Adaptive Cross-Modal Embeddings for Image-Text Alignment

Jônatas Wehrmann, Camila Kolling, Rodrigo C. Barros
Machine Intelligence and Robotics Research Group
School of Technology, Pontifícia Universidade Católica do Rio Grande do Sul
Av. Ipiranga, 6681, 90619-900, Porto Alegre, RS, Brazil
Email: {jonatas.wehrmann, camila.kolling}@edu.pucrs.br, rodrigo.barros@pucrs.br

Abstract

In this paper, we introduce a novel approach for training image-text alignment models, namely ADAPT. Image-text alignment methods are often used for cross-modal retrieval, i.e., to retrieve an image given a query text, or captions that successfully label an image. ADAPT is designed to adjust an intermediate representation of instances from a modality a using an embedding vector of an instance from modality b. Such an adaptation is designed to filter and enhance important information across internal features, allowing for guided vector representations – which resembles the working of attention modules, though far more computationally efficient. Experimental results on two large-scale Image-Text alignment datasets show that ADAPT-models outperform all the baseline approaches by large margins. Particularly, for Image Retrieval, ADAPT, with a single model, outperforms the state-of-the-art approach by a relative improvement of R@1 $\approx 24\%$ and for Image Annotation, R@1 $\approx 8\%$ on Flickr30k dataset. On MS COCO it provides an improvement of R@1 $\approx 12\%$ for Image Retrieval, and $\approx 7\%$ R@1 for Image Annotation. Code is available at https://github.com/jwehrmann/retrieval.pytorch.

Introduction

Neural networks can been seen as fully-differentiable end-to-end computational graphs, allowing for data-driven training of complete models. This particular feature enabled the possibility of learning multimodal representations and models, that have been used in many tasks, such as Image Captioning (Vinyals et al. 2015), VQA (Anderson et al. 2017), Text-to-Image Generation (Goodfellow et al. 2014; Zhang et al. 2017), Visually-grounded Translation (Elliott et al. 2016), and Image Search via textual queries (Kiros, Salakhutdinov, and Zemel 2014; Faghri et al. 2017; Wehrmann and Barros 2018; Lee et al. 2018).

Even though *multimodal model* is a broad term that comprises roughly any model trained over more than a single modality (e.g., images, videos, text, audio), in this work we focus only on the Images-Text Alignment problem – which is also referred in the literature as Multimodal Retrieval, Cross-modal Retrieval and Bidirectional Alignment. Image-Text alignment models are mainly used for two main tasks: (i) Image Retrieval by using a textual query; and (ii) Image Annotation, which consists in finding proper textual descriptions for a given image. A typical framework for training Image-Text Alignment models is the use of neural networks to extract high-level features of both images and captions. Those features are then projected onto the same shared space, the so-called multimodal embedding space – or visual-semantic embedding space (Kiros, Salakhutdinov, and Zemel 2014; Faghri et al. 2017; Vendrov et al. 2016). A pairwise loss function is used to approximate similar pairs, while making uncorrelated ones to be far from each other in that space. Recent work have demonstrated that the use of more integrated representations can be helpful as the model would be able to learn fine-grained correlations. For instance, work in (Lee et al. 2018) used visual features to compute textual ones and *vice-versa* with the aid of stacked attention layers, which helped to achieve better predictive results. Nevertheless, such an upgrade comes with a cost: it is much slower during both training and test times due to the need of computing distinct features for each image-text pair.

This work introduces ADAPT, a method to improve the embedded representation of instances from modality a based on the global information of the modality b. ADAPT is designed to modify intermediate features (word-level or region-wise projections) by using parameters predicted by the vector representation of the other modality. Such a feature adaptation procedure works as a filtering strategy. For instance, we can use visual-based features in order to filter the most important hidden-state dimensions of captions to build a better textual embedding. We show that such an approach, despite being quite faster during both training and test times, is able to outperform attention-based ones, specially in the Image Retrieval task.

We also provide a comprehensive study on the impact of each component within ADAPT. This allows us to present the importance of each architectural decision, as well as better understand the achieved results. Such a study have shown that it is possible to make our models even lighter and faster while sill outperforming SCAN (same R@1 but 25× faster). Finally, we introduce a strategy to visualize model predictions, and better understand its behavior.

Adaptive Cross-modal Embeddings

In this work we propose ADAPT: *Adapt*ive Cross-Modal Embeddings to improve image-text alignment. There are two main tasks that involves image-text alignment: (i) Image Retrieval by using a caption as a textual query (text-to-image retrieval); and (ii) Image Annotation (image-to-text retrieval), that consists in retrieving the most correlated captions for a given image. Those models also can be used to estimate a correlation between an image and a textual description.

There are three main approaches for learning image-text alignment models in regards to the modality entanglement level: (i) training a so-called visual-semantic embedding space (Kiros, Salakhutdinov, and Zemel 2014; Faghri et al. 2017; Wehrmann and Barros 2018; Gu et al. 2017), in which both images and texts are represented by vectors that are learned independently from each other; (ii) attention-based strategies (Nam, Ha, and Kim 2016; Lee et al. 2018) in which high-level features from a given modality affect the encoding of the other modality, e.g., image features are used to compute text representations; and (iii) approaches that handle fully-entangled representations (De Vries et al. 2017), as they combine both modalities in a neural network, so the network itself learns the similarity function (Ma et al. 2015).

