

Scene Graph Generation for Better Image Captioning?

Maximilian Mozes^{1,3} Martin Schmitt² Vladimir Golkov¹
Hinrich Schütze² Daniel Cremers¹

¹Computer Vision Group, Technical University of Munich

²Center for Information and Language Processing (CIS), LMU Munich

³University College London

maximilian.mozes@ucl.ac.uk

Abstract

We investigate the incorporation of visual relationships into the task of supervised image caption generation by proposing a model that leverages detected objects and auto-generated visual relationships to describe images in natural language. To do so, we first generate a scene graph from raw image pixels by identifying individual objects and visual relationships between them. This scene graph then serves as input to our graph-to-text model, which generates the final caption. In contrast to previous approaches, our model thus explicitly models the detection of objects and visual relationships in the image. For our experiments we construct a new dataset from the intersection of Visual Genome and MS COCO, consisting of images with both a corresponding gold scene graph and human-authored caption. Our results show that our methods outperform existing state-of-the-art end-to-end models that generate image descriptions directly from raw input pixels when compared in terms of the BLEU and METEOR evaluation metrics.

1 Introduction

Recent works dealing with the generation of text from data structures such as images (e.g., Karpathy and Li, 2015; Vinyals et al., 2015), videos (e.g., Venugopalan et al., 2015) or audio (e.g., Graves et al., 2013) have shown that supervised learning algorithms are capable of aligning semantic concepts across different modalities. In this work, we focus on the task of automatic image captioning, a widely-studied task at the intersection of vision and language research. Most approaches to image captioning operate by conditioning a decoder model on an abstracted representation of the input image instead of explicitly taking detected objects and visual relationships into account (e.g.,

Karpathy and Li, 2015; Xu et al., 2015). However, natural language descriptions in general and captions in particular are dominated by discrete objects standing in discrete relations. By forcing the generation process to go through a scene graph consisting of objects and relations, we impose an appropriate structural bias that is lacking in direct pixel-to-caption generation. We therefore approach the task of supervised image caption generation by developing an architecture that makes explicit use of detected visual objects and their semantic relationships in a given input image to generate an image description in natural language. More specifically, our method consists of a two-step approach that first extracts a scene graph (i.e., objects and their visual relationships) from an input image and then utilizes this representation to generate an image description in natural language. In doing so, we incorporate an existing method for supervised scene graph generation, i.e., MOTIFNET (Zellers et al., 2018), to extract visual semantic concepts from images and represent them in form of scene graphs.

Scene graphs have been utilized in a variety of tasks such as image retrieval (e.g., Johnson et al., 2015) and image generation (Johnson et al., 2018) and are of particular interest for tasks dealing with the alignment of visual and textual concepts, since the representations utilize words to describe phenomena that are present in visual scenarios. While numerous approaches for image-to-graph generation and visual relationship detection have been proposed in recent years (e.g., Lu et al., 2016; Newell and Deng, 2017; Li et al., 2017; Yang et al., 2018a; Zellers et al., 2018; Zhang et al., 2019), little attention has thus far been paid to the problem of graph-to-text generation. We hence propose a variety of methods utilizing recurrent neural network mechanisms operating on scene graphs for the generation of natural language and show that the presence of

Technical report. Work done and written in 2019.

visual objects and their relationships is beneficial for the automatic description of images.

Our work thus presents the following main contributions:

1. We propose a two-step supervised learning approach that generates scene graphs from raw input pixels and utilizes these graph representations to generate image descriptions in natural language.
2. We show that such a simple two-step approach outperforms conventional CNN-LSTM image captioning architectures.

2 Related work

The problem of end-to-end image caption generation has been studied widely in the context of deep learning in recent years. Pioneering approaches to this problem utilize a combination of convolutional and recurrent neural networks processing the visual and textual data representations, respectively. Multiple encoder-decoder approaches have been proposed that employ a CNN transforming a raw input image to a dense vector representation which is then used to condition a neural language model generating a descriptive sequence in natural language (e.g., [Chen and Zitnick, 2015](#); [Donahue et al., 2015](#); [Vinyals et al., 2015](#); [Karpathy and Li, 2015](#); [Wang et al., 2016](#)).

