Cross-lingual Visual Pre-training for Multimodal Machine Translation

Ozan Caglayan¹, Menekse Kuyu², Mustafa Sercan Amac², Pranava Madhyastha¹
Erkut Erdem², Aykut Erdem³ and Lucia Specia¹,4,5

Imperial College London¹, Hacettepe University², Koç University³
University of Sheffield⁴, ADAPT - Dublin City University⁵

o.caglayan@ic.ac.uk, meneksekuyu@gmail.com, sercanamac@gmail.com, pranava@ic.ac.uk
erkut@cs.hacettepe.edu.tr, aerdem@ku.edu.tr, l.specia@ic.ac.uk

Abstract

Pre-trained language models have been shown to improve performance in many natural language tasks substantially. Although the early focus of such models was single language pre-training, recent advances have resulted in cross-lingual and visual pre-training methods. In this paper, we combine these two approaches to learn visually-grounded cross-lingual representations. Specifically, we extend the translation language modelling (Lample and Conneau, 2019) with masked region classification and perform pre-training with three-way parallel vision & language corpora. We show that when fine-tuned for multimodal machine translation, these models obtain state-of-the-art performance. We also provide qualitative insights into the usefulness of the learned grounded representations.

1 Introduction

Pre-trained language models (Peters et al., 2018; Devlin et al., 2019) have been proven valuable tools for contextual representation extraction. Many studies have shown their effectiveness in discovering linguistic structures (Tenney et al., 2019), which is useful for a wide variety of NLP tasks (Talmor et al., 2019; Kondratyuk and Straka, 2019; Petroni et al., 2019). These positive results led to further exploration of (i) cross-lingual pre-training (Lample and Conneau, 2019; Conneau et al., 2020; Wang et al., 2020) through the use of multiple mono-lingual and parallel resources, and (ii) visual pre-training where large-scale image captioning corpora are used to induce grounded vision & language representations (Lu et al., 2019; Tan and Bansal, 2019; Li et al., 2020a; Su et al., 2020; Li et al., 2020b). The latter is usually achieved by extending the masked language modelling (MLM) objective (Devlin et al., 2019) with auxiliary vision & language tasks such as masked region classification and image sentence matching.

In this paper, we present the first attempt to bring together cross-lingual and visual pre-training. Our visual translation language modelling (VTLM) objective combines the translation language modelling (TLM) (Lample and Conneau, 2019) with masked region classification (MRC) (Chen et al., 2020; Su et al., 2020) to learn grounded cross-lingual representations. Unlike most of the prior work that use classification or retrieval based downstream evaluation, we focus on the generative task of multimodal machine translation (MMT), where images accompany captions during translation (Sulubacak et al., 2020). Once pre-trained, we transfer the VTLM encoder to a Transformer-based (Vaswani et al., 2017) MMT and fine-tune it for the MMT task. To our knowledge, this is also the first attempt of pre-training & fine-tuning for MMT, where the current state of the art mostly relies on training multimodal sequence-to-sequence systems from scratch (Calixto et al., 2016; Caglayan et al., 2016; Libovický and Helcl, 2017; Elliott and Kádár, 2017; Caglayan et al., 2017; Yin et al., 2020).

Our findings highlight the effectiveness of cross-lingual visual pre-training: when fine-tuned on the English→German direction of the Multi30k dataset (Elliott et al., 2016), our MMT model surpasses our constrained MMT baseline by about 10 BLEU and 8 METEOR points. The rest of the paper is organised as follows: §2 describes our pre-training and fine-tuning protocol. §3 presents our quantitative and qualitative analyses, and §4 concludes the paper with pointers for future work.