One can note that as the feature entanglement level increases, the computation required for computing similarities between all the image-text pairs increases accordingly. Hence, methods purely based on independently-computed vectors are often much faster than the other ones. Such efficiency often comes with a cost of predictive performance, given that it becomes harder for the model to learn similarities and differences across images and texts using only global generic information.

ADAPT is somewhat similar to the attention-based approaches, though presenting itself as a much more efficient choice. In summary, ADAPT is a lighter strategy for using high-level information from a base modality *a* instance to generate a filtered version of the intermediate features (i.e., time-steps or spatial embeddings) of a target modality *b*. For instance, ADAPT can use textual information to filter and approximate spatial-level features (e.g., image regions found by an object detector), generating a guided image embedding vector. Therefore, the final image representation vector would be generated based on the textual query at hand. Intuitively, a single image could be represented by many distinct vectors, all generated using the different query instances as guide.

The generic visual-semantic framework to train image-text alignment models consists in approximating correlated vectors in a shared multimodal space, while making uncorrelated ones far from each other. Our approach uses this concept, though ultimately the final vectors that represent image and text instances are embeddings generated by ADAPT.

ADAPT comprises three three main steps: (i) projection of scale (γ) and shift (β) vectors; (ii) filtering and adaptation of the other modality inner representations using γ and β; and (iii) a *fovea* module on the filtered feature map, that allows for the model to focus on important image/text details.

A generic formulation of ADAPT is given as follows. Assume, feature matrices $\mathcal{A}$ and $\mathcal{B}$, from modalities a and b, respectively. Those matrices are either spatial image regions or textual representations. In addition, consider that we will use a vector $\mathbf{a}$ generated from A, to adapt the representation of B, and get the adapted vector $\overline{\mathbf{b}}$. Therefore, $\mathcal{A}^{n_a \times f_a}$ is a feature matrix that contains n_a (time-steps or regions) and f_a-dimensional feature vectors, and $\mathcal{B}^{n_b \times f_b}$ is a feature matrix from the other modality. We project each vector comprised in $\mathcal{A}$ and $\mathcal{B}$ into the d-dimensional latent space using functions

$$A = \psi(\mathcal{A}), B = \phi(\mathcal{B}) \tag{1}$$

generating a two new matrices $A^{n_a \times d}$ and $B^{n_b \times d}$. Both functions can be linear projections, or multi-layered non-linear neural networks. Vector $\mathbf{a}$ is obtained with a global pooling, $\mathbf{a} = \textsc{pooling}(A)$, which summarizes A into a single vector $\mathbf{a} \in \mathbb{R}^d$. $\mathbf{a}$ is then used to project $g_a(\mathbf{a}) = \gamma_a$ and $b_a(\mathbf{a}) = \beta_a$,

$$\gamma_a = g(\mathbf{a}, \theta_g), \beta_a = b(\mathbf{a}, \theta_b) \tag{2}$$

whose function is to adapt, i.e, filter and shift all the vectors in B (features from the *b* modality), by

$$\overline{B_i} = B_i \odot \gamma_a + \beta_a \tag{3}$$

where $\odot$ depicts a point-wise vector multiplication, which generates the adapted $\overline{B}$ matrix. That matrix is then processed by the *fovea* module, that consists in applying a per-dimension λ-smoothed $\textsc{Softmax}$ across all the n_b features in $\overline{B}$, producing a channel-wise attention-like mask *M*,

$$M_{ij} = \left(\frac{e^{\overline{B}_{ij}}}{\sum_{i=1}^{n_b} e^{\overline{B}_{ij}}} \lambda \right) \tag{4}$$

Finally, we obtain the filtered vector representation of B, namely $\overline{\mathbf{b}}$, by applying the *fovea* mask *M* over $\overline{B}$, followed by a global average pooling,

$$\overline{\mathbf{b}} = \frac{1}{n_b} \sum_{i=1}^{n_b} \left(\overline{B} \odot M \right)_i \tag{5}$$

where $\overline{\mathbf{b}}$ is the adapted version of the B features, using the projections of γ_a and β_a from the mapping over $\mathbf{a}$. $\textsc{pooling}$ is an average-pooling layer unless stated otherwise. It acts over the variable-sized (temporal or spatial) dimension, i.e., on the first matrix dimension. In this work, we mainly use $\gamma_a = g(\mathbf{a}, \theta_g)$ and $\beta_a = b(\mathbf{a}, \theta_b)$ as linear projections, i.e. $\mathbf{a}\theta^T$, since it performed best on validation data. Nevertheless, one could explore them as non-linear projections and try to reduce the computation required in this step. Moreover, functions ψ and ϕ are typically linear projections, followed by a normalization layer $\textsc{norm}$, that can be Batch Normalization (Ioffe and Szegedy 2015), L2 Normalization, Instance Normalization (Ulyanov, Vedaldi, and Lempitsky 2016) and the like. We observed that Batch Norm performed best, and therefore, we use it as default strategy.

