

# Guiding Visual Question Generation

Nihir Vedd, Zixu Wang, Marek Rei, Yishu Miao, and Lucia Specia

Imperial College London

{n.vedd19, zixu.wang, marek.rei, y.miao20, l.specia}@imperial.ac.uk

<https://github.com/nihirv/guiding-vqg>

## Abstract

In traditional Visual Question Generation (VQG), most images have multiple concepts (e.g. objects and categories) for which a question could be generated, but models are trained to mimic an arbitrary choice of concept as given in their training data. This makes training difficult and also poses issues for evaluation – multiple valid questions exist for most images but only one or a few are captured by the human references. We present Guiding Visual Question Generation - a variant of VQG which conditions the question generator on categorical information based on expectations on the type of question and the objects it should explore. We propose two variant families: (i) an explicitly guided model that enables an actor (human or automated) to select which objects and categories to generate a question for; and (ii) 2 types of implicitly guided models that learn which objects and categories to condition on, based on discrete variables. The proposed models are evaluated on an answer-category augmented VQA dataset and our quantitative results show a substantial improvement over the current state of the art (over 9 BLEU-4 increase). Human evaluation validates that guidance helps the generation of questions that are grammatically coherent and relevant to the given image and objects.

## 1 Introduction

In the last few years, the AI research community has witnessed a surge in multimodal tasks such as Visual Question Answering (VQA) (Antol et al., 2015; Anderson et al., 2018), Multimodal Machine Translation (Specia et al., 2016; Elliott et al., 2017; Barrault et al., 2018; Caglayan et al., 2019), and Image Captioning (IC) (Vinyals et al., 2015; Karpathy and Fei-Fei, 2015; Xu et al., 2015). Visual Question Generation (VQG) (Zhang et al., 2016; Krishna et al., 2019; Li et al., 2018), a multimodal task which aims to generate a question given an image, remains relatively under-researched despite

the popularity of its textual counterpart. Throughout the sparse literature in this domain, different approaches have augmented and/or incorporated extra information as input. For example, Pan et al. (2019) emphasised that providing the ground truth answer to a target question is beneficial in generating a non-generic question. Krishna et al. (2019) pointed out that requiring an answer to generate questions violates a realistic scenario. Instead, they proposed a latent variable model using answer categories to help generate the corresponding questions. Recently, Scialom et al. (2020) incorporated a pre-trained language model with object features and image captions for question generation.

In this work, we explore VQG from the perspective of ‘guiding’ a question generator. Guiding has shown success in image captioning (Zheng et al. (2018) and Ng et al. (2020)), and in this VQG work we introduce the notion of ‘guiding’ as conditioning a generator on inputs that match specific chosen properties from the target. We use the answer category and objects/concepts based on an image and target question as inputs to our decoder. We propose our **explicit guiding** approach to achieve this goal. We additionally investigate an **implicit guiding** approach which attempts to remove the dependency on an external actor (see more below).

The explicit variant (Section 3.1) is modelled around the notion that an actor can select a subset of detected objects in an image for conditioning the generative process. Depending on the application, this selection could be done by a human, and algorithm or chosen randomly. For example, imagine either a open-conversation chat-bot or a language learning app. In the chat-bot case, a human may show the bot a picture of something. The bot may use randomly sampled concepts from the image (e.g. an object-detected tree) to ask a human a question upon. In the language learning case, the human may wish to select certain concepts they want the generated question to reflect. For exam-

ple, they might select a subset of animal-related objects from the whole set of detected objects in order to generate questions for teaching the animal-related vocabulary in a language learning setting. Alongside the objects, the actor may also provide, or randomly sample, an answer category to the question generator.

The implicit variant (Section 3.2), on the other hand, is motivated by removing the dependency on the aforementioned actor. We provide two methodologies for our proposed implicit variant. The first uses a Gumbel-Softmax (Jang et al., 2016) to provide a discrete sample of object labels that can be used for generating a question. The second method employs a model with two discrete latent variables that learn an internally-predicted category and a set of objects relevant for the generated question, optimised with cross-entropy and variational inference (Kingma and Welling, 2014; Miao et al., 2016).

Human evaluation shows that our models can generate realistic and relevant questions, with our explicit model *almost* fooling humans when asked to determine which, out of two questions, is the generated question. Our experiments and results are presented in Section 5.

