Improving the Performance of Image Captioning Models Trained on Small Datasets

Mikkel du Plessis and Willie Brink

Stellenbosch University, Stellenbosch, South Africa
mikkeldp@gmail.com, wbrink@sun.ac.za

Abstract. Recent work in image captioning seems to be driven by increasingly large amounts of training data, and requires considerable computing power for training. We propose and investigate a number of adjustments to state-of-the-art approaches, with an aim to train a performant image captioning model in under two hours on a single consumer-level GPU using only a few thousand images. Firstly, we address the issue of sparse object and scene representation in a small dataset by combining visual attention regions at various levels of granularity. Secondly, we suppress semantically unlikely caption candidates through the introduction of language model rescoring during inference. Thirdly, in order to increase vocabulary and expressiveness, we propose an augmentation of the set of training captions through the use of a paraphrase generator. State-of-the-art performance on the Flickr8k test set is achieved, across a number of evaluation metrics. The proposed model also attains competitive test scores compared to existing models trained on a much larger dataset. The findings of this paper can inspire solutions to other vision-and-language tasks where labelled data is scarce.

Keywords: Image captioning, Deep learning, Low-data regime

1 Introduction

Image captioning is the task of creating a short natural language expression to describe the visual content of a given image (as illustrated in Fig. 1). An image captioning model should in concept learn to identify salient objects within an image, determine relationships between different objects, form an understanding of the image as a whole, and then generate a sensible and semantically correct phrase. In order to generalise well, current state-of-the-art models require large amounts of diverse training data (samples of image-caption pairs) as well as considerable computing power for training. Oscar [1], for example, was pre-trained on 6 million image-caption pairs, and fine-tuned on the Microsoft COCO dataset [2] which has 328,000 images. Training this model required 22 days on eight Tesla V100 GPUs, with VRAM usage peaking at 168GB. Of course, in some domains access to these amounts of labelled data and computing resources can be challenging. A question arises: can comparable performance be achieved by models trained on much smaller datasets?

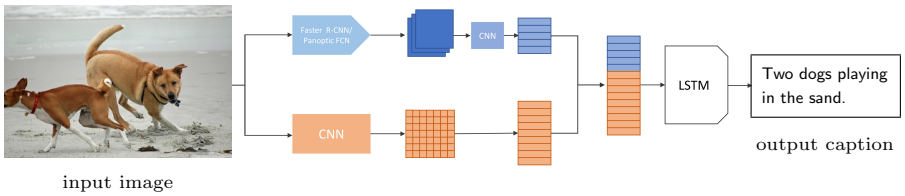

Fig. 1. The task of an image captioning model is to generate a short natural language description of a given input image. Detail of our proposed model is given in Fig. 2.

In this paper we focus specifically on models that can be trained on a single consumer-level GPU in under 2 hours, using only a few thousand images. We make use of the Flickr8k dataset [3], and investigate various strategies that may improve performance compared to existing work.

We implement the seminal work of Xu et al. [4] as a base model. They introduced the idea of an encoder-decoder architecture within the context of image captioning, with a transparent attention mechanism that enables the model to focus on appropriate parts of the image while generating a caption. Attention also provides a level of explainability, and gives the user a means to evaluate a model's understanding of objects, interactions, and the scene as a whole.

We propose the following ideas, and explore their ability to improve performance of image captioning models trained on small datasets:

1. A significant challenge in image captioning is the sparse representation of objects and scenes within a limited set of examples. By extracting attention regions at various levels of granularity, we aim to present the caption generator (the decoder) with encodings of the image that carry richer information and context.
2. The expressiveness of the caption generator is learned from a limited set of human-annotated captions in the training set, and a model trained on a small dataset might struggle to generate semantically correct captions. We therefore introduce a language model during inference, to rescore caption candidates in a beam-search scheme and suppress semantically unlikely instances.
3. As mentioned in the previous point, a small dataset may not encapsulate the diversity of a particular language to a sufficient degree, thus restricting the vocabulary and expressiveness of the model. As a potential remedy we augment the training captions by means of a paraphrase generator.

It is shown experimentally that by implementing these ideas, state-of-the-art performance can be achieved on the Flickr8k test set, across a number of evaluation metrics. Additionally, our final model is able to yield competitive results compared to models from the literature trained on substantially more data.