Building upon this idea, [Xu et al. \(2015\)](#) propose the first approach to incorporate an additional attention mechanism into the model’s decoder, enabling it to refer back to the abstracted image representation at each time step during the generation of an image caption. Subsequent approaches extend the incorporation of attention mechanisms for image captioning (e.g., [Yang et al., 2016](#); [Lu et al., 2017](#); [Khademi and Schulte, 2018](#)). For instance, [Lu et al. \(2017\)](#) extend the idea of incorporating visual attention to the image caption generation task by introducing an adaptive attention mechanism allowing the model to decide to what extent it should rely on the visual and linguistic features when generating an image caption.

2.1 Generating image captions from visual relationships

Although comparatively little attention has been paid to the generation of image captions via visual relationships, there exists a variety of works

employing these characteristics to generate image captions.

[Yao et al. \(2018\)](#) propose an architecture that utilizes region-based visual relationships to generate an image caption for a given image. Specifically, their method uses the *Faster R-CNN* ([Ren et al., 2015](#)) object detector to identify a set of objects present in an input image. Afterwards, a classification method is applied on pairs of detected objects to identify their most probable semantic relationship. The resulting graph representation is then forwarded to two Graph Convolutional Neural Networks (CGN) that generate relation-aware region features for all the detected regions based on their predicted visual relationships. Finally, a two-layer LSTM is conditioned on the region-level features generated by the CGN module, and generates the image caption based on this representation. [Yao et al. \(2018\)](#) additionally install an attention mechanism in the LSTM decoder that operates over the region features at each time step when generating the output predictions.

Moreover, [Yang et al. \(2018b\)](#) propose the incorporation of scene graphs into image captioning by utilizing them to model language inductive bias during the task, and [Kim et al. \(2019\)](#) propose a dense captioning mechanism that produces multiple individual captions per image. Their approach initially uses a bounding box object detector that identifies object regions present in an input image. Afterwards, a recurrent neural network is trained to generate a caption for each relational pair of identified objects.

Two recent works published by [Li and Jiang \(2019\)](#) and [Hou et al. \(2019\)](#) present approaches that are similar to our work. [Li and Jiang \(2019\)](#) combine scene graphs for image captioning in conjunction with a hierarchical attention network. Their approach first uses a Region Proposal Network ([Girshick, 2015](#)) to compute object proposals for an input image. These proposals are then used to generate both a visual feature representation and semantic relationship features, which are forwarded to an LSTM decoder with a hierarchical attention module generating the image caption. [Hou et al. \(2019\)](#) provide a different method for incorporating scene graphs into the image captioning pipeline by utilizing scene graphs sourced from the Visual Genome dataset as external prior knowledge graphs.

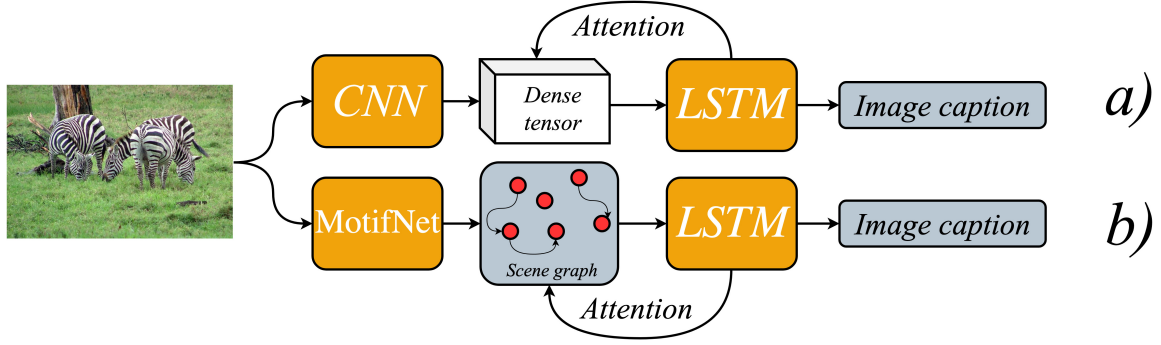


Figure 1: Illustration and comparison of the **Pixel2Caption+att** (described in *a*) and our **G-LSTM+att** (shown in *b*) methods. **Pixel2Caption+att** processes the input image by computing a dense numerical representation using a CNN which is then used to condition an LSTM that generates the image caption. **G-LSTM+att**, in contrast, utilizes MOTIFNET to craft a scene graph from the input image, which is then processed by an LSTM to generate the final caption.

3 Method

The proposed approach for generating image captions via visual relationships is divided into two parts. Our model tackles the *image-to-text* generation task by first generating an intermediate scene graph representation of the input image and then decodes an image caption from this representation. Hence, our method conducts *image-to-graph-to-text* generation by approaching the subtasks of *image-to-graph* and *graph-to-text* in an isolated fashion. To achieve this, we use two neural network architectures that focus on each task independently, and stack both architectures together once they have been trained.