To summarise, our **main contributions** are: 1) The first work to explore guiding using object labels in Visual Question Generation; 2) A novel generative Transformer-based set-to-sequence approach for Visual Question Generation; 3) The first work to explore discrete variable models in Visual Question Generation; and 4) A substantial increase in quantitative metrics - our explicit model improves the current state of the art setups by over 9 BLEU-4 and 110 CIDEr.

## 2 Related Work

### 2.1 Visual Question Generation

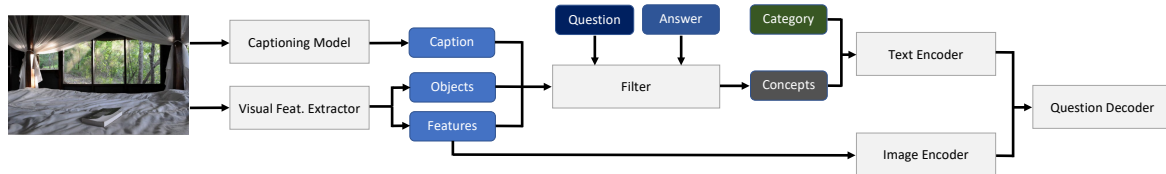
Zhang et al. (2016) introduced the first paper in the field of VQG, employing an RNN based encoder-decoder framework alongside model-generated captions to generate questions. Since then, only a handful of papers have investigated VQG. Fan et al. (2018) demonstrated the successful use of a GAN in VQG systems, allowing for non-deterministic and diverse outputs. Jain et al. (2017) proposed a model using a VAE instead of a GAN, however their improved results require the use of a target answer during inference. To overcome this unrealistic requirement, Krishna et al. (2019) augmented the VQA (Antol et al., 2015) dataset with answer

categories, and proposed a model which doesn't require an answer during inference. Because their architecture uses information from the target as input (i.e. an answer category), their work falls under our definition of guided generation. More recently, Scialom et al. (2020) investigate the cross modal performance of pre-trained language models by fine-tuning a BERT (Devlin et al., 2018) model on model-based object features and ground-truth image captions. Other work, such as Patro et al. (2018), Patro et al. (2020) and Uppal et al. (2020), either do not include BLEU scores higher than BLEU-1, which is not very informative, or address variants of the VQG task. In the latter case the models fail to beat previous SoTA on BLEU-4 for standard VQG. Recently and (Xu et al., 2021) and (Xie et al., 2021) achieve SoTA in VQG using graph convolutional networks. However, both works follow an unrealistic setup by conditioning their model on raw answers during training and inference - a dependency we attempt to remove.

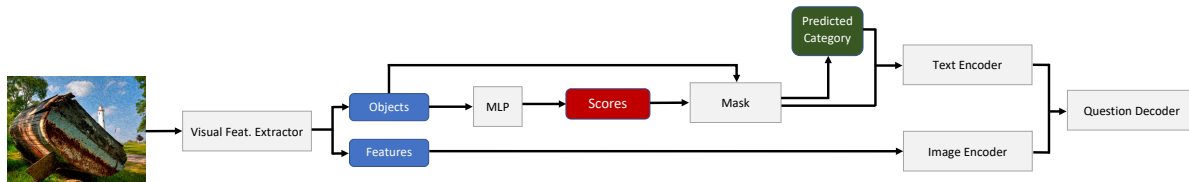
### 2.2 Discrete (Latent) Variable Models

Discrete variable models are ideal for tasks which require controllable generation (Hu et al., 2017) or 'hard' indexing of a vector (Graves et al., 2016). Existing literature provide several methods to achieve discretization. NLP GAN literature (such as SeqGAN (Yu et al., 2016) and MaskGAN (Fedus et al., 2018)) commonly use REINFORCE (Williams, 1992) to overcome differentiability issues with discrete outputs. Other discretization methodologies can be found in Variational Auto Encoder (VAE) literature (Kingma and Welling, 2014). Some older methodologies are NVIL (Mnih and Gregor, 2014) and VIMCO (Mnih and Rezende, 2016). However, VAE literature also introduced Concrete (Maddison et al., 2016), Gumbel-Softmax (Jang et al., 2016) and Vector Quantization (Oord et al., 2017) as discretization strategies (technically speaking, Concrete and Gumbel-Softmax are strongly peaked continuous distributions).

