Final Report: Image Captioning Project

Methods and Applications in the Digital Humanities

Caroline Berg*, Cian Dadgar†
(*cb63myxu@studserv.uni-leipzig.de|†117478762@umail.ucc.ie)

Fakultät für Mathematik und Informatik
Universität Leipzig

February 9, 2020

Abstract. We trained a German and English image captioning model with different settings. Although we obtain low results, our models are able to produce grammatical captions that at times describe the pictures quite well, given that our approach is fairly simple and the size of the training set is quite small. We start this report with a general introduction and definitions of terminology and present the results for the different training settings. We conclude with an analysis and explanation of reoccurring problems.

1 Introduction to Image Captioning

Image captioning is the process of automatically assigning captions to images. Among other commercial usage, it is adopted by social media platforms and search engines to facilitate search requests, to gather metadata in an unsupervised manner, or to make content accessible for people with special needs.
On a technical basis, neural image captioning is a generative task, where an encoder structure encodes a numeral representation if the input image. It is then forwarded to a decoder structure which calculates word probabilities over a pre-defined vocabulary step by step, by conditioning the decoder on the encoder output and the sequence of words created up to the current time-step t_i. During training, the current sequence is usually updated to the reference caption up to t_{i-1}, which is provided by the training set. The loss is then calculated by comparing the produced sequence (=*hypothesis*) and the assigned caption (=*reference*). During interference, the actual sequence produced by the decoder model is used for predictions at time-step t_i.

2 Terminology

The following section gives a short definition of the terminology used in this report.

2.1 Neural Layer Architecture

Dropout Dropout as explained by [11] is a way of applying regularization to a neural network structure during training. Since neural networks are proven to perform excellent on pattern recognition, they are always prone to over-fitting to a specific task, especially when running multiple epochs on the same small data set. When dropout is applied to a layer, it means that random units are zeroed-out with probability p, as depicted in Figure 1. Doing this for multiple iterations essentially means that numerous networks which are very similar are trained on the same data. Finally, they are combined during interference.
[7] points out that dropout also prevents co-adaption of cells in a neural network, since each unit can not rely on the output of other units and in turn is forced to adapt in a more stable manner to the training data.

LSTM Long short-term memory cells, in short LSTMs, are a special type of recurrent network units. They are often chosen for language generation or sequence prediction tasks, because their inherent structure enables them to store long-term dependencies between former inputs and the current input. An LSTM-cell is composed of four gates, namely the input gate, output gate, forget gate and the current cell state. They control in which manner new and old information are combined with each other. We point to the famous paper of [6] for more information.

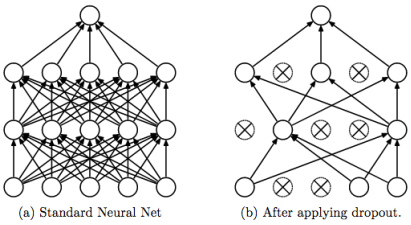


Fig. 1: Depiction of dropout in a neural network structure from [11] (Figure 1).

Attention The idea of integrating an attention mechanism into neural network architectures was inspired by the fact that, when looking at a picture, we focus on specific parts or items displayed in it, depending on the type of information we wish to retrieve from it. Similarly attention enable a neural network to focus on specific parts of the input when calculating the next state. The general notion of attention is to map a query and a set of key-value pairs to an output, where queries, keys, values and output are all vectors. The output is computed of the weighted sum of weights assigned to each value, which is computed by a compatibility function of the query q with the corresponding key-value pairs (K, V).

$$\text{A(q, K, V)} = \sum_i \frac{exp(e_{qk_i})}{\sum_j exp(e_{qk_j})} v_i$$

There are many possible ways to compute attention[1]. We chose the implementation provided by the KERAS-SELF-ATTENTION.