3.1 Scene graph generation

We initially aim to solve the problem of *image-to-graph* generation, i.e., generating a scene graph consisting of objects and visual relationships present in a given input image. Formally, our scene graph generator crafts a scene graph $G_I = (V, E)$ for an input image I that consists of a set of nodes V and corresponding directed edges $E \subseteq V \times V$. Each node v is associated with a label $\kappa(v)$, representing an object in an image (e.g., *car*, *person*, *building*). Likewise, each edge $e = (v_i, v_j) \in E$ is assigned a label $\kappa(e)$ denoting a relationship between the two objects $\kappa(v_i)$ and $\kappa(v_j)$ (e.g., *above*, *on*).

In order to generate a graph G_I from raw input pixels I , we make use of an existing scene graph generation model called MOTIFNET (Zellers et al., 2018). This method represents a scene graph as a triplet $G_M = (B, O, R)$, with

$B = \{b_1, \dots, b_n\}, b_i \in \mathbb{R}^4$ a set of bounding boxes, $O = \{o_1, \dots, o_n\}$ a set of objects where each o_i corresponds to a bounding box b_i , and $R = \{r_1, \dots, r_m\}$ a set of relationships where each relationship r_k is a triplet $r_k = ((b_i, o_i), (b_j, o_j), x_{i \rightarrow j})$. Here, $(b_i, o_i), (b_j, o_j) \in B \times O$ represent the start and end node of the relationship and $x_{i \rightarrow j} \in \mathcal{R}$ denotes the relationship between both nodes from all possible relationships \mathcal{R} . Based on this scene graph representation, MOTIFNET computes the probability $P(G_M | I)$ of observing graph G_M given image I by decomposing it into three parts:

$$P(G_M | I) = P(B | I) \cdot P(O | B, I) \cdot P(R | B, O, I)$$

Zellers et al. (2018) model $P(B | I)$ with the *Faster R-CNN* (Ren et al., 2015) bounding box detection model. They then employ two LSTM networks (Hochreiter and Schmidhuber, 1997) to estimate the bounding box labels $P(O | B, I)$. Subsequently, the authors employ a bidirectional LSTM to compute the relationships between objects identified by the object detector as denoted by $P(R | B, O, I)$. To do so, all possible pairs of detected objects are taken into account and the LSTM computes a probability distribution over all potential relationships in \mathcal{R} for each pair of objects.

3.2 Graph-to-text generation

Once we have generated a graph representation $G_I = (V, E)$ for an input image I , we utilize an LSTM decoder with an additional attention mechanism over the graph to generate an output sequence in natural language. Our architecture receives a

set of graph nodes V and maps each node $v \in V$ to an embedding representation $\mathbf{v} \in \mathbb{R}^D$ corresponding to its node label $\kappa(v)$. Hence, in order to represent visual relationships in this setup, we first transform our graph G_I to a new representation $G'_I = (V', E')$ that differs from G_I in that each edge label is now assigned an individual node in the graph, i.e., for each $e = (v_i, v_j) \in E$ we create a new node v' such that $\kappa(e) = \kappa(v')$ and add edges $e'_i = (v_i, v')$, $e'_j = (v', v_j)$ to E' with $\kappa(e'_i) = \kappa(e'_j) = \text{None}$.

Our method then applies an LSTM to the matrix $\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_n] \in \mathbb{R}^{D \times n}$ of each node's embedding representation. To do so, we follow Xu et al. (2015) and first initialize the LSTM's hidden and cell states as

$$\mathbf{h}_0 = \psi_h \left(\frac{1}{n} \sum_{i=1}^n \mathbf{v}_i \right) \quad \mathbf{c}_0 = \psi_c \left(\frac{1}{n} \sum_{i=1}^n \mathbf{v}_i \right),$$

where ψ_h and ψ_c are two independent multilayer perceptrons. Based on this initial conditioning, we then decode the image caption by sampling from

$$p(\mathbf{y}_t | \mathbf{y}_{t-1}, \mathbf{V}) \propto \exp(\mathbf{P}_o \tanh(\mathbf{E}_W \mathbf{y}_{t-1} + \mathbf{P}_h \mathbf{h}_t))$$

at each time step t , thereby also following Xu et al. (2015). Here, $\mathbf{E}_W \in \mathbb{R}^{D \times V}$ represents our word embedding matrix (V is the vocabulary size), $\mathbf{y}_{t-1} \in \{0, 1\}^V$ is a one-hot representation of the model's prediction at time step $t-1$ (or a special start token at $t=0$), \mathbf{h}_t is the LSTM's hidden state at time step t and $\mathbf{P}_o, \mathbf{P}_h$ are trainable parameter matrices. In the remainder of this work, we refer to the combination of our graph encoder and this type of decoder as **G-LSTM**.