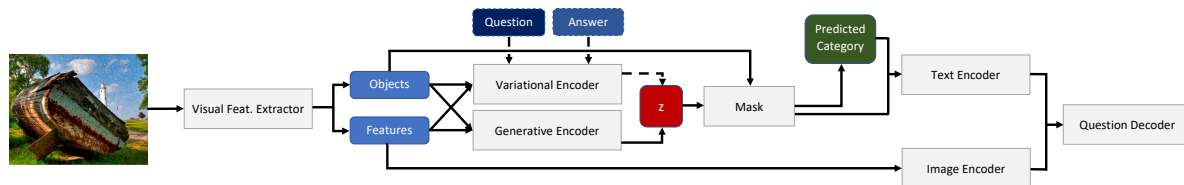
In this work, we use a Gumbel-Softmax approach to sample a distribution over objects. At inference time, given a set of object tokens, learning this 'hard' distribution allows the model to internally sample a subset of objects that produce the most informative question. Our variational model additionally learns a generative and variational distribution that allow the model to implicitly learn which objects are relevant to a question and an-



(a) Architecture of our explicit model. Given an image, first an object detection model is used to extract object labels and object features; a captioning model is used to generate relevant captions. Questions and answers are concatenated to filter the conceptual information from generated objects and captions. Next the filtered concepts are combined with the category as the input to the text encoder; the extracted object features are fed into an image encoder. Finally the outputs from the text encoder and the image encoder are fused into the decoder for question generation.



(b) Architecture of our implicit model. Similar to the explicit model, first an object detection model is used to extract object labels and object features. Object labels are sent to a non-linear MLP after which a Gumbel-Softmax is applied to obtain the discrete vector ‘Scores’. The Scores are then used to mask the object labels and predict a category. The masked object labels and predicted category are then sent to the text encoder. The outputs are fused with the image encoder outputs and sent to the decoder.



(c) Architecture of our variational implicit model. After the object detection model extracts the object labels and object features, they are sent to the variational and generative encoders. The variational encoder is used at train time only, and also receives the question and answer pair. Depending whether we’re training or in inference, we obtain a discrete vector  $z$  from the respective distribution.  $z$  is then used to mask the object labels. This variant then follows the same methodology as its non-variational counterpart. For this sub-figure only, the dashed lines indicate training.

Figure 1: Architecture of the explicit model (a) and implicit model (b)

swer pair whilst incorporating non-determinism for diverse outputs.

### 3 Methodology

We introduce the shared concepts of our explicit and implicit model variants, before diving into the variant-specific methodologies (Section 3.1 & 3.2).

For both variants, we keep the VQG problem grounded to a realistic scenario. That is, during inference, we can only provide the model with an image, and data that can either be generated by a model (*e.g.* object features or image captions) and/or trivially provided by an actor (*i.e.* answer category and a selected subset of the detected objects). However, during training, we are able to use any available information, such as images, captions, objects, answer categories, answers and target questions, employing latent variable models to minimise divergences between feature representations

of data accessible at train time but not inference time. This framework is inspired by Krishna et al. (2019). In Appendix A, we discuss the differences of input during training, testing and inference.

Formally, the VQG problem is as follows: Given an image  $\tilde{i} \in \tilde{I}$ , where  $\tilde{I}$  denotes a set of images, decode a question  $q$ . In the **guided** variant, for each  $\tilde{i}$ , we also have access to textual utterances, such as ground truth answer categories and answers. The utterances could also be extracted by an automated model, such as image captions (Li et al., 2020), or object labels and features (Anderson et al., 2018). In our work, answer categories take on 1 out of 16 categorical variables to indicate the type of question asked. For example, “*how many people are in this picture?*” would have a category of “*count*” (see Krishna et al. (2019) for more details).

**Text Encoder.** For encoding the text, we use BERT (Devlin et al., 2018) as a pre-trained lan-

guage model (PLM). Thus, for a tokenised textual input  $\tilde{S}$  of length  $T$ , we can extract a  $d$ -dimensional representation for  $\tilde{s}_t \in \tilde{S}$ :  $X = \text{PLM}(\tilde{S}) \in \mathbb{R}^{T \times d}$

**Image Encoder.** Given an image  $\tilde{i}$ , we can extract object features,  $f \in \mathbb{R}^{k_o \times 2048}$ , and their respective normalized bounding boxes,  $b \in \mathbb{R}^{k_o \times 4}$ , with the 4 dimensions referring to horizontal and vertical positions of the feature bounding box. Following the seminal methodology of Anderson et al. (2018),  $k_o$  is usually 36. Subsequent to obtaining these features, we encode the image using a Transformer (Vaswani et al., 2017), replacing the default position embeddings with the spatial embeddings extracted from the bounding box features (Krasser and Stumpf, 2020; Cornia et al., 2019). Specifically, given  $f, b$  from image  $\tilde{i}$ :  $i = \text{Transformer}(f, b) \in \mathbb{R}^{k_o \times d}$

