# Contextual knowledge is important: utilizing top-down information in generating structured and ordered image paragraphs

Anonymous ACL submission

## Abstract

abstract

## 1 Introduction

Humans are typically able to effortlessly describe real-world images when required: we easily identify objects, attributes and relations between them. Diversity, richness and complexity of such human-produced image descriptions have been observed in several benchmark image description datasets, including MSCOCO (Lin et al., 2014; Chen et al., 2015), Flickr8k (Hodosh et al., 2013), Flickr30k (Young et al., 2014), Visual Genome (Krishna et al., 2016). These datasets were collected to address the task of automatic image description (Bernardi et al., 2016), a long-standing and active field of research, placed in intersection between computer vision and natural language processing (generation, in particular). This problem of mapping visual data to text can be viewed as the specific example of one of the core goals of NLG: 'translating' source data into a natural language representation.

In natural language generation community, the task of text generation has been typically divided into multiple sub-tasks, including *content determination (selection)*, the task of deciding which parts of the source information should be included in the output description, and *text structuring*, the task of ordering selected information (Gatt and Krahmer, 2017). However, with the rise of neural networks in many NLP areas, the generation tasks are now seen as a continuous, non-modular process of automatically learning relations between input and expected output. Specifically, neural models of image captioning (Kiros et al., 2014; Vinyals et al., 2014) are trying to implicitly learn what is important about an image (content selection) and how this information should be structured in the generated caption (text structuring). Such mechanisms as attention (Xu et al., 2015; Anderson et al., 2017) further improve ability of the models to locate important parts in an image and utilize them for caption generation. Some recent advances in image captioning include application of transformer architecture (Vaswani et al., 2017; Herdade et al., 2019).

While it is clear that neural networks demonstrate good performance in generating *well-structured* single-sentence image captions with *relevant knowledge*, the problem of selecting and ordering information becomes significantly harder when generating multiple sentences for a single image. The corresponding task of *image paragraph generation* has been initially introduced in Krause et al. (2017), proposing the challenge of generating a text, consisting of several sentences that would form a coherent whole. Most of the following work (Liang et al., 2017; Chatterjee and Schwing, 2018; Wang et al., 2019) has focused on *generating* good, diverse and human-like paragraphs as measured by various automatic evaluation metrics like BLEU (Papineni et al., 2002) or CIDEr (Vedantam et al., 2014).

In this paper we look at the different aspect of image paragraph generation and address the problem of **information order** in the multi-sentence image captioning setting. We argue that utilizing top-down knowledge (information about context available to the captioner) is beneficial for the task of image paragraph generation. We show that the model conditioned on both low-level visual features and high-level top-down information is able to learn human-like distributions of attended objects, attributes and relations generated in the paragraph. We introduce several image paragraph models based on the hierarchical paragraph generator Krause et al. (2017) and also demonstrate that using bottom-up information exclusively is not enough to learn good paragraph structure. We employ transfer learning and use model pre-trained for the dense im-

age captioning task (Johnson et al., 2016) to obtain representations of background information, which we treat as our top-down features. We evaluate how close our models are compared to the human performance in terms of attending to objects in visual scenes in a particular order.

## 2 Related Work

[Nikolai: This whole section needs to be shrinked. Or not?]

**Human visual attention** Humans are quite efficient in detecting objects and separating them from the rest of the visual scene (Ullman, 1987). We are also fluent in using visual attention, which allows us to single out particular objects in the visual scene, significantly reducing our perceptual load when needed (Lavie et al., 2004), therefore, preventing us from being overwhelmed by typically complex real-world visual scenes. Our ability to attend to particular parts of the environment is based on both bottom-up information (low-level visual stimuli) and top-down information (high-level goal-related stimuli, discourse knowledge) (Zarcone et al., 2016). Stimuli that attracts our attention is said to be *salient* (relevant). Salience of objects affects our *surprisal* towards particular visual input: discourse-salient entities cause less surprisal (e.g. 'bed' in bedroom) unlike the visually salient objects (e.g. 'large pink elephant' in bedroom).