2.2 Activation Functions

In our model architecture we use two types of activation functions.
The first on is the Rectified Linear Unit (ReLU function):

$$f(x) = max(0, x)$$

ReLU returns zero if the value is negative and otherwise the input value. The ReLU function and other variations of it are very popular. The function does not saturate when the input gets large, since it is linear.It is also easy to compute and thus does not hinder the speed of the training process.
The second activation function we use in our model is called the Softmax function. Given an input vector the Softmax function normalises the vector to a probability distribution. Assuming a vector of size K we get:

$$f(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{K} e^{x_j}}$$

In our use-case the Softmax function enables the model to choose the vocabulary index with the highest probability to generate the next output of the caption.

[1] See this blogpost for a comprehensive overview: https://lilianweng.github.io/lil-log/2018/06/24/attention-attention.html#attention-mechanisms

2.3 Optimization

Categorical cross-entropy loss The categorical cross-entropy loss is commonly applied as image captioning loss, since it evaluates multi-class classification with probability outputs. It is also known as the Softmax loss because it is a combination of the Softmax function and the cross-entropy loss. We use one-hot encoding for the multi-class classification case so the loss can be defined as :

$$CE = -log\frac{e^{x_p}}{\sum_j^C e^{x_j}}$$

with x_p being the model score for the positive value and C the number of classes, i.e. the size of our vocabulary[2].

2.4 Evaluation scores

There are a number of evaluation scores to choose from when evaluating a natural language generation model, where a ground-truth sequence is evaluated against the model output.
Bleu measures precision ([9]). It compares the n-gram overlap of the hypothesis with the reference text.
Rouge measures recall ([8]). It measures how much the n-grams of the reference text appeared in the hypothesis.
The CIDEr score was specifically developed for image captioning ([13]). The first step is to stem all words inside the reference and hypothesis. For each n-gram in the captions a TF-IDF weighting is computed, to give more common n-grams a lower weight than infrequent ones. The CIDEr score is finally computed as the cosine similarity between hypothesis and reference.
Other popular metrics that we did not use in our evaluation of the models are Meteor ([2]) and Spice ([1]).

2.5 Beam Search

When generating language through a probability distribution during interference, the best guess is usually greedy search. At each time-step the word with the highest score from the vocabulary is chosen. However the greedy approach is not ideal. A word that at time-step t_i gets lower probability than another word, might lead to an overall better hypothesis at a later time-step. Also greedy search often yields faulty grammar with lots of repetitions of single words or n-grams that have a high score.
Beam search is a remedy to this problem. A beam of size $n \geq 2$ is defined, depending on the computing capacity. Common values are $3 \leq n \leq 5$. While producing the sequence n hypothesis are stored and each hypothesis is expanded with the entire vocabulary. Finally we keep n hypothesis with the highest score from the model at each time-step.

[2] We refer to https://gombru.github.io/2018/05/23/cross_entropy_loss/ for an explanation.

3 Model Architecture

Our approach for image captioning is based on a tutorial by Harshall Lambda[3]. We deployed the model on Google Colab[4] for GPU access and we then modularised and adapted the code.
For the evaluation of our models we adopted an implementation[5] of the standard score set by [4].
We trained two models for each language, to be able to observe the potential improvement in performance for image captioning by adding more trainable parameters to the architecture.
Both models are composed of a combination of pre-trained Fasttext word embeddings by [3] and the pre-trained InceptionV3 model by [12] for image encoding, which can be imported via the Keras library[6]. The encoder-decoder architectures of both models are described in the following subsection.

3.1 Base Model

We refer to this architecture as the base model. The encoder of both models is two-fold, since both the image and word input for the task need to be encoded. We take the image encoding from the InceptionV3 model of size 2048, add dropout of 0.5, and project to size 256 via a dense layer with ReLU-activation. The input for the word encoding is set to the maximum length of a caption inside the training and development set. This enables fixed size encoding via zero masking. The word input is then forwarded to a matrix of size $(vocab\_size, embedding\_dimension)$ to map to the pretrained word embeddings. This particular layer is fixed during training (i.e. not updated) to avoid catastrophic forgetting, since we are not working with a large amount of text data. We add dropout and an LSTM-layer of size 256. The final part of the encoder is the addition of both image and word representation.
The decoder is a simple dense layer of size 256 and ReLU-activation. The final layer is a projection to the vocabulary size followed by the Softmax-function to obtain probabilities over all the words.