2 Method

We propose Visual Translation Language Modelling (VTLM) objective to learn multimodal cross-lingual representations. In what follows, we first describe the TLM objective (Lample and Conneau, 2019) and then introduce the modifications required to extend it to VTLM.
2.1 Translation language modelling

The TLM objective is based on Transformer networks and assumes the availability of parallel corpora during training. It defines the input $x$ as the concatenation of $m$-length source language sentence $s_{1:m}^{(1)}$ and $n$-length target language sentence $s_{1:n}^{(2)}$:

$$x = [s_{1}^{(1)}, \ldots, s_{m}^{(1)}, s_{1}^{(2)}, \ldots, s_{n}^{(2)}]$$

For a given input, TLM follows (Devlin et al., 2019), and selects a random set of input tokens $y = \{s_{1}^{(l)}, \ldots, s_{k}^{(l)}\}$ for masking. Let us denote the masked input sequence with $\tilde{x}$, and the ground-truth targets for masked positions with $\tilde{y}$. TLM employs the masked language modelling (MLM) objective to maximise the log-probability of correct labels $\tilde{y}$, conditioned on the masked input $\tilde{x}$:

$$\mathcal{L} = \frac{1}{|x|} \sum_{x \in X} \log \Pr(\tilde{y}|\tilde{x}; \theta)$$

where $\theta$ are the model parameters. We keep the standard hyper-parameters for masking, i.e. 15% of inputs are randomly selected for masking, from which 80% are replaced with the [MASK] token, 10% are replaced with random tokens from the vocabulary, and 10% are left intact.

2.2 Visual translation language modelling

VTLM extends the TLM by adding the visual modality alongside the translation pairs (Figure 1). Therefore, we assume the availability of sentence pair & image triplets and redefine the input as:

$$x = [s_{1}^{(1)}, \ldots, s_{m}^{(1)}, s_{1}^{(2)}, \ldots, s_{n}^{(2)}, v_{1}, \ldots, v_{o}]$$

where $\{v_{1}, \ldots, v_{o}\}$ are features extracted from a Faster R-CNN model (Ren et al., 2015) pre-trained on the Open Images dataset (Kuznetsova et al., 2018). Specifically, we extract convolutional feature maps from $o = 36$ most confident regions, and average pool each of them to obtain a region-specific feature vector $v_{i} \in \mathbb{R}^{1536}$. Each region $i$ is also associated with a detection label $\hat{v}_{i}$ provided by the extractor. Before encoding, the feature vectors and their bounding box coordinates are projected into the language embedding space.

The final model processes translation pairs and projected region features in a single-stream fashion (Su et al., 2020; Li et al., 2020a), and combines the TLM loss with the masked region classification (MRC) loss as follows:

$$\mathcal{L} = \frac{1}{|X|} \sum_{x \in X} \log \Pr(\tilde{y}, \hat{v} | \tilde{x}; \theta)$$

Masking. 15% random masking ratio is applied separately to both language and visual streams, and the $\hat{v}$ above now denotes the correct region labels for the masked feature positions. Different from previous work that zeroes out masked regions (Tan and Bansal, 2019; Su et al., 2020), VTLM replaces their projected feature vectors with the [MASK] token embedding. Similar to textual masking, 10% of the random masking amounts to using regional features randomly sampled from all images in the batch, and the remaining 10% of regions are left intact.

2.3 Pre-training

VTLM requires a three-way parallel multimodal corpus, which does not exist in large-scale. To ad-

footnotes:

1The “faster rcnn inception resnet v2 atrous oid v4” model from TensorFlow.

2Although this choice is mostly practical, we hypothesise that using the same signal for both language and visual masking can be beneficial for grounding.
dress this, we extend\footnote{https://hucvl.github.io/VTLM} the Conceptual Captions (CC) (Sharma et al., 2018) dataset with German translations. CC is a large-scale collection of \( \sim 3.3 \text{M} \) images retrieved from the Internet, with noisy alt-text captions in English. The translation of English captions into German was automatically performed using an existing NMT model (Ng et al., 2019) provided\footnote{The transformer.wmt19.en-de model.} in the Fairseq (Ott et al., 2019) toolkit. Since some of the images are no longer accessible, the final corpus’ size is reduced to \( \sim 3.1 \text{M} \) triplets. We used byte pair encoding (BPE) (Sennrich et al., 2016) to learn a joint 50k BPE model on the CC dataset. The pre-training was conducted for 1.5M steps, using a single RTX2080-Ti GPU, and best checkpoints were selected with respect to validation set accuracy.