Attention has also been employed in formal theories of interaction. One of the approaches has been proposed by Dobnik and Kelleher (2016), who link attention with judgements as defined in Type Theory of Records (Cooper, 2008). They introduce a Bayesian-based framework in which attention controls the extent to which context induced judgements ($\sim$ task-based top-down information) are utilized by an agent. This allows for topic modelling at each timestamp in interaction. Thus, it follows along the lines of our proposal about attention using both contextual and low-level visual information in selecting relevant information for each individual sentence in the image paragraph.

**Neural image paragraph captioning** The task of generating more complex descriptions of images such as paragraphs has been introduced in Krause et al. (2017) along with the dataset of image-paragraph pairs. The paper adopts a hierarchical structure for the model of paragraph generation: sentence RNN is conditioned on visual features and unrolled for each sentence in the paragraph, giving a sentence topic as its output. Then, each of these topics is used by another RNN to generate actual sentences. We start by implementing this hierarchical image paragraph model, since it inherits the modular nature of human image paragraph production (given an image, plan structure of your paragraph and identify its sentence topics, then incrementally generate sentences). Liang et al. (2017) use similar hierarchical network in addition with adversarial discriminator, that forces model to generate realistic paragraphs with smooth transitions between sentences. Chatterjee and Schwing (2018) also address cross-sentence topic consistency by modelling global coherence vector, conditioned on all sentence topics. Different from these approaches, Melas-Kyriazi et al. (2019) employ self-critical training technique (Rennie et al., 2016) to directly optimize a target evaluation metric for image paragraph generation. Lastly, Wang et al. (2019) use convolutional auto-encoder for topic modelling based on region-level image features. They demonstrate that extracted topics are more representative and contain information relevant for sentence generation. In this paper we similarly model better topic representations. However, we use additional language representations as part of the input to our topic generator, which is an LSTM.

**Language attention in language and vision models** [Nikolai: wordy section, needs to be shorter, keep all information here for now]

Only a limited number of models for image captioning has been supplied with both visual and background information for caption generation. You et al. (2016) detect visual concepts found in the scene (objects, attributes, etc.) and extract top-down visual features. Both of these modalities are then fed to the RNN-based caption generator. Attention is applied on detected concepts to inform generator about how relevant a particular concept is at each timestamp. Different to their model, we do not use any attribute detectors to identify objects in the scene, instead relying on the output of the model pre-trained for the task of dense captioning. Lu et al. (2016a) emphasize that image is not always useful in generating some function words ('of', 'the', etc.). They introduce adaptive attention, which determines when to look at the image and when it is more important to use the language model to generate the next word. In their work, the

attention vector is the mixture of visual features and visual sentinel, a vector obtained through the additional gate function on decoder memory state. Our model is focused on a similar task: we are interested in deciding which type information is more important at a particular timestamp, but we also look at how *merging* two modalities into a single representation performs and how it affects attention of the model. Closest to our work is the work by (Liang et al., 2017), who apply language attention on region captions and use it to assist recurrent word generation in producing sentences in a paragraph. They embed region descriptions into the same embedding space that their word RNN is operating on. While we also believe that feeding information about semantic concepts found in an image is beneficial for the model, we propose to employ transfer learning. We use hidden states of the RNN trained for the task of dense captioning (Johnson et al., 2016) as our background information representation. Outside of image paragraph captioning, Lu et al. (2016b) have proposed a joint image and question attention model for the task of visual question answering. [Nikolai: Any work on language attention in visual dialog? I think one sentence with some citations would be nice to have.]