2 Related Work

Like many problems in modern computer vision, image captioning has been approached predominantly with variants of deep neural networks. One of the first models to utilise a neural network [5] uses a multimodal language network that jointly learns an image-text representation. The model can generate descriptions of images, and retrieve images given a natural language query. This work was followed by the first encoder-decoder architecture [6], where an encoder learns a joint image-text representation and a neural language decoder generates a description. Mao et al. [7] replaced the feed-forward neural language model with a recurrent one. A dynamically sized context vector is used, as opposed to a fixed context window, allowing the decoder access to all previously generated words in the description. Vinyals et al. [8] used an LSTM as decoder, and their model presents the image only at the first step of the decoder instead of every step in the output word sequence.

The above-mentioned works all represent an input image as a single static feature vector from the last layer of a pre-trained convolutional neural network. More recent work attempts a more dynamic multi-vector representation of the image. Xu et al. [4] pioneered this approach, by using the feature maps from early layers in a pre-trained CNN as a set of feature vectors, and feeding those to an attention mechanism for information aggregation. We will adopt this architecture as a base model. The model of Xu et al. [4] considers a uniform grid over the input image, that does not adapt to the content of the image. To address this limitation, object-level attention regions have been proposed [9, 10] to enable the encoding of more fine-grained information.

When training data is limited, the use of object-level attention regions may not be sufficient for a model to adequately learn about the many appearance variations of objects and salient regions. In an effort to remedy this, we propose a combination of object-level attention regions and multi-layer feature map attention regions. The former typically provides fine-grained image representation, while the latter might be more coarse-grained. Through a combination of the two, we construct a richer representation of the image that could lead to improved caption generation.

A further challenge in image captioning is the potentially low diversity within the set of sample captions in a typical training dataset. To address this, Atliha and Šešok [11] proposed using the bidirectional Transformer-based language model BERT [12] to predict randomly masked out words in a sentence, thereby providing synonyms for the masked out words. The training set of captions is thereby expanded, and diversity is increased. We will investigate a similar idea, but instead of just substituting synonymous words, we train a paraphrase generator to rephrase entire sentences (i.e. to provide synonyms for words and to restructure the sentence).

An overwhelming trend in improving the performance of general-purpose image captioning models is to increase the amount of training data and, consequently, the computing resources required for training. Conversely, little work has been done on image captioning with limited training data and resources.

Park et al. [13] proposed a model for chest X-ray report generation in an abnormality detection pipeline. They trained this model on about 7,500 X-ray images, and used a coarse-grained attention mechanism similar to that of Xu et al. [4] on uniformly sized feature map abstractions. Their output domain is rather narrow, whereas our aim is to train a general-purpose image captioning model on a small dataset and compare it to models trained on much larger datasets.

3 Implementation

This section gives details of our proposed model. Firstly, we describe the architecture of the base encoder-decoder model. Secondly, we describe how a joint embedding is created through a concatenation of high-level attention regions from early convolutional layers of a pre-trained CNN, and low-level attention regions from either the bounding boxes of an object detection module or the pixel-level masks of an object segmentation module. Thirdly, we describe our decoder's beam search procedure for language model suppression of semantically unlikely caption candidates. Finally, we describe our approach to caption data augmentation through a paraphrase generator.

3.1 Base Model

Our base model is an encoder-decoder with attention, that jointly learns to align word-to-region mappings in order to generate a descriptive caption for an input image. A basic approach would be to encode the image as a single fixed-sized vector, to serve as a static representation of the image during decoding (caption generation). We make use of an attention mechanism first proposed by Dzmitry et al. [14] for neural machine translation, and adapted for image captioning by Xu et al. [4]. It has been shown that the addition of attention leads to significantly improved performance over the basic encoder-decoder approach. The model encodes the input image as a sequence of vectors, instead of a single vector, and adaptively selects subsets of these vectors during decoding.

The encoder takes an image as input and produces n vectors $\{\mathbf{a}_1, \ldots, \mathbf{a}_n\}$, where each is d-dimensional and corresponds to an attention region of the image. We explore two approaches to find these vectors for a given image, as explained in Section 3.2.