Our second model variant incorporates an additional attention mechanism operating over the latent graph representation \mathbf{V} at each time step t of the LSTM. We adapt Xu et al. (2015)'s approach for image captioning with visual attention and replace the latent image representation with our graph nodes, thus enabling our model to refer back to the graph representation and identify the most salient nodes at each time step during the generation of the output sequence. We call this extended approach **G-LSTM_A**.

3.3 Encoding visual relationships

The aforementioned **G-LSTM+att** does not explicitly incorporate the visual relationships between objects as represented in the scene graph, but instead only processes all object and relationship

nodes to generate an image caption. We thus experiment with the incorporation of an additional graph encoder that maps the initial graph representation $\mathbf{V} = \{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ to an output representation $\mathbf{V}' = \{\mathbf{v}'_1, \dots, \mathbf{v}'_n\}$. The task of this graph encoder is to encode relational information for each graph node into its corresponding graph embedding to provide the decoder with semantic dependencies between individual nodes in the graph. Additionally, the encoder has the ability to process indirect connections between entities in order to contextualize global relationships between entities that are indirectly connected through multiple edges. *Graph Attention Networks* (GAT; Veličković et al. (2018)) represent a gradient-based approach that transforms an input graph by individually attending over each node's neighborhood to encode relational information into the resulting node representations. For a given input graph $G = (V, E)$, we then define a graph representation $\mathbf{G} = (\mathbf{V}, \mathbf{E})$, where

$$\mathbf{E} = \{\{\mathbf{v}_i, \mathbf{v}_j\} \mid (v_i, v_j) \in E \vee (v_j, v_i) \in E\}$$

represents the set of undirected edges in G corresponding to E .

GAT layers transform the node representations by computing attention over their neighborhoods. Formally, let \mathcal{N}_i denote the neighborhood of a node embedding $\mathbf{v}_i \in \mathbf{V}$. A GAT layer $\phi: \mathbb{R}^D \rightarrow \mathbb{R}^{D'}$ then transforms each \mathbf{v}_i to \mathbf{v}'_i by computing

$$\mathbf{v}'_i = \phi(\mathbf{v}_i) = \sigma \left(\sum_{\mathbf{v}_j \in \mathcal{N}_i} \alpha_{ij} \mathbf{W} \mathbf{v}_j \right).$$

Here, σ represents the sigmoid function and α_{ij} is an attention coefficient with respect to the nodes \mathbf{v}_i and \mathbf{v}_j . We follow Veličković et al. (2018) and set

$$\alpha_{ij} = \frac{\exp(\text{LR}(\mathbf{a}^T [\mathbf{W} \mathbf{v}_i \parallel \mathbf{W} \mathbf{v}_j]))}{\sum_{\mathbf{v}_k \in \mathcal{N}_i} \exp(\text{LR}(\mathbf{a}^T [\mathbf{W} \mathbf{v}_i \parallel \mathbf{W} \mathbf{v}_k]))}$$

where $\mathbf{W} \in \mathbb{R}^{D' \times D}$, $\mathbf{a} \in \mathbb{R}^{2D'}$ are trainable weight matrices, \parallel represents vector concatenation and $\text{LR}(\cdot)$ denotes the LeakyReLU activation function. In our experiments, we define $\mathcal{N}_i := \{\mathbf{v} \in \mathbf{V} \mid \{\mathbf{v}_i, \mathbf{v}\} \in \mathbf{E}\} \cup \{\mathbf{v}_i\}$ to ensure a direct connection between an input node \mathbf{v}_i and its transformation $\phi(\mathbf{v}_i)$ in each GAT layer.

Our final graph encoder then consists of multiple GAT layers that are executed sequentially to transform the node embedding representations with respect to their relationships in the graph. Once

our encoder has processed the initial graph embedding representation, we then feed our **G-LSTM** models with this representation and train the entire model in an end-to-end fashion. We denote both model variants with **G-LSTM+enc** and **G-LSTM+enc+att**.