## 3 Approach

**Overview** As our base model in the experiments, we implement the hierarchical image paragraph generation model by (Krause et al., 2017). We change most parts of this model when implementing various model configurations, which we describe below. To identify image regions and extract their corresponding features, We also utilize the pre-trained model for dense captioning (Johnson et al., 2016). For our language representations, we use hidden states from the last layer of the DenseCap RNN, which is supposed to generate region descriptions in the original architecture. We fuse both modalities into a single vector represented by an affinity matrix, similar to Xu and Saenko (2015) and Lu et al. (2016b). This matrix captures similarities between visual and language representations for all combinations of regions. Then, the similarity vector is used by a two-level hierarchical paragraph generator. First, the sentence-level LSTM transforms its input into the sentence topics, capturing information flow between sentences. Second, each of the topics is employed by word-level LSTM to generate a sentence. Finally, all generated sentences per image are concatenated to form a final paragraph. An overview of our model is shown in Fig. We also deploy various strategies for decoding and analyse differences in corresponding generated paragraphs. [Nikolai: model structure scheme is in progress, not sure there will be space for decoding strategies though]

Note that we were not able to acquire a source code for the original hierarchical model. Therefore, our model's performance in terms of automatic evaluation might not necessarily be comparable to the one described in Krause et al. (2017). Though we report automatic evaluation scores, in this paper, we instead focus on showing that background knowledge incorporated in the paragraph generation model leads to more detailed and diverse image descriptions.

### 3.1 Input Features

**Visual Features** Here, we describe our image representation in terms of its regions. We first dissect our image into multiple salient regions and extract their corresponding convolutional features, using the region detector introduced for the task of dense captioning (Johnson et al., 2016)[1]. procedure: first, First, a resized image is passed through the VGG-16 network (Simonyan and Zisserman, 2014) to output a feature map of the image. A region proposal network is conditioned on the feature map to identify the set of salient image regions, which are then mapped back onto the feature map to provide us with corresponding map regions. Each of these map regions is then fed to the two-layer perceptron to obtain a set of the final region features $\{v_1, ..., v_M\}$, with each feature dimension being $1 \times D$. Finally, each image is represented by the matrix of the region features $V \in \mathbb{R}^{M \times D}$, which are passed through the simple feed-forward layer $W_v \in \mathbb{R}^{D \times H}$ followed by ReLU non-linearity. These visual feature vectors provide us with fine-grained image representation on the object level.

**Language Features** As a part of the dense captioning task, a single layer LSTM (Hochreiter and Schmidhuber, 1997) is conditioned on region features to produce descriptions of these regions in natural language. We propose to utilize its outputs as language features, which provide us with additional information about detected objects. Specifically, we condition pretrained LSTM on region features

[1] Available at: https://github.com/jcjohnson/densecap

to obtain a set of outputs $Y = \{y_m, ..., y_M\}$, where $y_m \in \mathbb{R}^{1 \times T \times H}$. We use mean pooling over the $T$ dimension, which determines a number of words in each description and receive a single representation per region. Our final matrix of language features per image $L \in \mathbb{R}^{M \times H}$ captures semantic information about objects from detected regions. [Nikolai: do not forget to mention RNN-GAN work on paragraphs, which also uses densecap regions, but they learn embeddings of words in phrases directly]

**Multimodal Features** We wish to leverage language features in our paragraph generation model that is conditioned on visual information. Similar to (Lu et al., 2016b), we teach our model to co-attend to both of these modalities simultaneously. Specifically, we use our region feature map $V$ and language features $L$ to compute the following matrix $C$:

$$C = ReLU(L^T W_v V), \quad (1)$$

where $W_v$ is used to learn a mapping from vision space to language space. The final multimodal vector $C \in \mathbb{R}^{M \times H}$ is used as a multimodal input to our paragraph generator.

### 3.2 Sentence LSTM

Our sentence-level LSTM is responsible for modeling topics of each of the individual sentences in the paragraph. At each timestamp, it is conditioned either on visual or multimodal features, and its output is a set of hidden states $\{h_1, ..., h_S\}$ of length $S$, where each state is used as an input to the word-level LSTM. In its nature, sentence LSTM has to simultaneously complete at least two tasks: produce a topic with relevant information for each sentence, while preserving some type of *ordering* between topics. Such topic ordering is essential for keeping a smooth transition between sentences (discourse items) in the paragraph (mini-discourse).