For the decoder we make use of an LSTM network [15] that produces a word at every time step t of the caption generation process. The prediction of a next word is conditioned on a context vector $\mathbf{z}_t$, the previous hidden state $\mathbf{h}_{t-1}$, and previously generated words. The context vector can be a dynamic representation of the image at time t, and in our case is produced by the attention mechanism. This mechanism takes as input the feature vectors $\mathbf{a}_i$ from the encoder and provides a weight $\alpha_{t,i}$ for each, as follows:

$$\alpha_{t,i} = \frac{\exp(e_{t,i})}{\sum_{j=1}^{n} \exp(e_{t,j})} \ , \quad \text{with} \ \ e_{t,i} = f_{att}(\mathbf{a}_i, \mathbf{h}_{t-1}) \ . \tag{1}$$

In the above equation, f_{att} is a multi-layer perceptron conditioned on the previous hidden state $\mathbf{h}_{t-1}$. The weights $\alpha_{t,i}$ are used to create a context vector for time step t:

$$\mathbf{z}_t = \sum_{i=1}^{n} \alpha_{t,i} \mathbf{a}_i \ . \tag{2}$$

The emphasis placed on each attention region is therefore dependent on the sequence of words generated thus far, and informs the LSTM decoder what image content to focus on when generating the next word.

3.2 Multi-level Attention Regions

The soft attention model of Xu et al. [4] makes use of high-level image abstractions from convolutional layers in a pre-trained CNN. This leads to rectangular attention regions of predetermined size, that cannot adapt to the appearance of objects within a particular image. In an attempt to provide richer context to the LSTM decoder, we consider attention regions that contain whole objects and other salient image regions. Two ways of extracting such regions from an image are investigated: bounding boxes from the Faster R-CNN object detection model [16], and pixel-level masks from the Panoptic FCN segmentation model [17].

Due to a potential representation sparseness when considering only object-level attention regions, we propose a joint embedding of these low-level regions and the high-level attention regions from the convolutional layers of a pre-trained CNN. We therefore increase the number of feature vectors produced by the encoder (the $\mathbf{a}_i$ vectors in Section 3.1), for a richer representation of the image. Figure 2 provides a detailed schematic of the proposed model, with the two levels of attention regions that are concatenated and fed to the LSTM decoder.

For high-level attention regions (HLAR), we feed the image through the convolution block of a pre-trained ResNet-152. An early convolutional layer produces a feature map of size $(14, 14, 2048)$ which we flatten to $(196, 2048)$. For

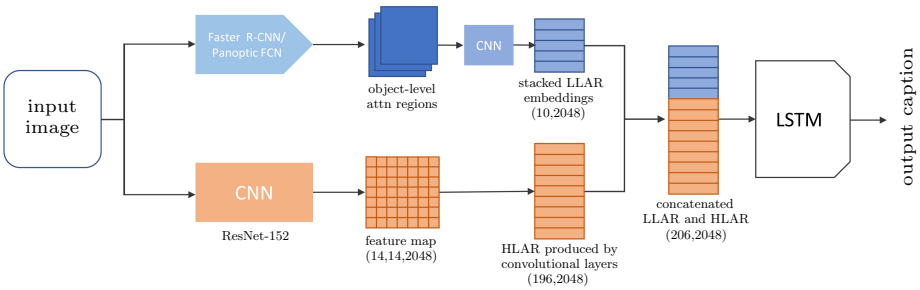

Fig. 2. Pipeline of our image captioning model. The encoder learns a joint embedding of high-level attention regions (HLAR) from ResNet-152 layers, and low-level attention regions (LLAR) from either the bounding boxes of Faster R-CNN or the segmentation masks from Panoptic FCN. This is fed to an LSTM decoder for caption generation.

Fig. 3. Faster R-CNN object detection [16] or Panoptic FCN image segmentation [17] can guide attention regions. The former produces bounding boxes around objects (left), and the latter a pixel-level mask around objects and salient image regions (right).

low-level attention regions (LLAR), we use either a Faster R-CNN model or a Panoptic FCN model. Example outputs of these models are shown in Fig. 3.

Faster R-CNN [16] detects objects in two stages. The first stage outputs object proposals through the refinement of bounding boxes at multiple scales and aspect ratios, and an assignment of class-agnostic objectness scores. Top scoring proposals form input to the second stage, where region-of-interest pooling extracts small feature maps for the classification and further refinement of each bounding box. We take the output from a pre-trained Faster R-CNN model and perform non-maximal suppression on each detected object. We crop out the top 10 bounding boxes (based on their Faster R-CNN classification confidence scores), and feed each one through a pre-trained ResNet-152 network with the final softmax layer removed. This yields 10 feature vectors, each being 2048-dimensional. Panoptic FCN [17] is an efficient fully convolutional network for pixel-level segmentation of an image into foreground objects and background regions. It outputs a mask for each object and region. We overlay each mask on the original image and crop out the tight bounding box. Similar to the above, we feed crops of the 10 highest-scoring objects or regions through ResNet-152 with the softmax layer removed.