Relation to Conditional BatchNorm (De Vries et al. 2017). The first part of ADAPT resembles Conditional

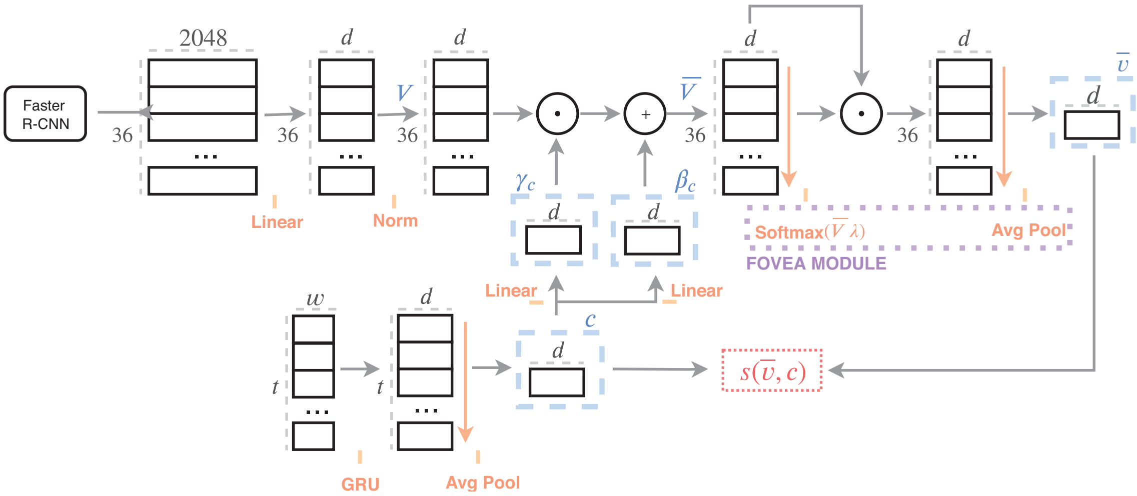

Figure 1: Overall architecture of ADAPT-T2I.

BatchNorm (De Vries et al. 2017) when using BatchNorm as NORM($\cdot$) function within ϕ and ψ. One of the key differences is that we apply the normalization directly on high-level features of the extracted image regions, rather than in the early stages of the spatial visual processing. It would be computationally prohibitive to employ that approach for the entire network given that we need to compute image embedding vectors for each image-text pair. For instance, it would require 5×10^6 forward passes in a Deep ConvNet for evaluating the 1K MS COCO Validation Set. On the other hand, in ADAPT we can pre-compute all the high-level features, and only compute the forward pass of the adaptation procedure, which roughly comprehends the asymptotic cost of a convolutional layer – being actually much lighter in practical terms. Apart from that, consider that the complete incarnation of ADAPT also employs the *fovea* module. That module proved to be one of the key components behind the performance of ADAPT, being even more important than the normalization strategy.

Relation to Dynamic Convolution (Wu et al. 2019). The adaptation procedure is also somehow related to Dynamic Convolutions, though in our case, the operation we perform is not a complete convolution, given that we process the input only with d weights and given by values of $\gamma_a \in \mathbb{R}^d$ (kernel weights) and $\beta_a \in \mathbb{R}^d$ (kernel biases) – which would be equivalent to a single convolutional filter of size 1, without summing the results across the channel dimension. We could have projected γ and β to be matrices in order to parameterize a complete convolutional layer, though, in that case, the computational cost would be prohibitive.

Text-to-Image Adaptation

For Text-to-Image embedding adaptation, namely ADAPT-T2I (Figure 1), we use the averaged d-sized representation extracted from the text encoder $\mathbf{c}$ as a base vector to project $\gamma_c = g(\mathbf{c})$ and $\beta_c = b(\mathbf{c})$ to process the intermediate visual features V. Recall that $\mathbf{c} \in \mathbb{R}^d$ is the global embedding vector for the caption features, which is projected onto two separate spaces in form of vectors $\gamma_\mathbf{c}$ and $\beta_\mathbf{c}$. The formulation of the visual features adaptation $\mathcal{V}$ based on a caption is as follows.

$$\overline{V_i} = \left(\psi(\mathcal{V}_i) \odot \gamma_c + \beta_c\right) \tag{6}$$

$$\overline{\mathbf{v}} = \text{POOLING}\left(\text{SOFTMAX}(\overline{V}\lambda) \odot \overline{V}\right) \tag{7}$$

The resulting vector $\mathbf{v} \in \mathbb{R}^d$ is further normalized to have unit euclidean norm, so the inner product between $\overline{\mathbf{v}}$ and $\mathbf{c}$ results in the cosine similarity.

Image-to-Text Adaptation

Image-to-Text Adaptation models (ADAPT-I2T) are similar to the Text-to-Image ones. For the ADAPT-I2T, we use visual features in order to adapt and filter caption features so as to generate a final textual vector representation $\overline{\mathbf{c}}$. In this case, we apply a global average pooling on the first dimension of the image visual matrix V, and the resulting vector $\mathbf{v} \in \mathbb{R}^d$ is used as input to compute both γ_v and β_v, that are finally applied over each time-step representation, as follows.

$$\overline{C_i} = \left(\phi(\mathcal{C}_i) \odot \gamma_v + \beta_v\right) \tag{8}$$

$$\overline{\mathbf{c}} = \text{POOLING}\left(\text{SOFTMAX}(\overline{C}\lambda) \odot \overline{C}\right) \tag{9}$$

Recall that the SOFTMAX operator generates per-dimension masks that weights each visual region in ADAPT-T2I, and each time step in ADAPT-I2T. In addition, POOLING operates only to reduce temporal and spatial dimensions to a fixed-sized vector.

Text Encoder

For encoding image captions we make use of the widely adopted GRU networks (Cho et al. 2014), which are naturally suited to process temporal data. We encode the temporal data in a bidirectional manner, i.e., text is processed in

Table 1: Cross-modal results on Flickr30k test set. Underlined values outperform best published results. Bold values highlight current state-of-the-art results.