**Text Decoder.** We employ a pretrained Transformer decoder for our task (Wolf et al., 2020). Following standard sequence-to-sequence causal decoding practices, our decoder receives some encoder outputs, and auto-regressively samples the next token, for use in the next decoding timestep. Our encoder outputs are the concatenation (; operator) of our textual and vision modality representation:  $X = [S; i] \in \mathbb{R}^{(T+k_o) \times d}$ , and our decoder takes on the form:  $\hat{q} = \text{Decoder}(X)$ , where  $\hat{q}$  is the predicted question.

In this work, we primarily focus on a set-to-sequence problem as opposed to a sequence-to-sequence problem. That is, our textual input is not a natural language sequence, rather an unordered set comprising of tokens from the answer category, the object labels, and the caption. How this set is obtained is discussed in following section. Due to the set input format, we disable positional encoding on the PLM encoder (Text Encoder in Figure 1).

### 3.1 Explicit Guiding

As mentioned in Section 1, the explicit variant requires some actor in the loop. Thus, in a real world setting, this model will run in two steps. Firstly, we run object detection (OD) and image captioning (IC) over an image and return relevant guiding information to the actor. The actor may then select or randomly sample a subset of objects which are sent to the decoder to start its generation process. If the actor opts for a random sample strategy, no human is needed during the inference process (see Appendix A for examples).

To enable this setup, we create paired data based

on the guided notion. At a high level, our approach creates this data in three steps: 1) obtain object labels; 2) obtain concepts via IC Formally,

$$\begin{aligned} \text{objects} &= \text{OD}(i) \in \mathbb{R}^{k_o} \\ \text{cap} &= \text{CaptionModel}(i) \in \mathbb{R}^{T_{cap}} \\ \text{cap} &= \text{rmStopWords}(\text{caption}) \in \mathbb{R}^{<T_{cap}} \\ \text{candidate\_concepts} &= \text{set}(\text{objects}; \text{cap}) \in \mathbb{R}^{T_{cc}} \end{aligned} \quad (1)$$

Here, OD stands for an object detector model, rmStopWords is a function which removes the stop words from a list, and set is a function which creates a set from the concatenation (the ; operator) of the detected objects and obtained captions. cap stands for caption. The set is of size  $T_{cc} < k_o + T_{cap}$ . Using this obtained candidate\_concepts set, we run our filtration process.

Once the set of candidate concepts has been constructed, we filter them to only retain concepts relevant to the target QA pair. After removing stop words and applying the set function to the words in the QA pair, we use Sentence-BERT (Reimers and Gurevych, 2019) to obtain embeddings for the candidate QA pair and candidate\_concepts (Eq 1). We subsequently compute a cosine similarity matrix between the two embedding matrices, and then select the top  $k$  most similar concepts. The chosen  $k$  concepts,  $\tilde{S}$ , are always a *strict subset* of the candidate concepts that are retrieved using automated image captioning or object detection. This process emulates the selection of objects an actor would select in an inference setting when given a choice of possible concepts, and creates paired data for the guided VQG task. We now concatenate an answer category to  $\tilde{S}$ :  $S = \text{PLM}([\tilde{S}; \text{category}]) \in \mathbb{R}^{T \times d}$ .

With text encoding  $S$ , we run the model, optimizing the negative log likelihood between the predicted question and the ground truth. Note that the concatenation in the decoder below is along the sequence axis (resulting in a tensor  $\in \mathbb{R}^{T+k_o \times d}$ ).

$$\begin{aligned} \hat{q} &= \text{Decoder}([S; i]) \\ \mathcal{L} &= \text{CrossEntropy}(\hat{q}, q) \end{aligned} \quad (2)$$

### 3.2 Implicit Guiding

We now introduce our experiments for the implicit variant for VQG. This variant differs from its explicit counterpart as it aims to generate questions using only images as the input, while internally learning to predict the relevant category

and objects. Mathematically, the explicit variant models  $\hat{q} = p(w_t|i, \tilde{S}, category, w_0, \dots, w_{t-1}; \theta)$  where  $\tilde{S}$  and *category* are obtained as described in Section 3.1. During inference, the implicit variant instead attempts to model  $\hat{q} = p(w_t|i, \tilde{e}_{obj}, e_{cat}, w_0, \dots, w_{t-1}; \theta)$  where  $\tilde{e}_{obj}, e_{cat}$  are **not** explicitly fed in to the model. Rather, they are determined internally as defined in Equation 6.