To assist sentence LSTM in its multiple objectives, we propose to use attention on the sentence topic level. Attention alleviates the task of weighing specific parts of the input as more important for a particular sentence, thus allowing sentence LSTM to learn more precise sentence representations and sentence order. In particular, we use soft version of global attention as introduced in (Bahdanau et al., 2014) and applied in image captioning (Xu et al., 2015; Luong et al., 2015).

[Nikolai: I separate between sentence inputs and word inputs by using varsigma for sentence input, and omega for word input, original input is simply x] Let $\oplus$, $\sigma$ and $\odot$ denote concatenation, logistic sigmoid function and element-wise multiplication respectively. At each time step our attention module receives a feature vector $x$ and previous hidden state $h^{\varsigma}_{s-1}$ of the sentence LSTM to produce attended input features $\hat{x}^{\varsigma}_s$. With both $W_m$ and $W_a$ denoting trainable parameters, attention first computes the attention weights $\alpha_{i,s}$ for each element $x_i$ in the input feature $x$ using additive (concatenative) alignment function as follows:

$$a(x, h^{\varsigma}_{s-1})_{i,s} = W_a^{\top} \tanh(W_m[x_i \oplus h^{\varsigma}_{s-1}]) \quad (2)$$

$$\alpha_s = \frac{exp(a_{i,s})}{\sum_j^M exp(a_j))} \quad (3)$$

Then, the sum of the elements in the combined vector of attention weights and input features is calculated, and passed to the sentence LSTM as its input:

$$\hat{x}^{\varsigma}_s = \sum_{i=1}^{M} \alpha_{i,s} \odot x_i \quad (4)$$

To demonstrate the difference between simple methods to represent collection of image regions and attention, we also use max-pooling in our experiments to obtain inputs for the sentence LSTM:

$$\hat{x}^{\varsigma}_s = max^M_{i=1}(x) \quad (5)$$

### 3.3 Word LSTM

Our word LSTM is a single-layer LSTM similar to Krause et al. (2017). We create $S$ copies of word LSTM (one for each sentence), and use concatenation of the corresponding hidden state of sentence LSTM with the learned embeddings of the words in the target sentence $y_s$ as its input:

$$x^{\omega}_s = [h^{\varsigma}_s \oplus Ey_s] \quad (6)$$

Our word embedding matrix $E \in \mathbb{R}^{K \times H}$ is learned from scratch, $K$ is the vocabulary size. Each copy of word LSTM is unrolled for $M$ timestamps and at each step its hidden step is used to predict a probability distribution over the words in the vocabulary. The final set of sentences is concatenated together to form a paragraph.

### 3.4 Learning Objective

Our hierarchical networks are trained end-to-end with each corresponding image-paragraph pair $(x, y)$ from training data. Our training loss is a simple cross-entropy loss on the word level:

$$loss_{\omega}(x, y) = -\sum_{i=1}^{S}\sum_{j=1}^{M_i} log(p_{j,s}) \qquad (7)$$

where $p_{j,s}$ is the softmax probability of the $j^{th}$ word in the $i^{th}$ sentence given all previously generated words for the current sentence $y_{1:j-1,i}$. For our first sentence, hidden states for both LSTMs are initialized with zeros. For every next sentence, both LSTMs use last hidden states generated for the previous sentence from the corresponding layers. During training, we use teacher forcing and feed ground-truth words as target words at each timestamp. We use Adam (Kingma and Ba, 2014) as our optimizer and choose the best model based on the validation loss (early stopping).

Note that different from previous work, we do not implement sentence-level loss for the end of paragraph prediction. Instead, we generate the same number of sentences for each image, as we find in its ground-truth paragraph. We leave the task of predicting the number of sentences to generate in a paragraph for future work.