Note that the bounding boxes produced by Faster R-CNN are likely to contain irrelevant background pixels. The pixel-level masks from Panoptic FCN blank out pixels that do not belong to a particular object or region class, and might therefore give more precise attention regions for better decoding. However, Panoptic FCN is computationally more demanding than Faster R-CNN.

The final set of feature vectors $\{\mathbf{a}_1, \ldots, \mathbf{a}_{206}\}$ is formed as a concatenation of the HLAR vectors from ResNet-152 and the LLAR vectors from either Faster R-CNN or Panoptic FCN. This set is used in equation (2) to determine a context vector at each time step in the LSTM decoder.

3.3 Language Model Rescoring

During training the LSTM decoder learns to model the conditional probability distribution over the next word given a sequence of already generated words, and a small training set may limit the decoder's abilities for expressiveness and

semanticity. For this reason, we propose the incorporation of a pre-trained language model in the decoder.

In the basic LSTM formulation, words are picked greedily by taking at every time step the most likely word from the softmax output probabilities. Instead, we implement a beam search scheme which expands on all possible versions of the caption. The k most likely expansions are stored at every time step, where k is a hyperparameter that controls the number of beams of parallel searches through the sequence of probabilities. The search process may halt separately for each candidate sequence by reaching either a maximum length, the end-of-sequence token, or a threshold likelihood. We penalise a candidate only when the end token is reached, since the semantic legitimacy of an incomplete sentence is hard to assess.

Our aim is to keep computational requirements low, and therefore make use of GPT-2 [18]. It is a network with 1.5 billion parameters (orders of magnitude fewer than its successor GPT-3), trained on 40GB of Internet text. It uses roughly 3.4GB of memory and takes less than a second to evaluate a given sentence. The semantic legitimacy of a sentence $S = \{w_1, \ldots, w_m\}$ is linked to the probability of the words w_i occurring in a certain order. Autoregressive language models like GPT-2 define the perplexity of S in terms of the negative log-likelihood of each word conditioned on its predecessors in the sequence:

$$\mathrm{ppl}(S) = \exp\left(-\frac{1}{m}\sum_{i=1}^{m}\log p(w_i|w_1,\ldots,w_{i-1})\right). \tag{3}$$

The likelihood function $p(w_i|w_{1:i-1})$ is learned by the language model during training. We note that perplexity is always positive, and that lower values imply higher model confidence in the semantic legitimacy of S.

During beam search, partially generated captions are scored by their cumulative log-likelihood up to the current time step in the LSTM. We penalise the score of a completed caption candidate by subtracting λ times its GPT-2 perplexity, where λ is a hyperparameter (through cross-validation we found $\lambda = 1$ to work well). This suppresses the scores of semantically unlikely candidates, and increases the quality of output captions. We emphasise that rescoring happens only at test time, and does not have any effect on training resources.

3.4 Caption Data Augmentation

The Flickr8k dataset [3] contains 8,000 images, with 5 human-annotated captions for each. As a means of increasing diversity in the set of training captions, we propose a simple data augmentation strategy. We make use of the Text-To-Text Transfer Transformer (T5) framework [19] and train a T5 model specifically as a paraphrase generator using the PAWS dataset [20], which consists of sentence pairs (paraphrases) with low lexical overlap. Different versions of the Flickr8k captions can thus be generated, as illustrated in Fig. 4. In doing so, vocabulary and structure diversity in the training captions are increased, which may lead to a more expressive model.

Original captions from dataset	Paraphrased captions
A child in a pink dress is climbing up a set of stairs in an entry way.	A child dressed in a pink dress climbing a set of stairs in an entry way.
A girl going into a wooden building.	A girl walking into a wooden building.
A little girl climbing into a wooden playhouse.	A girl climbs into a wood playhouse.
A little girl climbing the stairs to her playhouse.	A little girl climbs the stairs to a playhouse.
A little girl in a pink dress going into a wooden cabin.	A little girl dressed in a pink dress is going into a wooden cabin.

Fig. 4. An image from the Flickr8k training set along with the human annotated captions (left column) and augmented captions achieved through paraphrasing (right column).

4 Experiments

In this section we describe the dataset and metrics to quantitatively evaluate the components of our model. We compare our full model with a number of existing models from the literature, when trained on the same dataset and when trained on a much larger dataset. The section ends with a brief qualitative analysis.