3.2 Advanced Model

The second model, which we will refer to as the advanced model, shares all architectural compontents with the base model, except for the encoding of the word input. To expand the learning capabilities for this representation, we add an LSTM-layer of size 512 with self attention after the fixed-embedding layer with dropout. We add another LSTM-layer of size 256 before the addition of the image representation.

4 Data and Training

We separately train a German and English captioning model on the FlickR30k dataset, introduced by [10]. It consists of 30.000 images with descriptions in both English and German. We split the dataset into training-, development- and testset (80%/10%/10%). Table 1 displays the counts necessary for the model construction for both languages.

	English	German
Vocabulary Size	2115 (+6 OOV[7])	2048 (+173 OOV)
Maximum caption length	35	41

Table 1: Number of parameter for each model setting.

[3] See https://github.com/hlamba28/Automatic-Image-Captioning
[4] https://colab.research.google.com
[5] https://github.com/tylin/coco-caption
[6] https://keras.io/

The numbers indicate, that the task for German image captioning is much more complex, as there are more words, which can not be set to a pre-trained embedding. Also, captions are required to be longer, i.e. more outputs need to be produced by the decoder. It is generally known that solving natural language processing tasks in German is much more complex than in English. Among other grammatical properties, this is due to more complex inflection and frequent use of composita.
We train all models on batch size 32 with the Adam optimizer and categorial cross-entropy loss. The learning rate for the base model is set to 0.0001. For the advanced model setting we experiment with learning rate decay by a factor of 0.001 starting with learning rate 0.01 to speed up the training process and to avoid overfitting. All models are trained for 10 epochs.
This small scale training setting is only possible, since we are using pre-trained word embeddings and a readily available image encoding model for the encoding part of the whole structure. Table 2 shows the number of parameters for each model setting. The number of parameters for the English models is generally larger, since the encoder part that handles the word embeddings holds a larger vocabulary.

	English parameters	German parameters
Base model	2.338.759	2.329.290
Advanced model	4.483.016	4.473.547

Table 2: Number of parameter for each model setting.

The loss plots in Figure 2 indicate, that the base model architecture is not suitable for the task of German image captioning. However the loss can be decreased significantly for both the English and the German task when applied to the advanced model structure. The next section will describe the results obtained on the test set and discuss the performance measured by a number of scores developed for evaluation of natural language generation.