3.4 Conventional image captioning baselines

To provide a comparison between our approach and the conventional CNN-LSTM image captioning, we adapt Xu et al. (2015)’s method. We pre-process each input image using the VGG19 network (Simonyan and Zisserman, 2015) pre-trained on ImageNet, and condition our LSTM language model on the $14 \times 14 \times 512$ feature representation emitted by the fifth layer of VGG19 before applying max-pooling. Analogously to the graph-to-text models, we furthermore experiment with an additional visual attention mechanism operating over the input image (see Xu et al. (2015)). We denote both approaches with **Pixel2Caption** and **Pixel2Caption+att**.

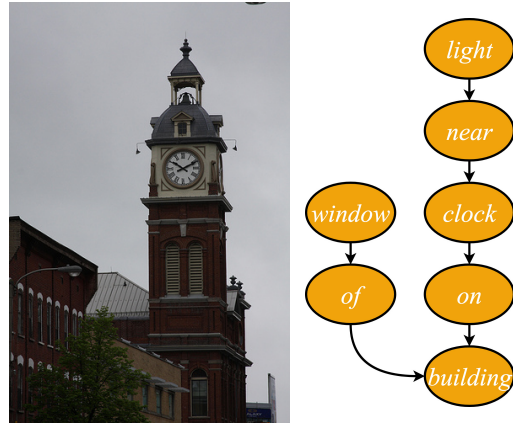
Figure 1 provides an overview and comparison of both the **G-LSTM+att** and **Pixel2Caption+att** models. Both follow a similar technique of firstly encoding an input image by transforming it to a latent representation. This latent representation is then used to decode the corresponding image caption using an attention mechanism. However, a major difference between **Pixel2Caption+att** and **G-LSTM+att** is that the latent representation of the latter (i.e., the scene graph) allows humans to explicitly observe which visual and contextual information have been extracted from the image. This property is not given for the **Pixel2Caption+att** approach, since the latent representation emitted by the CNN is highly abstracted and hence less interpretable.

4 Experiments

We conduct a series of experiments on a subset of the Visual Genome (Krishna et al., 2017) and MS COCO (Lin et al., 2014) datasets consisting of images accompanied by bounding boxes, scene graphs and individual image captions.

We use the BLEU (Papineni et al., 2002) and METEOR (Denkowski and Lavie, 2014) evaluation metrics to measure the performance of our proposed approaches and to be able to compare them to existing methods for image caption generation. Both metrics have been used in a variety of studies

related to image caption generation (e.g., Xu et al., 2015; Vinyals et al., 2015; Lu et al., 2017).



1. *a clock tower is in the gray sky.*
2. *clock tower ascending into overcast sky from buildings below.*
3. *a church tower that has a clock for the public.*
4. *brick building with clock tower in urban setting.*
5. *a tall clock tower near a building.*

Figure 2: An example triplet of our generated dataset. The image is present in both the Visual Genome and MS COCO datasets. The scene graph is taken from a modified set of scene graphs from Visual Genome (Xu et al., 2017) and the five image captions are taken from MS COCO.

4.1 Datasets

Our dataset consists of a subset of all 51,498 images at the intersection of the Visual Genome and MS COCO datasets. First, we split the 51,498 images into a test set of 5,000 images, a validation set of 1,000 images and a training set of 45,498 samples. Operating on the intersection of VG and MS COCO allows us to craft triplet samples consisting of an image, a corresponding scene graph and a list of captions describing the image. In order to be as consistent as possible with the existing literature on scene graph generation, we then match all dataset samples with a modified Visual Genome dataset as explained in Xu et al. (2017), considering only the 150 most common object categories and 50 most common relationships. As each image in the training set is on average accompanied by 5.002 captions sourced from MS COCO, the graph-to-text generation module can be trained with a total amount of 221,792 (*scene graph, caption*) pairs.

Model	B-1	B-2	B-3	B-4	METEOR
Pixel2Caption	65.58	43.93	29.58	20.40	22.90
Pixel2Caption+att	66.09	44.04	29.32	19.96	22.65
G-LSTM	67.29	45.47	30.48	20.85	23.79
G-LSTM+att	67.71	45.95	30.63	20.70	23.87
G-LSTM+enc	66.30	43.56	28.33	18.82	22.75
G-LSTM+enc+att	65.63	43.69	28.81	19.48	23.33

Table 1: BLEU and METEOR scores for all trained models when evaluated on the test set. The **Pixel2Caption** and **Pixel2Caption+att** models were evaluated on the VGG image representations, and all **G-LSTM** models were evaluated on the scene graphs generated by MOTIFNET (trained and tuned on our training and validation sets, respectively). Bold values indicate best performances for each evaluation criterion across all models.