**Settings.** We use a small version of the TLM (Lample and Conneau, 2019)\footnote{https://github.com/facebookresearch/XLM} and set the model dimension, feed-forward layer dimension, number of layers and number of attention heads to \( d = 512, f = 2048, l = 6 \) and \( h = 8 \), respectively. We randomly initialise model parameters, instead of using pre-trained LM checkpoints such as BERT or XLM. We use Adam (Kingma and Ba, 2014) with the mini-batch size and the learning rate set to 64 and 0.0001, respectively. The dropout (Srivastava et al., 2014) rate is set to 0.1 in all layers. The pre-training is done for 1.5M steps using a single RTX2080-Ti GPU, and best checkpoints are selected with respect to validation accuracy.

### 2.4 Baseline MT models and fine-tuning

Our experimental protocol consists of initialising the encoder and the decoder of Transformer-based NMT and MMT models with weights from TLM/VTLM, and fine-tuning them with a smaller learning rate. The architectural difference between the NMT and the MMT models is that the latter encodes 36 regional visual features as part of the source sequence, similar to the VTLM (§ 2.2). As a natural baseline, we train constrained (trained only on the MT dataset) models without transferring weights from the pre-trained TLM/VTLM models. We refer to these models as **from-scratch**. For the fine-tuning experiments, we train three runs with different seeds. For evaluation, we use the models with the lowest validation set perplexity to decode translations with beam size equal to 8.

**Dataset.** We use the standard MMT corpus **Multi30k** (Elliott et al., 2016) for both fine-tuning and from-scratch runs. It contains 30k image descriptions from Flickr30k (Young et al., 2014) and their human translations in German for training, along with three test sets of 1K samples each: the original and the most in-domain **2016** test set, as well as **2017** and **COCO** test sets created using images and descriptions collected from sources other than Flickr.

**Settings.** For fine-tuning, we use the same hyper-parameters as the pre-training phase, apart from decreasing the learning rate to \( 1e^{-5} \). For MT models that are trained from scratch, we increase the dropout rate to 0.4 and linearly warm up the learning rate from \( 1e^{-7} \) to \( 1e^{-4} \) during the first 4,000 iterations. Inverse square-root annealing is applied after 4,000 iterations.

### 3 Results

#### 3.1 Machine translation

Table 1 reports METEOR and BLEU scores across three different test sets of Multi30k. First, we observe that the MMT system trained from scratch is consistently worse than its NMT counterpart. However, the gap disappears when pre-trained TLM/VTLM checkpoints are fine-tuned for MT. This suggests that pre-training may be necessary for **single-stream** multimodal encoding, where the number of regions (36) outnumbers the avg. number of source tokens (13 for Multi30k).

Second, we see that the best performances are obtained when models are first pre-trained on the **three-way** parallel Conceptual Captions (CC) dataset. To validate this further, we train a baseline NMT on the concatenation of Multi30k and CC (NMT+CC) and an MMT that uses only Multi30k for both pre-training and fine-tuning. The results clearly show that these systems lag behind the ones pre-trained on CC.