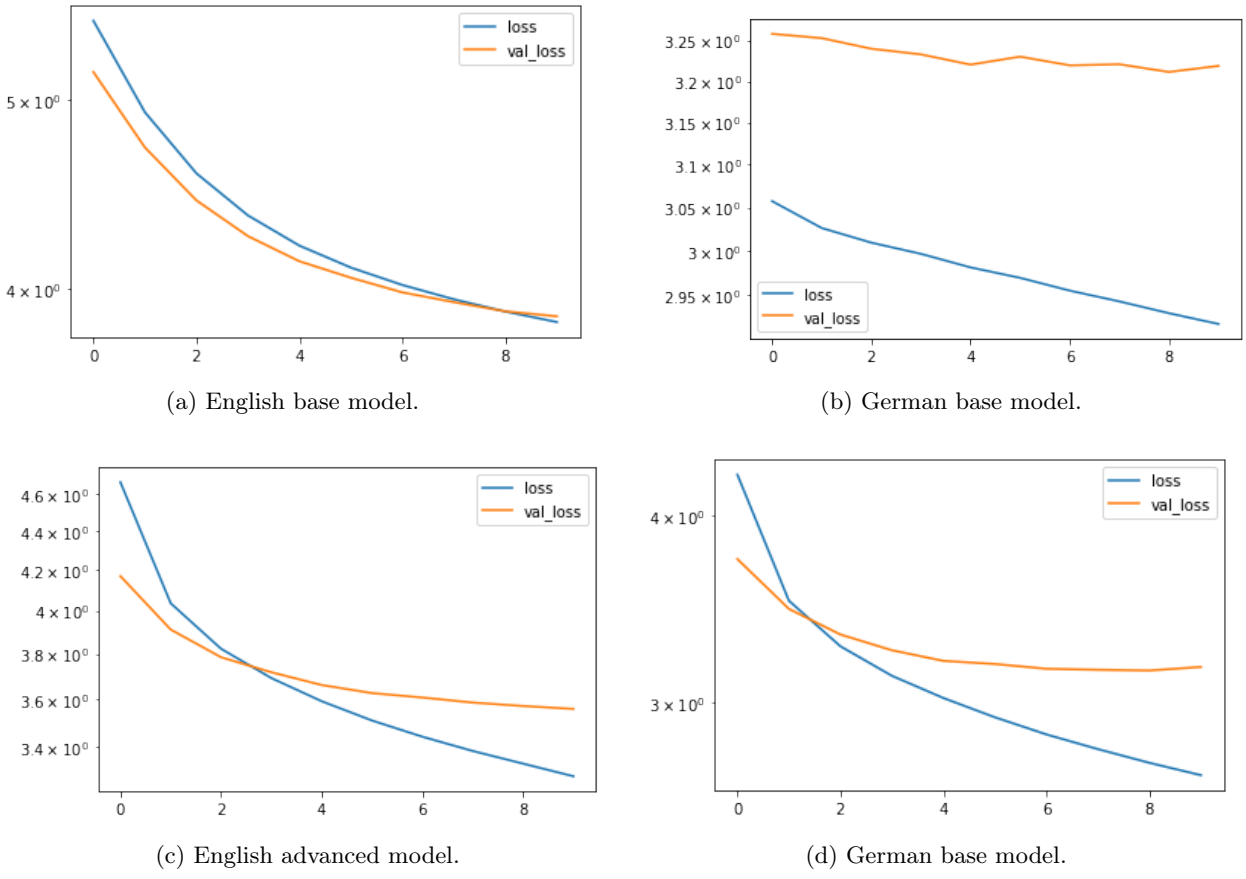
5 Results

We present the results obtained on the test sets in Table 3. The right score was obtained by producing the caption in a greedy manner and the left score shows the results for our beam search algorithm. We chose a beam of size 4 to speed up the generation process and modified it with some restraints. First we require each finished hypothesis to end on a non-stopword, as we discovered that some generated captions where ungrammatical. We also filter hypothesis which contain repetitions of bi-grams, tri-grams and immediate repetitions of uni-grams. We also experimented with a length score but could not obtain any improvement.

	English base model	German base model	English advanced model	German advanced model
Bleu-2	7.1/7.7	9.7/12.6	9.1/8.1	10.8/12.2
Rouge	16.4/18.6	15.9/21.4	18.2/18.0	0.176/0.210
CideR	26.1/29.6	21.9/28.9	27.4/17.9	20.9/21.0

Table 3: Results on the test set. Greedy search is displayed on the left and beam search on the right side.

As we can see from the scores in Table 3, results generally improve when using beam search as oppose to greedy search except for the English advanced model. We suppose that, as the general performance of the model increases (see Figure 2c), more tailored measurements would be required, such as encouraging the



(a) English base model.

(b) German base model.

(c) English advanced model.

(d) German base model.

Fig. 2: Training and evaluation loss.

use of infrequent words and shortness penalty. The margin of improvement is large for the German models, which further indicates that positive outcomes with restrictive measures for the generation process does not generally apply to all languages.
The maximum achievable score for all measures we report is 100.0 and well performing image captioning models usually achieve scores between 70.0 to 80.0. We thus note that our model performance is generally low. However, we achieve the highest values with the CideR score, which was specifically developed for image captioning evaluation. The bleu score is known to not correlate well with human perception and rouge is generally biased towards longer sentences. Bleu and rouge can be seen as complementing scores. Bleu measures precision and rouge measures recall. We tend to achieve higher rouge score than bleu throughout the different settings, which means that there is a higher overlap between the words occurring in the human references to the models hypothesis, than the other way around. We receive this as a positive observations and assume, that more improvements can be achieved with a deeper model structure, which will in turn also require more computational power.