	Image Annotation			Image Retrieval			Total
Method	R@1	R@5	R@10	R@1	R@5	R@10	$\sum$
DVSA (Karpathy and Fei-Fei 2015)	15.2	37.7	50.5	22.2	48.2	61.4	235.2
SM-LSTM (Huang, Wang, and Wang 2017)	30.2	60.4	72.3	42.5	71.9	81.5	358.8
VSE++ (Faghri et al. 2017)	52.9	-	87.2	39.6	-	79.5	-
DAN (Nam, Ha, and Kim 2016)	55.0	81.8	89.0	39.4	69.2	79.1	413.5
DPC (Zheng et al. 2017)	55.6	81.9	89.5	39.1	69.2	80.9	416.2
SCO (Huang, Wu, and Wang 2017)	55.5	82.0	89.3	41.1	70.5	80.1	418.5
SCAN-t2i-avg (Lee et al. 2018)	61.8	87.5	93.7	45.8	74.4	83.0	446.2
SCAN-i2t-avg (Lee et al. 2018)	67.9	89.0	94.4	43.9	74.2	82.8	452.2
SCAN-ens	67.4	90.3	95.8	48.6	77.7	85.2	465.0
ADAPT-I2T	70.2	90.8	95.8	55.5	82.7	89.8	484.8
ADAPT-T2I	73.6	93.7	96.7	57.0	83.6	90.3	494.8
ADAPT-ENS(T2I+I2T)	**76.6**	**95.4**	**97.6**	**60.7**	**86.6**	**92.0**	**508.9**

both forward ($\overrightarrow{GRU}(\cdot)$) and backward ($\overleftarrow{GRU}(\cdot)$) ways, that produce two $t \times d$ vectors, where d is the number of hidden units, so we can use them as a direct projection for the shared cross-modal embedding space. Backward and forward representations are element-wise averaged, so each time-step d-dimensional embedding contains context information from both the beginning and the ending of the sentences.

Image Encoder

The image encoder, inspired by (Lee et al. 2018), encapsulates three main steps: (i) a forward pass of an object detector network (Faster R-CNN (Ren et al. 2015)) trained on the Visual Genome dataset (Krishna et al. 2017) for extracting the k most important regions within the image; (ii) reduction of the negative values through a Leaky ReLU activation function; and a (iii) global average pooling for generating a global representation of the original image.

Note that the used features were extracted from variable-sized images, where the smallest dimension is limited to a maximum of 500 pixels, and the largest dimension is limited to 800 pixels. This is considered quite a high resolution, and makes our *fovea* module even more relevant, once it can leverage features from details of the original image. Namely, images are processed by a ResNet152 (He et al. 2016) + Faster R-CNN (Ren et al. 2015) network fine-tuned on the Visual Genome dataset (Krishna et al. 2017), which outputs k regions found by the Region Proposal Network, generating a $k \times 2048$ feature matrix. Region features are then projected onto the semantic space by applying an one-dimensional convolutional layer, with a filter-size of $f = 1$, over the regions with d filters (parameters shared across regions), which outputs the $V \in \mathbb{R}^{k \times d}$ matrix. Following related work, we use a fixed value of $k = 36$.

Loss Function

Recent work (Faghri et al. 2017) have shown that hard-contrastive mining plays an important role for training image-text models. However, we have observed that using only hard-contrastive pairs for the entire training often causes unstable gradients during the early stages of the process. To counteract this issue, we make use of the loss introduced in (Wehrmann et al. 2019), which gives more importance for the hard-contrastive instances accordingly to the number of gradient descent steps performed. Such a loss function is as follows,

$$\mathcal{J} = \tau(\epsilon) \cdot \mathcal{J}_m + (1 - \tau(\epsilon)) \cdot \mathcal{J}_s \tag{10}$$

$$\tau = (1 - \eta^{\epsilon}) \tag{11}$$

where ϵ is the number of the current gradient descent step being performed, τ is the trade-off weight computed based on ϵ, and η defines the exponential growth rate of τ, regulating the importance of $\mathcal{J}_s$ and $\mathcal{J}_m$. Those functions are defined by:

$$\mathcal{J}_s(\mathbf{a}, \mathbf{b}) = \sum_{b'} [\alpha - s(\mathbf{a}, \mathbf{b}) + s(\mathbf{a}, \mathbf{b}')] \tag{12}$$

$$+ \sum_{\mathbf{a}'} [\alpha - s(\mathbf{b}, \mathbf{a}) + s(\mathbf{b}, \mathbf{a}'))] \tag{13}$$

$$\mathcal{J}_m(\mathbf{a}, \mathbf{b}) = \max_{\mathbf{b}'} [\alpha - s(\mathbf{a}, \mathbf{b}) + s(\mathbf{a}, \mathbf{b}')]_+ \tag{14}$$

$$+ \max_{\mathbf{a}'} [\alpha - s(\mathbf{b}, \mathbf{a}) + s(\mathbf{b}, \mathbf{a}')]_+ \tag{15}$$

where $\mathbf{b}$ is image $\mathbf{a}$'s description vector representation. $\mathbf{b}'$ and $\mathbf{a}'$ denote the contrastive examples for the image and description queries, respectively. $s(\mathbf{a}, \mathbf{b})$ is the computed similarity between $\mathbf{a}$ and $\mathbf{b}$. To compute $s(\mathbf{a}, \mathbf{b})$ we first scale $\mathbf{a}$ and $\mathbf{b}$ to have unit norm, so the inner product of both results become the cosine similarity. Note that we use $\mathbf{a}$ and $\mathbf{b}$ given that when training ADAPT-I2T $\mathbf{a} = \mathbf{v}$ and $\mathbf{b} = \overline{\mathbf{c}}$, while for ADAPT-T2I $\mathbf{a} = \overline{\mathbf{v}}$ and $\mathbf{b} = \mathbf{c}$.