An example for a single element from our generated dataset (image, scene graph and captions) can be found in Figure 2.

During validation and testing, we evaluated our model’s predictions using all available captions for a given image.

4.2 Implementation details and training

We trained the individual submodules responsible for the image-to-graph and graph-to-text generation independently on the aforementioned datasets.

The scene graph generator was trained by strictly following Zellers et al. (2018)’s approach to train their proposed model.¹ This approach consists of three phases. First, a Faster R-CNN object detector with a VGG backbone is pre-trained in isolation to learn the extraction of objects and corresponding bounding boxes from images. We adhered to the architecture and parameter setup as explained in their work, and trained the detector for 50 epochs. After training the object detector, we trained the MOTIFNET module for 26 epochs without modifying the authors’ implementation setup (this includes the adaptation to scene graph detection as explained in Zellers et al. (2018), Section 5.2). For the graph-to-text models, we tokenized all sequences used during training using the NLTK `tokenize` package (Loper and Bird, 2002). We did not exclude infrequent vocabulary tokens during our analysis. All reported models were trained using the *Adam* optimizer (Kingma and Ba, 2014) with a learning rate of $1 \cdot 10^{-4}$. In terms of model regularization, we used *dropout* (Srivastava et al., 2014) in both

¹We followed the authors’ instructions on <https://github.com/rowanz/neural-motifs>.

the encoder and the decoder during training. In the encoder, we added a dropout mechanism with a rate of 0.25 at each GAT layer directly before computing the weighted sum of the transformer graph inputs. In the decoder, we adhered to the use of dropout as realized by Xu et al. (2015) and used a dropout rate of 0.5. Moreover, we use *batch normalization* (Ioffe and Szegedy, 2015) in the LSTM decoder by normalizing the encoder outputs before transforming them to the LSTM’s initial hidden and cell states. Our graph encoder consists of two consecutive GAT layers that are operating on a dimension of $D = 512$. We set the dimension of the trainable graph and word embeddings to the same size and utilize a single-layer LSTM with 1024 hidden units as decoder.

We trained our two conventional image captioning baselines **Pixel2Caption** and **Pixel2Caption+att** with the same hyperparameter settings.

4.3 Tuning MOTIFNET on the validation set

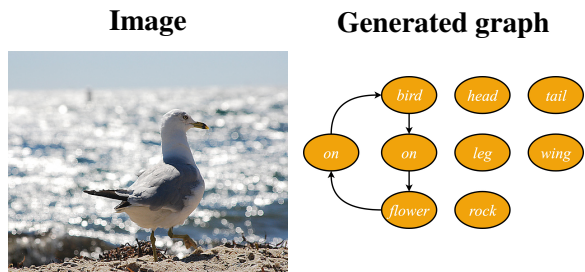
For a given input image, the trained MOTIFNET generates both a list of detected bounding boxes along with their predicted labels as well as a list of relationship predictions between the identified objects. In detail, it outputs a probability distribution over all possible 50 relationship predicates for each pair of predicted objects. However, the *Faster R-CNN* object detector predicts certain bounding box labels with low confidence values which might result in scene graph representations with high model uncertainty. To account for this problem, and to limit the size (i.e., number of nodes) of the generated scene graphs, we experimented with various confidence threshold values representing lower bounds for the confidence values of the object detector to be considered a valid object of an image. Specifically, we considered the confidence thresholds 0.2, 0.4, 0.6, and 0.8 for our trained models. For each of the four **G-LSTM** model variants, we thus evaluated to what extent these different confidence thresholds affected the overall model performances (in terms of METEOR) by experimenting how the model variants perform on the validation set with each of the parameter values. Our results suggest that the **G-LSTM**, **G-LSTM+enc+att** and **G-LSTM+enc** models exhibit their best performance with a confidence threshold of 0.4, while the **G-LSTM+att** variant performs best with a confidence threshold of 0.2.

Model	B-1	B-2	B-3	B-4	METEOR
G-LSTM	69.09	47.85	32.76	22.77	24.80
G-LSTM+att	69.48	48.31	33.16	22.91	24.81
G-LSTM+enc	67.65	45.74	30.51	20.77	23.79
G-LSTM+enc+att	68.06	46.87	31.79	21.90	24.56

Table 2: BLEU and METEOR scores for all graph-based models when evaluated on the ground-truth scene graphs in the test set. Bold values indicate best performances for each evaluation criterion across all models.