5.1 Qualitative evaluation

Two examples from the test set are depicted in Figure 3. We observe that beam search does not improve performance for the English models in these cases. For the German models greedy search often generates ungrammatical captions. Here we see some improvement with the beam search strategy.
When the models fail to encode the image features correctly they generally predict phrases like *a man in blue shirt ...* (German: *ein mann in einem blauen oberteil ...*). We can also see differences between the base and the advanced model architectures, which we interpret as a proof, that the outcomes of image captioning models can be drastically changed by small changes in the parameters of the model.

5.2 Out-of-domain testing

To further evaluate our approach and show a potentially useful practical application, we tested our trained models on a number of paintings that we manually retrieved via Google image search. The paintings and captions generated by the German and English advanced architecture can be found in the appendix (section 6). In Figure 4 we show some examples, where our models could not capture what is depicted in the painting. Of course it is hard to decode the meaning of a distorted or abstract painting. In addition to that, our models were trained on data which mostly shows people doing things, like running, sitting or playing with a ball. Hence, they can not identify animals or species other than dogs or cats.
More promising results are shown in Figure 5. Although our models have a lot of trouble recognizing color and fail to describe some items due to limitations of the vocabulary[8] we see some potential in our approach. Without fine-tuning or ever being trained on paintings our models (sometimes) succeed in labelling the paintings with a reasonable caption.

6 Conclusion and future work

With this project we demonstrated, that image captioning can be achieved with relatively little training data when using pre-trained models for the image and word feature encoding. Although the score results for all architectures in both languages are quite low, we achieved some improvements by introducing an attention mechanism to the embedding encoder. We expected worse performance for the German models, however they are mostly able to generate grammatical an reasonable captions for the test images.
For future work we want to investigate the relatively poor performance for the English model and conclude, that a different approach in model architecture and beam search might be required. After all English and German have different characteristics. For example English captions are generally shorter, while having a

[8] For example mistaking a pitchfork for a microphone.

(a)

German base model

Greedy: ein mann mit freiem oberkörper und einem blauen oberteil sitzt auf einem

Beam: ein mann mit freiem oberkörper sitzt auf einem stuhl

English base model

Greedy: man in blue shirt and white shirt is sitting on the street

Beam: man in blue shirt and white shirt is sitting on the street

German advanced model

Greedy: ein mann sitzt auf einem stuhl und verkauft ein

Beam: ein mann sitzt auf einem stuhl

English advanced model

Greedy: man in blue shirt is sitting on the ground in front of large building

Beam: man in blue shirt is sitting in front of large building

Actual title:

German: drei leute sitzen an einem picknicktisch vor einem gebäude das wie der union jack bemalt ist

English: three people sit at picnic table outside of building painted like union jack

(b)

German base model

Greedy: ein mann in einem blauen oberteil läuft am strand entlang

Beam: zwei personen spielen am strand

English base model

Greedy: man is running on beach

Beam: man is running on beach

German advanced model

Greedy: ein mann in einem blauen oberteil und shorts läuft am strand entlang

Beam: eine gruppe von menschen läuft am strand entlang

English advanced model

Greedy: two people are playing in the sand

Beam: two people are playing in the sand

Actual title:

German: vier personen spielen fußball auf einem strand

English: four people are playing soccer on beach

Fig. 3: Examples from the test set.

higher variance in vocabulary use. We think that introducing contextualised word embeddings with an encoder similar to BERT ([5]) could further improve the performance of the model, since the language encoding and decoding part is usually harder to solve for neural networks than image encoding.
Finally we want to point out a potential use-case for image captioning models in digital archives. As we demonstrated in the last subsection, models trained solely on pictures can be applied to paintings and give reasonable output. This phenomenon can be leveraged via transfer learning, where a model is pre-trained on a large number of annotated images and successively fine-tuned on paintings of a digital archive. The output of the fine-tuned models can then be used as a search-function to retrieve images of interesting topics by matching keywords. It would also be interesting to see, if an image captioning model can be fine-tuned on a slightly different task, like describing stylistic features of a painting. We leave this task up for future work.