4.1 Dataset and Evaluation Metrics

Our models are trained and evaluated on the Flickr8k dataset [3]. It consists of 8,000 images, with 5 human-annotated captions for each, and is relatively small compared to the more widely used MS COCO [2] which contains about 328,000 images. Flickr8k has a standardised training-validation-test split with 6,000, 1,000 and 1,000 images in each set respectively.

For model evaluation we make use of standard image captioning metrics, namely BLEU [21], METEOR [22] and CIDEr-D [23]. The BLEU-n score measures the similarity between reference and generated sentences as the geometric mean of n-gram precision scores, with a penalty for short sentences. BLEU is widely used but has its limitations [24], and additional metrics should be considered. METEOR is the harmonic mean of precision and recall of uni-gram matches between the reference and generated sentence, and may accept synonyms and paraphrases. CIDEr-D measures similarity to a set of references using co-occurrence statistics of n-grams, where $n = 1, 2, 3, 4$. Common n-grams are inversely weighted and a cosine similarity is computed. All metrics except for CIDEr-D range from 0 to 1, with 1 indicating a perfect match. In theory CIDEr-D can reach a maximum of 10, but due to its strictness scores are generally also between 0 and 1. We will print all scores as percentages.

We compare a generated caption to each of the 5 human-annotated captions and take an average of the 5 scores in the case of BLEU, and a maximum in the

case of METEOR and CIDEr-D. Due to the non-standardised use of metrics in the context of image caption evaluation [25], we verified our code with that of cited work [4, 5, 8].

4.2 Quantitative Analysis

Table 1 shows the performance on the Flickr8k test set of our model's components as an ablation study, as well as the full model, compared to models from the literature. Missing values mean that those particular metrics were not reported in the cited papers. Our base model is a re-implementation of the soft attention approach of Xu et al. [4]. We measure the effects of incorporating object-level attention regions into this base model, using Faster R-CNN and Panoptic FCN separately. We also measure the effects of applying language model (LM) rescoring during inference in the base model, as well as caption data augmentation (again on the base model, without any of the other components). Our full model consists of the base model with additional attention regions from Panoptic FCN, language model rescoring during inference, and caption data augmentation.

The inclusion of object-level attention regions brings improvement across all metrics, and indicates that a richer representation of the input image at multiple granularities can be beneficial. The segmentation masks from Panoptic FCN lead to slightly better results over the bounding boxes from Faster R-CNN, probably due to the masks disregarding background information on a finer scale. It may be noted that the additional attention regions lead to the best CIDEr-D scores, across all versions of our model. Incorporating language model rescoring during beam search inference, and augmenting the training set of captions with a paraphrase generator, both lead to marginal improvements over the base model. These relatively small increases in performance may need to be weighed against the additional computational requirements. With the full model combining the base with high- and low-level attention regions, language model rescoring and caption augmentation, increases of 1.6 to 3.3 percentage points over published

Table 1. Versions of our model compared to models from the literature. All models were trained on the Flickr8k training set and evaluated on the Flickr8k test set. BL-n, MTR and CDR are short for the BLEU-n, METEOR and CIDEr-D evaluation metrics.

Model	**BL-1**	**BL-2**	**BL-3**	**BL-4**	**MTR**	**CDR**
Google NIC [8]	63.0	41.0	27.0	-	-	-
Log bilinear [5]	65.6	42.4	27.7	17.7	17.3	-
Soft attention [4]	67.0	44.8	29.9	19.5	18.9	-
Hard attention [4]	67.0	45.7	31.4	21.3	20.3	-
Our base model	66.3	44.2	28.7	20.6	18.6	48.6
Attn: Faster R-CNN	66.5	46.4	32.4	22.4	22.5	**53.5**
Attn: Panoptic FCN	67.1	47.2	33.2	23.2	22.6	**53.5**
LM rescoring	66.8	46.5	32.3	22.4	22.3	51.6
Caption augmentation	66.7	46.2	32.2	22.1	22.5	50.9
Our full model	**68.6**	**48.5**	**34.7**	**24.5**	**23.2**	49.2

results are achieved. Based on current trends in the image captioning literature, these increases are certainly not insignificant.