Experimental Setup

Datasets

We train and evaluate our models in two large-scale multimodal datasets, namely MS COCO (Lin et al. 2014) and

Table 2: Cross-modal results on MS COCO 1k test set. Underlined values outperform best published results. Bold values highlight current state-of-the-art results.

	Image Annotation			Image Retrieval			Total
Method	R@1	R@5	R@10	R@1	R@5	R@10	$\sum$
CHAIN (Wehrmann and Barros 2018)	61.2	89.3	95.8	46.6	81.9	90.9	465.7
VSE++ (Faghri et al. 2017)	64.6	-	95.7	52.0	-	92.0	-
DPC (Zheng et al. 2017)	65.6	89.8	95.5	47.1	79.9	90.0	467.9
GXN (Gu et al. 2017)	68.5	-	97.9	56.6	-	94.5	-
SCO (Huang, Wu, and Wang 2017)	69.9	92.9	97.5	56.7	87.5	94.8	499.3
SCAN-t2i-avg (Lee et al. 2018)	70.9	94.5	97.8	56.4	87.0	93.9	500.5
SCAN-i2t-lse (Lee et al. 2018)	69.2	93.2	97.5	54.4	86.0	93.6	493.9
SCAN-ens	72.7	94.8	98.4	58.8	88.4	94.8	507.9
ADAPT-I2T	74.5	94.2	97.9	62.0	90.4	95.5	514.5
ADAPT-T2I	75.3	95.1	98.4	**63.3**	90.0	95.5	517.6
ADAPT-ENS(T2I+I2T)	**76.5**	**95.6**	**98.9**	62.2	**90.5**	**96.0**	**519.8**

Flickr30k (Plummer et al. 2015). COCO is widely used for training and evaluating systems for image-caption alignment, and it has become the standard benchmark to evaluate the predictive performance of state-of-the-art methods. It comprises 113,287 images for training, 5,000 images for validation, and 5,000 images for testing. Flickr30k comprehends roughly 28,000 images for training and 1,000 for both validation and testing. We used the same splits as those used by state-of-the-art approaches.

For evaluating the results, we use the same measures as those in (Kiros, Salakhutdinov, and Zemel 2014; Vendrov et al. 2016; Faghri et al. 2017): $R@K$ (reads "Recall at K"), which is the percentage of queries in which the ground-truth is one of the first K retrieved results. The higher its value, the better.

Experimental Analysis

In this section we evaluate the performance of our models by comparing them to published state-of-the-art approaches. We first report results on the Flickr30k Test Set. Table 1 shows that our methods, namely ADAPT-T2Iand ADAPT-I2T, outperform all the baseline (including ensemble models) approaches by large margins in all metrics. Most notably, a single ADAPT-T2I model is able to outperform the current state-of-the-art approach, namely SCAN-i2t-avg, in absolute R@1 5.7% in image annotation, and R@1 11.2% image retrieval. Comparing the ensemble methods, we achieve top performance with R@1 76.6%, surpassing the strongest image annotation method by absolute 9.2%, and by 12.1% R@1 image retrieval, which represents a relative improvement of $\approx$ 25%. It is clear that ADAPT helps in both tasks, though benefits more the image retrieval task. Results also highlight that using text information to improve visual information seems to be more effective than the use of visual information to adapt text embeddings. Moreover, ADAPT is also faster than its strongest competitor, as detailed in the ablation study.

Table 2 shows results for the MS COCO Test set. ADAPT-based models present once again the best overall results. As before, we observe a major increase in Image Retrieval performance ($\approx 8\%$ in R@1 terms). Finally, note that our single models are even capable of surpassing model ensembling. Current experiment provides additional evidence that ADAPT seems to provide more improvement to the Image Retrieval Task, in which our approach leads to $\approx 12\%$ relative improvement when using a single model (ADAPT-T2I).

Effects of the *fovea* module. The first component we analyze is the *fovea* module applied after the multimodal interaction. As shown in Table 3 and in Figure 2, the use of Softmax normalization grants 4% to 5% R@1 in both tasks for the ADAPT-T2I approach. Those results provide reassurance behind the importance of the *fovea* module, which in theory allows our models to focus on relevant high-resolution image regions given a caption, and use them for building better global embedding vectors. Another component within the *fovea* module is the Softmax smoothing setting λ. We have found that λ is quite important for ADAPT-T2I, while ADAPT-I2T seems to benefit less from it. It is clear that ADAPT-T2I benefits from larger values, i.e., $\gamma = \{\approx 6, 10, 15\}$, different from ADAPT-I2T that performs best with $\gamma = 1$. We also tried to optimize λ via backpropagation in an end-to-end fashion (depicted by $\Delta\gamma$). It was initialized at 10, and at the end of the procedure it converged to 6.27, achieving the third best result as measured by the recall sum $\sum$.