Once we have identified all valid predicted objects present in the image, we selected the graph’s relationships by considering all relationships between valid objects suggested by MOTIFNET and assigned the predicate with highest probability as the relationship label.

Finally, we removed all duplicate nodes and identical relationships from the crafted scene graph. If a generated scene graph exceeds the maximum size (i.e., number of nodes) of the graphs used during training, we limit the graph’s size to this maximum size by removing the object nodes exhibiting the lowest prediction confidences. Moreover, if a scene graph consists of less than two predicted object nodes, we ignore the sample during testing.



G-LSTM+att: *a bird is perched on a rock in the water.*

Pixel2Caption+att: *a bird is flying in the air on a beach.*

Figure 3: Comparison of generated image captions from both the **G-LSTM+att** and the **Pixel2Caption+att** models. The graph shown on the right-hand side has been generated from the input image (left-hand side) using the trained MOTIFNET. The sequences below the image and graph represent the predicted captions for both systems.

4.4 Results

Quantitative results of all our models can be found in Table 1. The results of all graph-based models are based on the scene graphs generated by

MOTIFNET, which we trained before on our new dataset. The results in Table 1 show that both the **G-LSTM+att** and the **G-LSTM** outperform both the **Pixel2Caption** and **Pixel2Caption+att** in every metric, indicating that our proposed models represent a suitable alternative to the conventional image captioning approaches. Figure 3 shows qualitative results of the **Pixel2Caption+att** and **G-LSTM+att** approaches in comparison, showing that our model is able to produce accurate captions even in the presence of imperfect auto-generated scene graphs. Furthermore, it is interesting to observe that the additional graph encoder operating over the input scene graph leads to performance decreases of our **G-LSTM** model. In addition to that, for both the conventional and the captioning model based on scene graphs, the attention mechanism operating on the decoding LSTM only slightly improves the overall model performance across our evaluation metrics.

To further assess the performance of our models when operating on generated scene graphs, we provide metrics for all model variants when evaluated on the ground-truth gold scene graphs as provided in the Visual Genome dataset in Table 2. Our models exhibit even higher performances when evaluated on the gold scene graphs, indicating that our method has the potential to benefit from future progress in the field of scene graph generation.

4.5 Limitations

The presented method imposes a number of limitations that we would like to address in the following paragraph. First, our image-to-graph-to-text model utilizes a scene graph generation model that is restricted to predicting only 150 different object labels and 50 different edge labels. This represents a notable limitation to the model since it is explicitly trained to predict diverse English sentences from only a small subset of semantic concepts. Nevertheless, the fact that our proposed methods outperform conventional image captioning approaches (which do not have this additional constraint) suggests that the model still learns to predict semantic concepts outside of the 200 given ones in context, and achieves to reasonably generate other concepts that are likely to occur in the context of certain objects and relationships as represented by the scene graph.

Moreover, it is worth mentioning that our proposed approach arguably requires a larger amount

of computational resources to be trained properly when compared to conventional image captioning methods. In addition to that, the current study does not investigate the potential of our proposed architecture when trained in an end-to-end fashion, i.e., by developing a single pipeline that processes an input image, generates a scene graph representation and then uses this representation to create a corresponding image caption. At this point we would like to encourage other researchers focusing on image captioning to further explore the potential of explicitly incorporating visual objects and relationships with respect to this problem.

5 Conclusion

In this work, we proposed a supervised learning approach to generate image captions by explicitly leveraging detected objects and visual relationships. Our suggested model consists of a simple two-step procedure that first generates a scene graph representation from a given image and then uses this representation to generate an image description in natural language. Empirical results on a newly-generated dataset consisting of samples from the intersection of Visual Genome and MS COCO demonstrate the superiority of our model when compared to conventional image captioning approaches, indicating that our method provides a fruitful ground to further advance the task of image captioning.

Acknowledgements

We gratefully acknowledge a Ph.D. scholarship awarded to the second author by the German Academic Scholarship Foundation (Studienstiftung des deutschen Volkes). This work was supported by the BMBF as part of the project MLWin (Grant No. 01IS18050) as well as the Munich Center for Machine Learning (Grant No. 01IS18036B).

References

Xinlei Chen and C. Lawrence Zitnick. 2015. [Mind's eye: A recurrent visual representation for image caption generation](#). In *Proceedings of the 28th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2422–2431.

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*.