## 4 Experiments

We use beam search with beam width $B = 2$ to generate each sentence. Also, similar to Klein et al. (2017), we ensure that each generated sentence consists of at least $C$ words. Our experiments has shown that controlling for minimum length provides us with better overall automatic evaluation metric scores.

## 5 Evaluation

Our primary research question is whether supplying image paragraph models with additional background information affects the sentence diversity of generated paragraphs without hurting their accuracy. Therefore, we evaluate our models in terms of two important dimensions: *accuracy* and *diversity*. Accurate models are required to produce paragraphs grounded in an image and do not mention any irrelevant/incorrect information. Also, these paragraphs must be diverse enough to describe salient image objects and avoid nonsense repetitions within and between sentences.

### 5.1 Accuracy

Typically, a variety of n-gram based automatic metrics is used to measure the correctness/accuracy of image captions. Here, we evaluate our models across six different metrics: CIDEr (Vedantam et al., 2014), METEOR (Denkowski and Lavie, 2014), and BLEU (Papineni et al., 2002). The results are presented in Table 1. Our worst performing model is provided with visual input only, indicating that such input alone is not enough to produce better paragraphs as denoted by automatic evaluation scores. We observe a minor boost across BLEU scores when our model is given background information or includes attention on the sentence level. However, since both CIDEr and METEOR have shown to correlate better with human judgments than BLEU metric, we emphasize that both of these scores benefit from both language information and attention. Our best performing model uses attention on its sentence topics without incorporating semantic knowledge about objects. However, note that models that incorporate *LNG* as part of their input show better performance overall than the model, which is given visual information only.

### 5.2 Diversity

In order for paragraphs to be as close as possible to the human performance when describing images, they have to mention both various and diverse objects from sentence to sentence. Diversity in image descriptions has shown to be an important problem in captioning tasks and corresponding benchmark datasets (Devlin et al., 2015; Lindh et al., 2018). However, a standard evaluation metric to measure diversity in image descriptions has not yet been introduced, indicating an open question in evaluation research.

Here, we start by providing some general statistics about generated paragraphs. As Table 2 demonstrates, the model that incorporates language information has the largest vocabulary size compared to all other models. It also generates more unique nouns in paragraphs, which can be used as a rough (and quite idealistic) indicator of the number of image objects which are mentioned. Though our model with language as its input produces many repetitive nouns per paragraph compared to the ground-truth, it is still slightly better than other configurations. Finally, it has the largest overlap with human-produced paragraphs in terms of word types.

| Model | Input Feature | | Attention | CIDEr | METEOR | BLEU-1 | BLEU-2 | BLEU-3 | BLEU-4 |
|---|---|---|---|---|---|---|---|---|---|
| | Visual | Language | | | | | | | |
| IMG | ✓ | | | 17.85 | 14.31 | 38.76 | 21.92 | 12.51 | 7.06 |
| IMG+LNG | ✓ | ✓ | | 18.42 | 14.39 | 39.47 | 22.49 | 12.81 | 7.18 |
| IMG+ATT | ✓ | | ✓ | **20.28** | **14.54** | **39.79** | **22.64** | **12.91** | 7.27 |
| IMG+LNG+ATT | ✓ | ✓ | ✓ | 18.51 | 14.50 | 39.23 | 22.45 | 12.88 | **7.30** |

Table 1: Scores for automatic evaluation metrics. Each model is named based on the input it takes. Use of language features and attention is additionally specified.

| Model | Voc Size | # of unique nouns | % of rep. nouns per par. | unique word overlap with GT |
|---|---|---|---|---|
| IMG | 400 | 295 | 49 | 6.8% |
| IMG+LNG | **425** | **320** | **47** | **7.2%** |
| IMG+ATT | 413 | 313 | 48 | 7.0% |
| IMG+LNG+ATT | 403 | 298 | 51 | 6.9% |
| GT | **5835** | **3865** | **21** | **100%** |

Table 2: General statistics for generated image paragraphs.