Word-embedding impact. Our hypothesis is that ADAPT models would benefit from pre-trained word-embeddings, given that such vectors already carry rich semantic representations regarding concepts. This could ease the process of adaptation of the intermediate image representations to the global text vector, and *vice-versa*. Table 4 shows experiments that compare results from: (i) $\Delta Rand$, random word embeddings initialized from an uniform distribution $\sim \mathcal{U}(-0.1, 0.1)$ and trained via backpropagation; (ii) using only pre-trained fixed Glove embeddings; (iii) concatenating randomly initialized word-embeddings and Glove vectors, though updating the entire

Table 3: Ablation study on the focal module: cross-modal results on Flickr30k validation set.

	Image Annotation		Image Retrieval		Total
Method	R@1	R@10	R@1	R@10	$\sum$
ADAPT-I2T(No Fovea)	52.8	85.7	49.2	85.1	427.4
ADAPT-I2T	70.7	95.8	54.2	88.42	482.0
ADAPT-T2I(No Fovea)	72.4	95.2	52.9	86.7	478.2
ADAPT-T2I	**76.2**	**96.1**	**57.4**	**88.9**	**494.2**
ADAPT-T2I($\lambda = 1$)	72.4	**97.2**	56.8	88.7	489.8
ADAPT-T2I($\lambda = 5$)	75.1	96.4	57.5	88.3	492.8
ADAPT-T2I($\lambda = 10$)	**76.2**	96.1	57.4	**88.9**	**494.2**
ADAPT-T2I($\lambda = 15$)	76.1	96.5	58.0	88.7	494.1
ADAPT-T2I($\Delta\lambda$)	75.7	95.9	**58.1**	88.4	493.1

concatenated vector during training (ΔRand+ΔGlove); and (iv) using a concatenation of fixed pre-trained Glove vectors and randomly initialized word embeddings updated during training – which is our default option within ADAPT. We expected larger impact on the adoption of pre-trained word vectors, though results show that using Glove *versus* randomly trained word-embeddings leads to 1.6% and 1.9% of absolute R@1 improvement for image retrieval and image annotation, respectively. The best performing approach is ADAPT-T2I(ΔRand+Glove), in which Glove vectors are not updated. The clear drawback of that approach is a larger amount of required memory to store the model, once it doubles the number of word-embedding parameters, which may be problematic in some cases, as discussed in (Wehrmann et al. 2019; Wehrmann, Mattjie, and Barros 2018). Nevertheless, it is a good option when memory is not a constraint.

Table 4: Cross-modal results on Flickr30k validation set.

	Image Annotation		Image Retrieval		Total
Method	R@1	R@10	R@1	R@10	$\sum$
ADAPT-T2I(ΔRand)	74.1	95.4	54.7	87.0	484.1
ADAPT-T2I(ΔRand+ΔGlove)	75.4	96.0	**57.5**	88.4	492.3
ADAPT-T2I(Glove)	75.7	**97.0**	56.6	88.4	493.5
ADAPT-T2I(ΔRand+Glove)	**76.2**	96.1	57.4	**88.9**	**494.2**

Effects of the latent size and time complexity. The latent size is such an important hyper-parameter once it affects directly many aspects of the method: (i) number of parameters; (ii) predictive performance; and (iii) time for training and retrieval. Figure 3 depicts the impact of the latent size in each of those aspects. The y-axis depicts R@1 performance in the Flickr30k test set, x-axis shows time (in seconds) to build the similarity matrix ($N \times N$ matrix), while the marker size represents the number of parameters (both trainable and not trainable) as measured in millions (M). The dashed horizontal line depicts baseline state-of-the-art results (from SCAN), whose model comprises about 15 million parameters, and takes $\approx$ 247 seconds (using their own source code) to build the similarity matrix of $1,000$ images and $5,000$ captions. We ran all the time experiments on a server equipped with GTX 1080Ti GPU, 128GB RAM and Intel Core i9. It becomes quite clear that even using much more compact representation vectors ($d \in \{128, 256, 512\}$) ADAPT-T2I is able to surpass state-of-the-art results by a margin, while running up to one order of magnitude faster. When we compare models with similar complexity in terms of parameters ($d = 1024$), one can observe *absolute* improvements of $\approx$ 10% for both image retrieval and annotation. Note also that one could adopt strategies from (Burns et al. 2019; Wehrmann et al. 2018) to further improve model efficiency.

Qualitative Analysis

We also depict qualitative examples to provide some intuition behind the behavior of the *fovea* module within ADAPT. We aim to provide a visualization that shows the most relevant regions of the input image after performing the feature vector adaptation based on a given caption. To generate such a visualization we extract the feature map M after applying the softmax normalization within the *fovea* module. Different from some attention strategies, our approach generates per-dimension weights across the spatial (of temporal) dimensions. Therefore, we compute the L2 norm of each region vector, i.e., $\forall i \in \{1, 2, ..., k\}||\overline{V}_i||_2$ summarizing the entire channel dimension into a single attention weight. Next, we clip negative values, and normalize feature maps using the largest vector norm. Figure 4 shows results generated the proposed approach. In the leftmost part of the figure we depict the original image, the center part shows all the bounding boxes found by the Faster-RCNN model, and the rightmost part shows the focal points within the image given the caption described in the figure title. We can observe that ADAPT is able to filter irrelevant information, and focus on subjects that are described in the given captions.