We also experimented with an alternative pre-training strategy where we still have the MRC task to predict the object labels of randomly selected visual regions but the input regional features for those positions are not actually replaced with \([\text{MASK}])\footnote{In other words, we let the model predict the object labels using a cross-lingual and multimodal input where only input words are randomly masked. Interestingly, this alternative MMT (Table 1) reveals that not masking visual regions during pre-training}. In other words, we let the model predict the object labels using a cross-lingual and multimodal input where only input words are randomly masked. Interestingly, this alternative MMT (Table 1) reveals that not masking visual regions during pre-training...
<table>
<thead>
<tr>
<th></th>
<th>2016 METEOR</th>
<th>2016 BLEU</th>
<th>2017 METEOR</th>
<th>2017 BLEU</th>
<th>COCO METEOR</th>
<th>COCO BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Best RNN-MMT (Caglayan, 2019)</td>
<td>58.7</td>
<td>39.4</td>
<td>52.9</td>
<td>32.6</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Graph-based Transformers MMT (Yin et al., 2020)</td>
<td>57.6</td>
<td>39.8</td>
<td>51.9</td>
<td>32.2</td>
<td>37.6</td>
<td>28.7</td>
</tr>
<tr>
<td>Ensemble RNN-MMT (Delbrouck and Dupont, 2018)</td>
<td>59.6</td>
<td>40.3</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>Unconstrained Transformers MMT (Helcl et al., 2018)</td>
<td>59.1</td>
<td>42.7</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>

**Our Baseline Transformers (from scratch)**

<table>
<thead>
<tr>
<th></th>
<th>2016 METEOR</th>
<th>2016 BLEU</th>
<th>2017 METEOR</th>
<th>2017 BLEU</th>
<th>COCO METEOR</th>
<th>COCO BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>NMT</td>
<td>56.4</td>
<td>37.6</td>
<td>51.3</td>
<td>30.9</td>
<td>47.2</td>
<td>27.5</td>
</tr>
<tr>
<td>+CC</td>
<td>58.8</td>
<td>39.5</td>
<td>55.6</td>
<td>36.2</td>
<td>51.5</td>
<td>33.0</td>
</tr>
<tr>
<td>MMT</td>
<td>55.4</td>
<td>35.2</td>
<td>49.5</td>
<td>27.7</td>
<td>46.2</td>
<td>25.4</td>
</tr>
</tbody>
</table>

**VTLM: Pre-train and fine-tune on Multi30k**

<table>
<thead>
<tr>
<th></th>
<th>2016 METEOR</th>
<th>2016 BLEU</th>
<th>2017 METEOR</th>
<th>2017 BLEU</th>
<th>COCO METEOR</th>
<th>COCO BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>MMT</td>
<td>59.0</td>
<td>40.2</td>
<td>53.5</td>
<td>32.7</td>
<td>49.3</td>
<td>28.9</td>
</tr>
</tbody>
</table>

**TLM: Pre-train on CC – fine-tune on Multi30k**

<table>
<thead>
<tr>
<th></th>
<th>2016 METEOR</th>
<th>2016 BLEU</th>
<th>2017 METEOR</th>
<th>2017 BLEU</th>
<th>COCO METEOR</th>
<th>COCO BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>NMT</td>
<td>60.7</td>
<td>43.1</td>
<td>56.5</td>
<td>37.6</td>
<td>53.3</td>
<td>34.8</td>
</tr>
<tr>
<td></td>
<td>60.5 ± 0.21</td>
<td>42.5 ± 0.46</td>
<td>56.4 ± 0.10</td>
<td>37.3 ± 0.38</td>
<td>53.1 ± 0.13</td>
<td>34.6 ± 0.17</td>
</tr>
<tr>
<td>MMT</td>
<td>60.3</td>
<td>41.9</td>
<td>56.7</td>
<td>37.6</td>
<td>53.3</td>
<td>34.3</td>
</tr>
<tr>
<td></td>
<td>60.2 ± 0.08</td>
<td>41.7 ± 0.18</td>
<td>56.5 ± 0.16</td>
<td>37.5 ± 0.10</td>
<td>53.0 ± 0.20</td>
<td>34.1 ± 0.14</td>
</tr>
</tbody>
</table>