Given an image, we apply the same object detection model as in the explicit variants to extract object labels, which are then encoded using an embed layer. Formally,

$$\begin{aligned} \text{objects} &= \text{OD}(i) \in \mathbb{R}^{k_o} \\ e_{obj} &= \text{embed}(\text{objects}) \in \mathbb{R}^{k_o \times d} \end{aligned} \quad (3)$$

Since we would like the implicit model to learn relevant objects for an image internally, we project each object in  $e_{obj}$  to a real-valued score:

$$\text{scores} = \text{MLP}(e_{obj}) \in \mathbb{R}^{k_o} \quad (4)$$

Subsequently, we apply a hard Gumbel-Softmax (Jang et al., 2017) to obtain predictions over selected objects. Because Gumbel-Softmax samples from a log-log-uniform distribution, stochasticity is now present in our sampled objects. To sample  $k$  objects, we tile/repeat *scores*  $k$  times before putting it into the Gumbel-Softmax.  $\tilde{z}$ , our  $k$ -hot sampled objects vector, is then used to mask object embeddings for use in decoding:

$$\begin{aligned} \tilde{z} &= \text{gumbel-softmax}(\text{scores}, k) \in \mathbb{R}^{k_o} \\ \tilde{e}_{obj} &= \tilde{z} * e_{obj} \in \mathbb{R}^{k_o \times d} \end{aligned} \quad (5)$$

Where  $*$  denotes element-wise multiplication. Categories can also be a strong guiding factor and instead of making it an explicit input, we build a classifier to predict possible categories. In this variant,  $\tilde{e}_{obj}$  is used as an input to both our text encoder, and the MLP responsible for the category prediction:

$$\begin{aligned} S &= \text{PLM}(\tilde{e}_{obj}) \in \mathbb{R}^{k_o \times d} \\ p(\hat{cat}|\tilde{e}_{obj}) &= \text{softmax}(\text{MLP}(\tilde{e}_{obj})) \in \mathbb{R}^{k_{cat}} \end{aligned} \quad (6)$$

Using the one-hot representation of the predicted category (i.e.  $e_{cat} = \text{one-hot}(p(\hat{cat}|\tilde{e}_{obj}))$ ), we can concatenate our image, PLM representation of objects, and predicted category to feed into the decoder:  $\hat{q} = \text{Decoder}([i; S; e_{cat}]) \in \mathbb{R}^{T_{\hat{q}}}$ . However, during training, we teacher force against

the ‘gold’ set of objects,  $\tilde{S}$  (obtained using *candidate\_concepts* in Equation 1). Training and optimization thus follow:

$$\begin{aligned} \hat{q} &= \text{Decoder}([i; \tilde{S}; e_{cat}]) \in \mathbb{R}^{T_{\hat{q}}} \\ \mathcal{L} &= \text{CrossEntropy}(\hat{q}, q) + \\ &\quad \text{CrossEntropy}(p(\hat{cat}|\tilde{e}_{obj}), cat) + \\ &\quad \text{StartEnd}(\tilde{e}_{obj}, \tilde{S}) \end{aligned} \quad (7)$$

where StartEnd is a BERT QA-head style loss (Devlin et al., 2018) that uses binary cross entropy for each  $k$  in  $\tilde{e}_{obj}$ .

**Variational Implicit.** Hypothesising that ground-truth QA pairs might provide information useful to selecting objects, we additionally attempt to extend our model to incorporate QA pairs to learn a latent variational distribution over the objects. However, since QA pairs can only be used during training to learn a variational distribution, we introduce another generative distribution that is only conditioned on the images and extracted objects. We borrow the idea from latent variable models to minimise Kullback-Leibler (KL) divergence between the variational distribution and generative distribution, where the variational distribution is used during training and the generative distribution is used in inference.