In order to evaluate the diversity of paragraphs automatically, we use Self-BLEU (Zhu et al., 2018). This metric has been specifically proposed to assess the similarity between two sentences, and it can be used to measure how much one sentence resembles another. Higher self-Bleu indicates less diversity, e.g., more n-gram matching between sentences. We calculate Self-BLEU as follows: we split each generated paragraph into sentences and use one sentence as hypothesis and the others are regarded as reference. Then, BLEU score is calculated for every sentence in every paragraph, and the average BLEU score over the paragraphs is used as the Self-BLEU score of the whole set.

| Model | SB-1 | SB-2 | SB-3 | SB-4 |
|---|---|---|---|---|
| IMG | 78.18 | 66.69 | 57.45 | 49.90 |
| IMG+LNG | 77.93 | 66.26 | 56.81 | 48.87 |
| IMG+ATT | 77.75 | 66.37 | 57.27 | 49.84 |
| IMG+LNG+ATT | **77.24** | **65.49** | **56.33** | **48.91** |
| **GT** | **49.63** | **27.70** | **15.55** | **8.52** |

Table 3: Self-Bleu scores for all models. GT indicates scores for ground-truth paragraphs from the test set.

As table 3 shows, ground-truth paragraphs are unsurprisingly the most diverse ones in terms of repetitions of n-grams between sentences. All our models fall behind and by a large margin. However, a clear systematic trend is present: models with *LNG* as part of their input, demonstrate lower self-Bleu scores, compared to the scores by the models that are conditioned solely on visual information and incorporate attention on the sentence level. This finding indicates that contextual language information might help learn to generate more unique descriptions from sentence to sentence in an image paragraph.

## 6 Conclusion

## References

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2017. Bottom-up and top-down attention for image captioning and visual question answering.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate.

Raffaella Bernardi, Ruket Cakici, Desmond Elliott, Aykut Erdem, Erkut Erdem, Nazli Ikizler-Cinbis, Frank Keller, Adrian Muscat, and Barbara Plank. 2016. Automatic description generation from images: A survey of models, datasets, and evaluation measures.

Moitreya Chatterjee and Alexander G. Schwing. 2018. Diverse and coherent paragraph generation from images. In *ECCV*.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollar, and C. Lawrence Zitnick. 2015. Microsoft coco captions: Data collection and evaluation server.

Robin Cooper. 2008. Type theory with records and unification-based grammar. In *Logics for Linguistic Structures, pages 9 – 34. Mouton de Gruyter*.

Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the EACL 2014 Workshop on Statistical Machine Translation*.

Jacob Devlin, Hao Cheng, Hao Fang, Saurabh Gupta, Li Deng, Xiaodong He, Geoffrey Zweig, and Margaret Mitchell. 2015. Language models for image captioning: The quirks and what works.

Simon Dobnik and John D. Kelleher. 2016. A model for attention-driven judgements in type theory with records.

Albert Gatt and Emiel Krahmer. 2017. Survey of the State of the Art in Natural Language Generation: Core tasks, applications and evaluation.

Simao Herdade, Armin Kappeler, Kofi Boakye, and Joao Soares. 2019. Image captioning: Transforming objects into words.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Micah Hodosh, Peter Young, and Julia Hockenmaier. 2013. Framing image description as a ranking task: Data, models and evaluation metrics. *J. Artif. Int. Res.*, 47(1):853–899.

Justin Johnson, Andrej Karpathy, and Li Fei-Fei. 2016. Densecap: Fully convolutional localization networks for dense captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *International Conference on Learning Representations*.

Ryan Kiros, Ruslan Salakhutdinov, and Rich Zemel. 2014. Multimodal neural language models. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 595–603, Bejing, China. PMLR.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, Vancouver, Canada. Association for Computational Linguistics.