The use of a small dataset in training a deep neural network can easily lead to overfitting and poor generalisation. For this reason we made use of various regularisation strategies including dropout, small batch sizes, early stopping based on validation BLEU scores, and caption data augmentation. To get a sense of whether overfitting is present in our full model trained on Flickr8k, we plot in Fig. 5 the BLEU, METEOR and CIDEr-D scores achieved by the trained model on the training set and on the test set. A slight tendency to overfitting can be observed, but it does not seem severe. The difference in CIDEr-D scores seems large, but relative to the maximum for that metric (10, or 1000 percent) it is actually also quite small.

Next we compare the performance of our full model trained on 6,000 images from Flickr8k to models from the literature trained on MS COCO which has over 165,000 images in its training set. Results are given in Table 2, where all models are evaluated against the Flickr8k test set. We observe that our full model performs on average within 1.6% of the best model (Hard attention). All models in Table 2 except ours were trained with over 25 times more data than our model. With this in mind, the drop in performance can be deemed small and we achieved our objective of obtaining competitive results.

Table 3 shows the training times required by the variations of our model on the Flickr8k training set, with a single consumer-level GPU (specifically a GTX 1070 Ti). The language model rescoring variant is not included in this table since it does not affect training. It may be observed that, out of all the individual components, the low-level attention regions from Panoptic FCN leads to the highest increase in training time. However, this component also gives the greatest increase in performance over the base model (as shown in Table 1). Note that all versions of our model, including the full one, trains in under 2 hours on a single GPU.

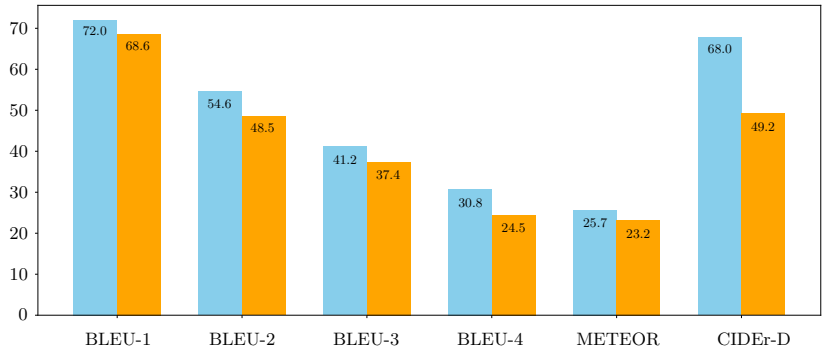

Fig. 5. Evaluation scores that our full model trained on the Flickr8k training set achieves on the Flickr8k training set (blue) and the Flickr8k test set (orange).

Table 2. Our model trained on the Flickr8k training set (6,000 images) compared to models from the literature trained on the MS COCO training set (165,482 images). All models are evaluated on the Flickr8k test set.

Model	BL-1	BL-2	BL-3	BL-4	MTR	CDR
Google NIC [8]	66.6	46.1	32.9	24.6	-	-
Log bilinear [5]	70.8	48.9	34.4	24.3	20.0	-
Soft attention [4]	70.7	49.2	34.4	24.3	23.9	-
Hard attention [4]	71.8	50.4	35.7	25.0	23.0	-
Our full model	68.6	48.5	34.7	24.5	23.2	49.2

Table 3. Training times required by versions of our model, using the Flickr8k training set and running on a GTX 1070 Ti GPU.

Model	Training time
Our base model	1h 01m 01s
Attn: Faster R-CNN	1h 14m 41s
Attn: Panoptic FCN	1h 23m 32s
Caption augmentation	1h 10m 25s
Our full model	1h 51m 25s

4.3 Qualitative Analysis

We visualise in Fig. 6 the attention mechanism as the decoder of our full model generates a caption for a sample image from the Flickr8k test set. At each step of caption generation the LSTM is fed a context vector, hidden state, and the previously generated word. The context vector consists of a weighted sum of attention maps, which in this case stem from a Panoptic FCN segmentation of the input image. We visualise these visual contexts, for an idea of which regions the model deems important while generating words. The weighted attention maps do align to some degree with human intuition. For example, in generating the word dogs the model focuses mostly on pixels belonging to the dogs and ignores all other regions. Similarly, when generating the word water the emphasis is mostly on the water region. Note that this ability is learned from image-caption pairs. The model does not receive any region-specific labels during training.

Contrary to conventional attention mechanisms, our model is able to associate entire objects and sub-object regions to words. Visual information is considered at different levels of granularity, thus avoiding the trade-off between focusing solely on coarse regions and focusing solely on finer regions. Our model can focus on the entire water surface when predicting the word water, and on smaller regions such as the dog's mouth when predicting the word playing.