References

1. Anderson, P., Fernando, B., Johnson, M., Gould, S.: Spice: Semantic propositional image caption evaluation. In: European Conference on Computer Vision. pp. 382–398. Springer (2016)
2. Banerjee, S., Lavie, A.: Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In: Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization. pp. 65–72 (2005)
3. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics **5**, 135–146 (2017)
4. Chen, X., Fang, H., Lin, T.Y., Vedantam, R., Gupta, S., Dollár, P., Zitnick, C.L.: Microsoft coco captions: Data collection and evaluation server. arXiv preprint arXiv:1504.00325 (2015)
5. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
6. Gers, F.A., Schmidhuber, J., Cummins, F.: Learning to forget: Continual prediction with lstm (1999)
7. Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.R.: Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580 (2012)
8. Lin, C.Y.: ROUGE: A package for automatic evaluation of summaries. In: Text Summarization Branches Out. pp. 74–81. Association for Computational Linguistics, Barcelona, Spain (Jul 2004), https://www.aclweb.org/anthology/W04-1013
9. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th annual meeting on association for computational linguistics. pp. 311–318. Association for Computational Linguistics (2002)
10. Plummer, B.A., Wang, L., Cervantes, C.M., Caicedo, J.C., Hockenmaier, J., Lazebnik, S.: Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In: Proceedings of the IEEE international conference on computer vision. pp. 2641–2649 (2015)
11. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research **15**, 1929–1958 (2014), http://jmlr.org/papers/v15/srivastava14a.html
12. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2818–2826 (2016)
13. Vedantam, R., Lawrence Zitnick, C., Parikh, D.: Cider: Consensus-based image description evaluation. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2015)

Appendix

(a)

German advanced model

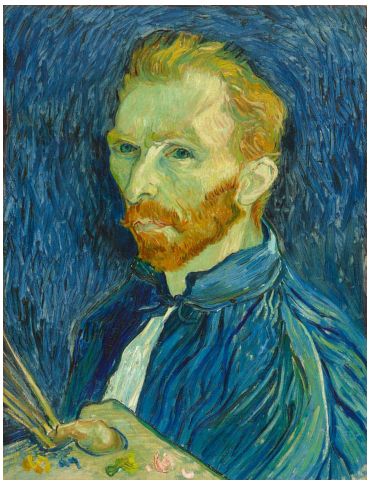
Greedy: -

Beam: ein mann in einem blauen oberteil spielt

English advanced model

Greedy: two men are playing game of martial

Beam: two men are playing game of martial



(b)

German advanced model

Greedy: ein mann mit einem roten hut und einem weißen oberteil sitzt auf einem sofa

Beam: ein kleines mädchen in einem rosa kleid sitzt auf einer bank

English advanced model

Greedy: man in blue shirt is sitting on the ground with his eyes closed

Beam: man in blue shirt is sitting on the ground

(c)

German advanced model

Greedy: ein mann mit einem roten oberteil und einem weißen oberteil steht auf einem

Beam: ein mann mit einem roten oberteil und einem weißen oberteil steht auf einem braunen teppich

English advanced model

Greedy: man in blue shirt is sitting on the ground

Beam: man in blue shirt is sitting on the ground

Fig. 4: Out-of-domain testing: Paintings that were not be captioned well.

(a)

German advanced model

Greedy: ein mann mit brille und einem schwarzen oberteil spricht in ein mikrofon

Beam: ein mann in einem anzug spricht in ein mikrofon

English advanced model

Greedy: **man in black shirt and blue pants is standing in front of microphone**

Beam: man in black shirt is holding microphone

(b)

German advanced model

Greedy: ein mann in einem roten oberteil sitzt auf einem sofa und hält ein buch

Beam: **ein junges mädchen sitzt auf einem sofa**

English advanced model

Greedy: woman in blue shirt and blue pants is sitting on the ground

Beam: woman in blue shirt is sitting on the ground

(c)

German advanced model

Greedy: **ein mann in einem weißen oberteil und shorts läuft durch den sand**

Beam: ein mann in einem weißen oberteil läuft durch den sand

English advanced model

Greedy: man in blue shirt and jeans is walking on the beach

Beam: man is walking on the beach

Fig. 5: Out-of-domain testing: Paintings that were not be captioned (surprisingly) well.