Conclusion

In this paper we proposed ADAPT, a method that improves Image-Text Alignment with Cross-modal Embedding Adaptation. It uses a global embedding of the base modality to adapt and filter intermediate features of the target modality, so we can have a guided vector representation. In addition, the proposed *fovea* module introduced within ADAPT have shown to be effective and efficient in replacing stacked attention ones. We have shown that one can use text features to improve the visual representation, allowing for a $\approx 24\%$ relative improvement on Flickr30k in terms of R@1 when compared to the strongest baseline to date – whilst being much faster. Additionally, for Image Annotation, our models also consistently outperformed state-of-the-art ones in most of the metrics. Moreover we perform extensive analysis on the impact of each part of our model. For future work, we intend to further explore ADAPT in another multimodal tasks, such as VQA, Image captioning and Text to image synthesis.

Acknowledgments

We thank Google (LARA), and the Brazilian research CNPq, and FAPERGS for funding this research. This study was financed in part by the *Coordenação de Aperfeiçoamento de Pessoal de Nivel Superior – Brasil (CAPES) – Finance Code 001.*

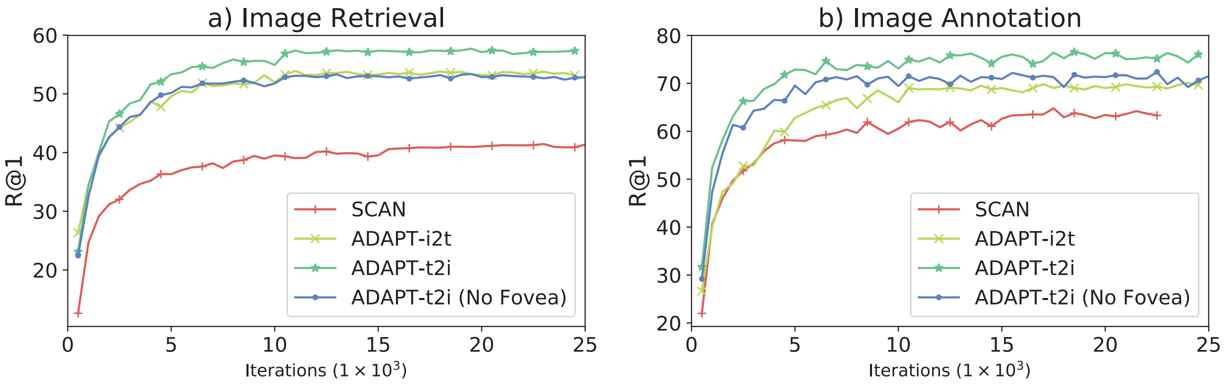

Figure 2: Flickr30k validation data R@1 cross-modal results during training.

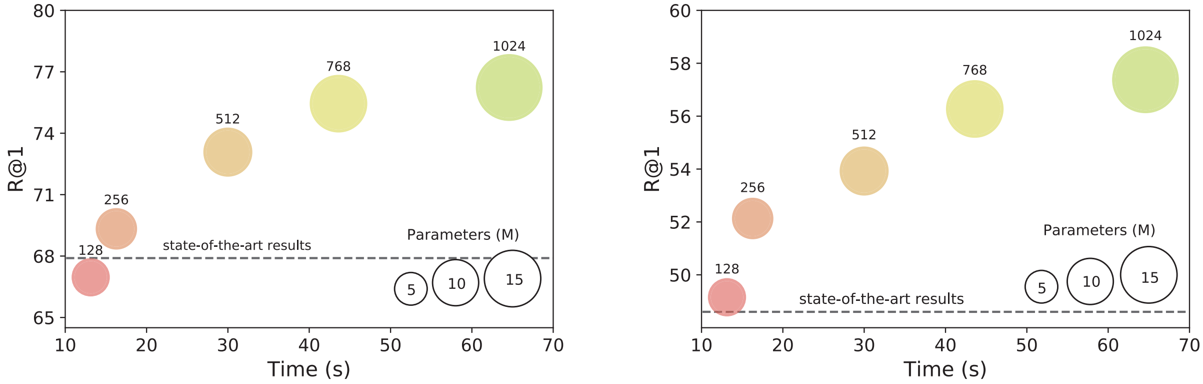

Figure 3: Model performance (R@1) compared to model complexity in terms of number of parameters and retrieval time for (a) Image Annotation and (b) Image Retrieval.

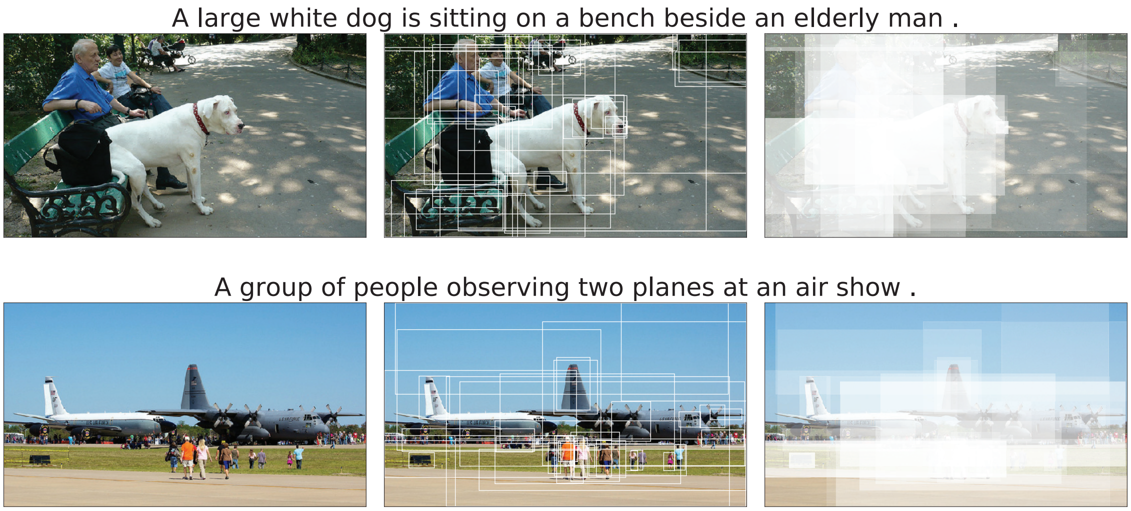
A large white dog is sitting on a bench beside an elderly man .
A group of people observing two planes at an air show .