Figure 7 provides examples of captions generated by our full model. Based on BLEU scores, captions in the top row are "good" while those in the bottom row are "bad". Some of the bad examples (e.g. bottom right) are subjectively accurate, highlighting an inherent limitation in the quantitative evaluation of machine generated text against human-annotated labels.

Fig. 6. A visualisation of the weighted average attention maps considered by our full model during caption generation, for a sample image from the Flickr8k test set.

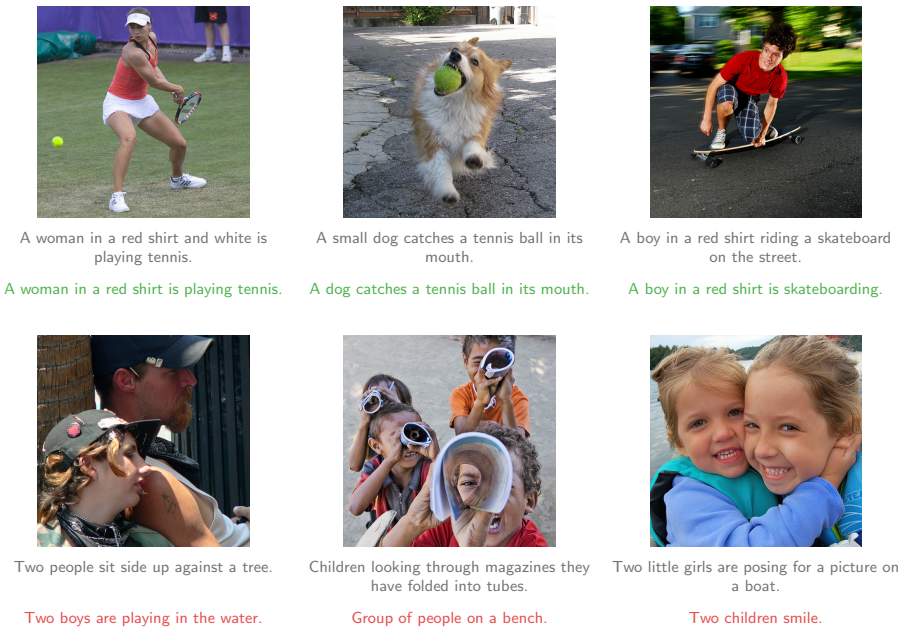

Fig. 7. Examples of captions generated by our full model for Flickr8k test images. Ground truth captions are shown in grey. Generated captions with high and low BLEU scores are shown in green and red, respectively.

5 Conclusion

This paper considered the challenge of training an effective image captioning model on a small dataset and limited hardware resources. We found that by exploiting the modularity of the encoder-decoder architecture, and leveraging off-the-shelf models pre-trained for other tasks, we could increase the perfor-

mance of a baseline attention model by Xu et al. [4]. Firstly, we showed that by representing image attention regions at multiple levels of granularity, richer context could be provided to the decoder. Secondly, we showed that by using a pre-trained language model to suppress semantically unlikely caption candidates in a beam search scheme during inference, better quality captions could be produced. Thirdly, we managed to improve the vocabulary and expressiveness of the model by augmenting training captions with the aid of a paraphrase generator. A combination of all these three ideas led to state-of-the-art results on the Flickr8k dataset, and competitive results compared to models trained on much larger datasets.

Further improvements in terms of training time and memory requirements are possible through the use of mixed precision training [26], which could in turn enable better hyperparameter tuning. In future we also aim to investigate the use of Transformers within an encoder-decoder architecture, and find a meaningful way in which to compare our resource-efficient model with even bigger models like Oscar [1]. Finally, it should be noted that our comparison in Table 2 has its limitations, since the models were trained on samples from (possibly) different data distributions. A fairer comparison might be to train our model on a carefully extracted subset of MS COCO.