Jonathan Krause, Justin Johnson, Ranjay Krishna, and Li Fei-Fei. 2017. A hierarchical approach for generating descriptive image paragraphs. In *Computer Vision and Patterm Recognition (CVPR)*.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. 2016. Visual genome: Connecting language and vision using crowdsourced dense image annotations.

Nilli Lavie, Aleksandra Hirst, Jan W de Fockert, and Essi Viding. 2004. Load theory of selective attention and cognitive control. *Journal of experimental psychology. General*, 133(3):339—354.

Xiaodan Liang, Zhiting Hu, Hao Zhang, Chuang Gan, and Eric P. Xing. 2017. Recurrent topic-transition gan for visual paragraph generation.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. 2014. Microsoft coco: Common objects in context.

Annika Lindh, Robert J. Ross, Abhijit Mahalunkar, Giancarlo Salton, and John D. Kelleher. 2018. Generating Diverse and Meaningful Captions. In *Artificial Neural Networks and Machine Learning – ICANN 2018*, Lecture Notes in Computer Science, pages 176–187. Springer International Publishing.

Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. 2016a. Knowing When to Look: Adaptive Attention via A Visual Sentinel for Image Captioning.

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2016b. Hierarchical Question-Image Co-Attention for Visual Question Answering.

Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation.

Luke Melas-Kyriazi, Alexander Rush, and George Han. 2019. Training for Diversity in Image Paragraph Captioning.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jarret Ross, and Vaibhava Goel. 2016. Self-critical Sequence Training for Image Captioning.

Karen Simonyan and Andrew Zisserman. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition.

S. Ullman. 1987. Visual routines. In M. A. Fischler and O. Firschein, editors, *Readings in Computer Vision: Issues, Problems, Principles, and Paradigms*, pages 298–328. Kaufmann, Los Altos, CA.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. 2014. Cider: Consensus-based image description evaluation.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2014. Show and tell: A neural image caption generator.

Jing Wang, Yingwei Pan, Ting Yao, Jinhui Tang, and Tao Mei. 2019. Convolutional auto-encoding of sentence topics for image paragraph generation.

Huijuan Xu and Kate Saenko. 2015. Ask, Attend and Answer: Exploring Question-Guided Spatial Attention for Visual Question Answering.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention.

Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. 2016. Image Captioning with Semantic Attention.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.

Alessandra Zarcone, Marten van Schijndel, Jorrig Vogels, and Vera Demberg. 2016. Salience and attention in surprisal-based accounts of language processing.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texygen: A Benchmarking Platform for Text Generation Models.

## A Appendices

Appendices are material that can be read, and include lemmas, formulas, proofs, and tables that are not critical to the reading and understanding of the paper. Appendices should be **uploaded as supplementary material** when submitting the paper for review. Upon acceptance, the appendices come after the references, as shown here.

**LaTeX-specific details:** Use `\appendix` before any appendix section to switch the section numbering over to letters.

## B Supplemental Material

Submissions may include non-readable supplementary material used in the work and described in the paper. Any accompanying software and/or data should include licenses and documentation of research review as appropriate. Supplementary material may report preprocessing decisions, model parameters, and other details necessary for the replication of the experiments reported in the paper. Seemingly small preprocessing decisions can sometimes make a large difference in performance, so it is crucial to record such decisions to precisely characterize state-of-the-art methods.

Nonetheless, supplementary material should be supplementary (rather than central) to the paper. **Submissions that misuse the supplementary material may be rejected without review.** Supplementary material may include explanations or details of proofs or derivations that do not fit into the paper, lists of features or feature templates, sample inputs and outputs for a system, pseudo-code or source code, and data. (Source code and data should be separate uploads, rather than part of the paper).

The paper should not rely on the supplementary material: while the paper may refer to and cite the supplementary material and the supplementary material will be available to the reviewers, they will not be asked to review the supplementary material.