Jeff Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Trevor Darrell, and Kate Saenko. 2015. [Long-term recurrent convolutional networks for visual recognition and description](#). In *Proceedings of the 28th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2625–2634.

Ross Girshick. 2015. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448.

Alex Graves, Abdel Rahman Mohamed, and Geoffrey E. Hinton. 2013. Speech recognition with deep recurrent neural networks. In *The 38th International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 6645–6649.

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

Jingyi Hou, Xinxiao Wu, Yayun Qi, Wentian Zhao, Jiebo Luo, and Yunde Jia. 2019. Relational reasoning using prior knowledge for visual captioning. *arXiv preprint arXiv:1906.01290*.

Sergey Ioffe and Christian Szegedy. 2015. [Batch normalization: Accelerating deep network training by reducing internal covariate shift](#). In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 448–456.

J. Johnson, R. Krishna, M. Stark, L. Li, D. A. Shamma, M. S. Bernstein, and L. Fei-Fei. 2015. [Image retrieval using scene graphs](#). In *Proceedings of the 28th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 28, pages 3668–3678.

Justin Johnson, Agrim Gupta, and Li Fei-Fei. 2018. Image generation from scene graphs. In *Proceedings of the 31st IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1219–1228.

Andrej Karpathy and Fei-Fei Li. 2015. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, pages 3128–3137. IEEE Computer Society.

Mahmoud Khademi and Oliver Schulte. 2018. Image caption generation with hierarchical contextual visual spatial attention. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Dong-Jin Kim, Jinsoo Choi, Tae-Hyun Oh, and In So Kweon. 2019. Dense relational captioning: Triple-stream networks for relationship-based captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6271–6280.

- Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*.
- Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73.
- Xiangyang Li and Shuqiang Jiang. 2019. Know more say less: Image captioning based on scene graphs. *IEEE Transactions on Multimedia*.
- Yikang Li, Wanli Ouyang, Bolei Zhou, Kun Wang, and Xiaogang Wang. 2017. Scene graph generation from objects, phrases and region captions. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1270–1279.
- Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer Vision – ECCV 2014*, pages 740–755, Cham. Springer International Publishing.
- Edward Loper and Steven Bird. 2002. Nltk: The natural language toolkit. In *In Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*. Philadelphia: Association for Computational Linguistics.
- Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. 2016. Visual relationship detection with language priors. In *Proceedings of the European Conference on Computer Vision (ECCV)*.
- Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. 2017. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3242–3250.
- Alejandro Newell and Jia Deng. 2017. **Pixels to graphs by associative embedding**. In *Advances in Neural Information Processing Systems 30*, pages 2171–2180.
- 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.
- Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. **Faster r-cnn: Towards real-time object detection with region proposal networks**. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 91–99. Curran Associates, Inc.
- K. Simonyan and A. Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In *Proceedings of the International Conference on Learning Representations (ICLR)*.
- Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. **Dropout: a simple way to prevent neural networks from overfitting**. *Journal of Machine Learning Research*, 15(1):1929–1958.
- Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. In *Proceedings of the 2018 International Conference on Learning Representations (ICLR)*.
- Subhashini Venugopalan, Marcus Rohrbach, Jeff Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. 2015. Sequence to sequence – video to text. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.
- Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator. In *Proceedings of the 2015 Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Cheng Wang, Haojin Yang, Christian Bartz, and Christoph Meinel. 2016. Image captioning with deep bidirectional lstms. In *ACM Multimedia*.
- Danfei Xu, Yuke Zhu, Christopher Bongsoo Choy, and Li Fei-Fei. 2017. Scene graph generation by iterative message passing. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3097–3106.
- 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. In *Proceedings of the 2015 International Conference on Machine Learning (ICML)*, pages 2048–2057.
- Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. 2018a. Graph r-cnn for scene graph generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*.
- Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. 2018b. Auto-encoding scene graphs for image captioning. *arXiv preprint arXiv:1812.02378*.
- Zhilin Yang, Ye Yuan, Yuexin Wu, William W Cohen, and Ruslan R Salakhutdinov. 2016. **Review networks for caption generation**. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 2361–2369. Curran Associates, Inc.
- Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. 2018. Exploring visual relationship for image captioning. In *ECCV*.

Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. 2018. Neural motifs: Scene graph parsing with global context. In *Conference on Computer Vision and Pattern Recognition*.

Ji Zhang, Yannis Kalantidis, Marcus Rohrbach, Manohar Paluri, Ahmed Elgammal, and Mohamed Elhoseiny. 2019. Large-scale visual relationship understanding. In *AAAI*.