**VTLM: Pre-train on CC – fine-tune on Multi30k**

<table>
<thead>
<tr>
<th></th>
<th>2016 METEOR</th>
<th>2016 BLEU</th>
<th>2017 METEOR</th>
<th>2017 BLEU</th>
<th>COCO METEOR</th>
<th>COCO BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>NMT</td>
<td>61.2</td>
<td>43.3</td>
<td>56.9</td>
<td>37.2</td>
<td>53.7</td>
<td>35.1</td>
</tr>
<tr>
<td></td>
<td>60.5 ± 0.46</td>
<td>42.5 ± 0.53</td>
<td>56.4 ± 0.34</td>
<td>37.0 ± 0.16</td>
<td>53.1 ± 0.42</td>
<td>34.6 ± 0.40</td>
</tr>
<tr>
<td>MMT</td>
<td>60.8</td>
<td>42.7</td>
<td>57.1</td>
<td>38.1</td>
<td>53.1</td>
<td>34.2</td>
</tr>
<tr>
<td></td>
<td>60.6 ± 0.15</td>
<td>42.6 ± 0.14</td>
<td>56.9 ± 0.20</td>
<td>37.7 ± 0.43</td>
<td>53.0 ± 0.05</td>
<td>33.9 ± 0.19</td>
</tr>
</tbody>
</table>

**VTLM: Alternative (0% visual masking during pre-training)**

<table>
<thead>
<tr>
<th></th>
<th>2016 METEOR</th>
<th>2016 BLEU</th>
<th>2017 METEOR</th>
<th>2017 BLEU</th>
<th>COCO METEOR</th>
<th>COCO BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>MMT</td>
<td>61.3</td>
<td>44.0</td>
<td>57.2</td>
<td>38.0</td>
<td>53.8</td>
<td>35.2</td>
</tr>
<tr>
<td></td>
<td>60.9 ± 0.30</td>
<td>43.3 ± 0.59</td>
<td>57.1 ± 0.07</td>
<td>37.6 ± 0.31</td>
<td>53.6 ± 0.17</td>
<td>35.1 ± 0.09</td>
</tr>
</tbody>
</table>

Table 1: Quantitative comparison of experiments: when the mean and the standard deviation is reported, the single numbers appearing above, denote the maximum across three different runs.

yields slightly better results overall. Overall, MMT fine-tuning on VTLM sets a new state of the art across all Multi30k test sets. We leave the exploration of visual region masking for the MRC task as future work and proceed with the alternative variant in the following experiments.

**Encoder attention parameters.** When fine-tuning the TLM for MT, the default XLM implementation randomly initialises the decoder’s missing encoder attention parameters. In our experiments, we noticed that copying those parameters from the TLM self-attention layers substantially improves the results up to 2.2 BLEU.

6We exclude Grönroos et al. (2018) as their improvements (45.5 BLEU) were not due to multi-modality but rather to other modifications such as heavy parallel data augmentation, domain fine-tuning, and ensembling.

3.2 Explicit masking

Here, we will evaluate the extent to which the visual information is taken into account (i) when TLM/VTLM predicts masked tokens, and (ii) when the fine-tuned NMT and MMT models are forced to translate source sentences with missing visual entities. For the latter, we use Flickr30k entities (Plummer et al., 2015) to mask head nouns in 2016 test set sentences, similar to Caglayan et al. (2019).

**Last-word masking.** In this experiment, we measure the target word prediction accuracy, when last tokens of input caption pairs are systematically masked during evaluation. Table 2 suggests

7We pre-process the sentences to ensure that they do not end with punctuation marks, which would make the task easier for masked punctuation.
Table 2: Masked last-word prediction accuracies: VTLM gains are with respect to TLM, whereas the incongruent (+shuf) drops are relative to VTLM.

