*.

Plummer, B. A.; Wang, L.; Cervantes, C. M.; Caicedo, J. C.; Hockenmaier, J.; and Lazebnik, S. 2015. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*.

Rajendran, J.; Khapra, M. M.; Chandar, S.; and Ravindran, B. 2015. Bridge correlational neural networks for multilingual multimodal representation learning. *arXiv:1510.03519*.

Sutskever, I.; Vinyals, O.; and Le, Q. V. 2014. Sequence to sequence learning with neural networks. In *NeurIPS*.

Tan, H., and Bansal, M. 2019. Lxmert: Learning cross-modality encoder representations from transformers. In *EMNLP*.

Tzeng, E.; Hoffman, J.; Zhang, N.; Saenko, K.; and Darrell, T. 2014. Deep domain confusion: Maximizing for domain invariance. *arXiv:1412.3474*.

Wang, L.; Li, Y.; Huang, J.; and Lazebnik, S. 2019. Learning two-branch neural networks for image-text matching tasks. *TPAMI* 41(2):394–407.

Wehrmann, J.; Souza, D. M.; Lopes, M. A.; and Barros, R. C. 2019. Language-agnostic visual-semantic embeddings. In *ICCV*.

Young, P.; Lai, A.; Hodosh, M.; and Hockenmaier, J. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *TACL* 2:67–78.