References

1. X. Li, X. Yin, C. Li, P. Zhang, X. Hu, L. Zhang, L. Wang, H. Hu, L. Dong, F. Wei, Y. Choi and J. Gao. *Oscar: object-semantics aligned pre-training for vision-language tasks*. European Conference on Computer Vision, 2020.
2. T. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J, Hays and P. Perora. *Microsoft COCO: common objects in context*. European Conference on Computer Vision, 2014.
3. M. Hodosh, P. Young and J. Hockenmaier. *Framing image description as a ranking task: data, models and evaluation metrics*. Journal of Artificial Intelligence Research, vol. 47, pp. 853–899, 2013.
4. K. Xu, J. Beam, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. Zemel and Y. Bengio. *Show, attend and tell: neural image caption generation with visual attention*. Proceedings of Machine Learning Research, vol. 37, pp. 2048–2057, 2015.
5. R. Kiros, R. Salakhutdinov and R. Zemel. *Multimodal neural language models*. International Conference on Machine Learning, 2014.
6. R. Kiros, R. Salakhutdinov and R. Zemel. *Unifying visual-semantic embeddings with multimodal neural language models*. arXiv preprint arXiv:1411.2539, 2014.
7. J. Mao, W. Xu, Y. Yang, J. Wang and A. Yuille. *Deep captioning with multimodal recurrent neural networks (m-RNN)*. International Conference on Machine Learning, 2015.
8. O. Vinyals, A. Toshev, S. Bengio and D. Erhan. *Show and tell: a neural image caption generator*. IEEE Conference on Computer Vision and Pattern Recognition, 2015.
9. W. Cai, Z. Xiong, X. Sun, P. Rosin, L. Jin and X. Peng. *Panoptic segmentation-based attention for image captioning*. Applied Sciences, vol. 10, art. 391, 2020.

10. P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould and L. Zhang. *Bottom-up and top-down attention for image captioning and VQA*. IEEE Conference on Computer Vision and Pattern Recognition, 2018.
11. V. Atliha and D. Šešok. *Text augmentation using BERT for image captioning*. Applied Sciences, vol. 10, art. 5978, 2020.
12. J. Devlin, M. Chang, K. Lee and K. Toutanova. *BERT: pre-training of deep bidirectional Transformers for language understanding*. Conference of the North American Chapter of the Association for Computational Linguistics, 2019.
13. H. Park, K. Kim, J. Yoon, S. Park and L. Choi. *Feature difference makes sense: a medical image captioning model exploiting feature difference and tag information*. Meeting of the Association for Computational Linguistics: Student Research Workshop, 2020.
14. D. Dzmitry, K. Cho and Y. Bengio. *Neural machine translation by jointly learning to align and translate*. International Conference on Learning Representations, 2015.
15. S. Hochreiter and J. Schmidhuber. *Long short-term memory*. Neural Computation, pp. 1735–1780, 1997.
16. S. Ren, K. He, R.B. Girshick and J. Sun. *Faster R-CNN: towards real-time object detection with region proposal networks*. Conference on Neural Information Processing Systems, 2015.
17. Y. Li, H. Zhao, X. Qi, L. Wang, Z. Li, J. Sun and J. Jia. *Fully convolutional networks for panoptic segmentation*. IEEE Conference on Computer Vision and Pattern Recognition, 2021.
18. A. Radford, J. Wu, R. Child, D. Luan, D. Amodei and I. Sutskever. *Language models are unsupervised multitask learners*. Technical Report, OpenAI, 2019.
19. C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li and P.J. Liu. *Exploring the limits of transfer learning with a unified text-to-text Transformer*. Journal of Machine Learning Research, vol. 21, pp. 1–67, 2020.
20. Y. Zhang, J. Baldridge and L. He. *PAWS: paraphrase adversaries from word scrambling*. Conference of the North American Chapter of the Association for Computational Linguistics, 2019.
21. K. Papineni, S. Roukos, T. Ward and W. Zhu. *BLEU: a method for automatic evaluation of machine translation*. Annual Meeting on Association for Computational Linguistics, 2002.
22. S. Banerjee and A. Lavie. *METEOR: an automatic metric for MT evaluation with improved correlation with human judgments*. ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, 2005.
23. R. Vedantam, C. Lawrence and D. Parikh. *CIDEr: consensus-based image description evaluation*. IEEE Conference on Computer Vision and Pattern Recognition, 2015.
24. N. Mathur, B. Baldwin and T. Cohn, *Tangled up in BLEU: reevaluating the evaluation of automatic machine translation evaluation metrics*. Meeting of the Association for Computational Linguistics, 2020.
25. B. Marie, A. Fujita and R. Rubino. *Scientific credibility of machine translation research: a meta-evaluation of 769 papers*. Meeting of the Association for Computational Linguistics, 2021.
26. P. Micikevicius, S. Narang, J. Alben, G. Diamos, E. Elsen, D. García, B. Ginsburg, M. Houston, O. Kuchaiev, G. Venkatesh and H. Wu. *Mixed precision training*. International Conference on Learning Representations, 2017.