<table>
<thead>
<tr>
<th></th>
<th>Valid</th>
<th>Test</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>EN</td>
<td>DE</td>
<td>BOTH</td>
<td>EN</td>
</tr>
<tr>
<td>TLM</td>
<td>89.0</td>
<td>87.3</td>
<td>55.2</td>
<td>88.5</td>
</tr>
<tr>
<td>VTLM</td>
<td>↑ 0.9</td>
<td>↑ 1.4</td>
<td>↑ 5.0</td>
<td>↑ 1.1</td>
</tr>
<tr>
<td>+shuf</td>
<td>⇓ 1.0</td>
<td>⇓ 0.2</td>
<td>⇓ 7.7</td>
<td>⇓ 1.3</td>
</tr>
</tbody>
</table>

Table 3: Entity masking on 2016 test set: results are BLEU averages of three fine-tuned MT systems.

<table>
<thead>
<tr>
<th></th>
<th>Mask</th>
<th>Remove</th>
</tr>
</thead>
<tbody>
<tr>
<td>TLM→NMT</td>
<td>31.44</td>
<td>27.38</td>
</tr>
<tr>
<td>TLM→MMT</td>
<td>↓ 0.43</td>
<td>↓ 0.26</td>
</tr>
<tr>
<td>VTLM→NMT</td>
<td>31.27</td>
<td>27.63</td>
</tr>
<tr>
<td>VTLM→MMT</td>
<td>↑ 1.65</td>
<td>↑ 0.65</td>
</tr>
</tbody>
</table>

that the visual information is much more helpful (i.e. up to 6% accuracy improvement) when last tokens are masked in both English and German captions. However, if one caption is available, it provides enough context for cross-lingual prediction. Finally, when we shuffle (+shuf) the test set features to introduce incongruence (Elliott, 2018), we see that the VTLM model deteriorates substantially. This confirms that the accuracy improvements are not due to side-effects of experimentation noise, such as regularisation or random seed related effects.

Entity masking in MT. We devise two ways of masking entities i.e. we either replace them with the [MASK] token or remove them entirely so that the masking phenomena is not known to the model. The results in Table 3 show that MMT models can recover the missing source context to some extent, only when they are pre-trained using the proposed VTLM objective. In other words, the grounding ability can only be acquired when visual modality is present for both pre-training and fine-tuning. The gap between MASK and REMOVE also seems to highlight the importance of reserving a source position even it is corrupted/masked.

3.3 Visual attention in MMT

Here we take the MMT decoder’s cross-attention layers and measure the attention mass they attribute to regional features in the input embeddings. Although the encoder’s self-attention layers produce increasingly mixed contextual embeddings as we move towards the top layers, Brunner et al. (2020) show that the final layer states still encode corresponding input embeddings to some extent. With this assumption at hand, Figure 2 shows the average attention mass attributed to the first 36 (visual) top-layer encoding states, by each cross-attention layer in the decoder. We find these results to be in agreement with the quantitative metrics (Table 1), with VTLM-MMT assigning substantially more attention to these positions, compared to TLM-MMT and MMT from scratch.

4 Conclusions

We proposed a novel cross-lingual visual pre-training approach and tested its efficacy for multimodal machine translation. Our pre-training approach extends the TLM framework (Lample and Conneau, 2019) with regional features and performs masked language modelling and masked region classification on a three-way parallel corpus. We show that this leads to substantial improvements compared to multimodal machine translation with cross-lingual pre-training only or without pre-training at all. As future work, we consider exploring more informed masking strategies for visual regions and investigating the impact of visual masking probability for the MRC pre-training task for downstream MMT performance.

Acknowledgments

This work was supported in part by TUBA GEBIP fellowship awarded to Erkut Erdem, and the MMVC project funded by TUBITAK and the British Council via the Newton Fund Institutional Links grant programme (grant ID 219E054 and 352343575). Lucia Specia, Pranava Madhyastha and Ozan Caglayan also received support from MultiMT project (H2020 ERC Starting Grant No. 678017) and Lucia Specia from the Air Force Office of Scientific Research (under award number FA8655-20-1-7006).
References


Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. **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.