Figure 4: Visualization of the weighted feature map projected onto the original image.

References

Anderson, P.; He, X.; Buehler, C.; Teney, D.; Johnson, M.; Gould, S.; and Zhang, L. 2017. Bottom-up and top-down attention for image captioning and vqa. *arXiv preprint arXiv:1707.07998*.

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*, 7474–7483.

Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; and Bengio, Y. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

De Vries, H.; Strub, F.; Mary, J.; Larochelle, H.; Pietquin, O.; and Courville, A. C. 2017. Modulating early visual processing by language. In *Advances in Neural Information Processing Systems*, 6594–6604.

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

Faghri, F.; Fleet, D. J.; Kiros, J. R.; and Fidler, S. 2017. Vse++: Improving visual-semantic embeddings with hard negatives.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *Advances in neural information processing systems*, 2672–2680.

Gu, J.; Cai, J.; Joty, S.; Niu, L.; and Wang, G. 2017. Look, imagine and match: Improving textual-visual cross-modal retrieval with generative models. *arXiv preprint arXiv:1711.06420*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*, 770–778.

Huang, Y.; Wang, W.; and Wang, L. 2017. Instance-aware image and sentence matching with selective multimodal lstm. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Huang, Y.; Wu, Q.; and Wang, L. 2017. Learning semantic concepts and order for image and sentence matching. *arXiv preprint arXiv:1712.02036*.

Ioffe, S., and Szegedy, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR* abs/1502.03167.

Karpathy, A., and Fei-Fei, L. 2015. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2015)*, 3128–3137.

Kiros, R.; Salakhutdinov, R.; and Zemel, R. 2014. Multimodal neural language models. In *Proceedings of the 31st International Conference on Machine Learning*, 595–603.

Krishna, R.; Zhu, Y.; Groth, O.; Johnson, J.; Hata, K.; Kravitz, J.; Chen, S.; Kalantidis, Y.; Li, L.-J.; Shamma, D. A.; et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision* 123(1):32–73.

Lee, K.-H.; Chen, X.; Hua, G.; Hu, H.; and He, X. 2018. Stacked cross attention for image-text matching. *arXiv preprint arXiv:1803.08024*.

Lin, T.; Maire, M.; Belongie, S. J.; Bourdev, L. D.; Girshick, R. B.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft COCO: common objects in context. In *European Conference on Computer Vision*.

Ma, L.; Lu, Z.; Shang, L.; and Li, H. 2015. Multimodal convolutional neural networks for matching image and sentence. In *Proceedings of the IEEE international conference on computer vision*, 2623–2631.

Nam, H.; Ha, J.-W.; and Kim, J. 2016. Dual attention networks for multimodal reasoning and matching. *arXiv preprint arXiv:1611.00471*.

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 *Proceedings of the IEEE International Conference on Computer Vision*, 2641–2649.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, 91–99.

Ulyanov, D.; Vedaldi, A.; and Lempitsky, V. 2016. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*.

Vendrov, I.; Kiros, R.; Fidler, S.; and Urtasun, R. 2016. Order-embeddings of images and language. In *International Conference on Learning Representations (ICLR 2016)*.

Vinyals, O.; Toshev, A.; Bengio, S.; and Erhan, D. 2015. Show and tell: A neural image caption generator. In *IEEE Internatinoal Conference on Computer Vision and Pattern Recognition (CVPR 2015)*, 3156–3164.

Wehrmann, J., and Barros, R. C. 2018. Bidirectional retrieval made simple. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7718–7726.

Wehrmann, J.; Armani, M.; More, M. D.; and Barros, R. C. 2018. Fast self-attentive multimodal retrieval. *WACV*.

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

Wehrmann, J.; Mattjie, A.; and Barros, R. C. 2018. Order embeddings and character-level convolutions for multimodal alignment. *Pattern Recognition Letters* 102:15–22.

Wu, F.; Fan, A.; Baevski, A.; Dauphin, Y. N.; and Auli, M. 2019. Pay less attention with lightweight and dynamic convolutions. *arXiv preprint arXiv:1901.10430*.

Zhang, H.; Xu, T.; Li, H.; Zhang, S.; Wang, X.; Huang, X.; and Metaxas, D. N. 2017. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, 5907–5915.

Zheng, Z.; Zheng, L.; Garrett, M.; Yang, Y.; and Shen, Y.-D. 2017. Dual-path convolutional image-text embedding with instance loss. *arXiv preprint arXiv:1711.05535*.