Continuing from Equation 3, we build two matrices,  $M_{gen}$  and  $M_{var}$ . The former is a concatenation of the image features and object embeddings, and the latter the concatenation between the encoded QA pair and  $M_{gen}$ . Depending on whether we’re in a training or inference regime, the CLS token of the relevant matrix is used to sample a mask,  $\tilde{z}$ , which is subsequently applied on the aforementioned object embeddings:

$$\begin{aligned} M_{gen} &= \text{encode}([e_{obj}; i]) \in \mathbb{R}^{2k_o \times d} \\ e_{qa} &= \text{embed}(\text{Q;A}) \in \mathbb{R}^{T_{qa} \times d} \\ M_{var} &= \text{encode}([e_{qa}; M_{gen}]) \in \mathbb{R}^{(2k_o + T_{qa}) \times d} \\ q_{\phi}(z|M_{gen}, M_{var}) &= \text{MLP}(M_{gen}^{CLS}; M_{var}^{CLS}) \in \mathbb{R}^{k_o} \\ p_{\theta}(z|M_{gen}) &= \text{MLP}(M_{gen}) \in \mathbb{R}^{k_o} \\ \tilde{z} &= \text{gumbel-softmax}(z, k) \in \mathbb{R}^{k_o} \\ \tilde{e}_{obj} &= \tilde{z} * e_{obj} \in \mathbb{R}^{k_o \times d} \end{aligned}$$

where  $q_{\phi}(z|M_{gen}, M_{var})$  is the variational distribution,  $p_{\theta}(z|M_{gen})$  is the generative distribution, and MLP denotes a multilayer perceptron for learning the alignment between objects and QA pairs.

encode is an attention-based function such as BERT (Devlin et al., 2018). From here, our methodology follows on from Equation 6. However, our loss now attempts to minimise the ELBO:

$$\begin{aligned} \mathcal{L} = & \mathbb{E}[\log p_{\theta}(\hat{q}|z, \hat{cat})] \\ & - D_{\text{KL}}[q_{\phi}(z|M_{gen}^{CLS}, M_{var}^{CLS})||p_{\theta}(z|M_{gen}^{CLS})] \\ & + \log p(\hat{cat}|M_{var}) \end{aligned}$$

## 4 Experiments

### 4.1 Datasets

We use the VQA v2.0 dataset<sup>1</sup> (Antol et al., 2015) (CC-BY 4.0), a large dataset consisting of all relevant information for the VQG task. We follow the official VQA partition, with *i.e.* 443.8K questions from 82.8K images for training, and 214.4K questions from 40.5K images for validation. Following Krishna et al. (2019), we report the performance on validation set as the annotated categories and answers for the VQA test set are not available.

We use answer categories from the annotations of Krishna et al. (2019). The top 500 answers in the VQA v2.0 dataset are annotated with a label from the set of 15 possible categories, which covers up the 82% of the VQA v2.0 dataset; the other answers are treated as an additional category. These annotated answer categories include objects (*e.g.* “mountain”, “flower”), attributes (*e.g.* “cold”, “old”), color, counting, *etc.*

We report BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), CIDEr (Vedantam et al., 2015), METEOR (Lavie and Agarwal, 2007), and MSJ (Montahaei et al., 2019) as evaluation metrics. The MSJ metric accounts for both the diversity of generated outputs, and the n-gram overlap with the ground truth utterances.

### 4.2 Comparative Approaches

We compare our models with four recently proposed VQG models *Information Maximising VQG* (IMVQG; supervised with image and answer category) (Krishna et al., 2019), *What BERT Sees* (WBS; supervised with image and image caption) (Scialom et al., 2020), *Deep Bayesian Network* (DBN; supervised with image, scenes, image captions and tags/concepts) (Patro et al., 2020), and *Category Consistent Cyclic VQG* (C3VQG; supervised with image and answer category) (Uppal et al., 2020). We follow IMVQG’s evaluation setup

<sup>1</sup><https://visualqa.org/>

because they hold the current SoTA in VQG for realistic inference regimes. We omit (Xu et al., 2021) and (Xie et al., 2021) from our table of results because these models follow an unrealistic inference regime, requiring an explicit answer during training and inference. Our baseline is an image-only model, without other guiding information or latent variables.

### 4.3 Implementation Details

In Section 3 we described the shared aspects of our model variants. The reported scores in Section 5 use the same hyperparameters and model initialisation. A table of hyperparameters and training details can be found in Appendix B. BERT Base (Devlin et al., 2018) serves as our PLM encoder and following Wolf et al. (2020); Scialom et al. (2020), we use a pre-trained BERT model for decoding too. Though typically not used for decoding, by concatenating the encoder inputs with a [MASK] token and feeding this to the decoder model, we are able to obtain an output (*e.g.*  $\hat{q}_1$ ). This decoded output is concatenated with the original input sequence, and once again fed to the decoder to sample the next token. Thus, we use the BERT model as a decoder in an auto-regressive fashion.

To encode the images based on the Faster-RCNN object features (Ren et al., 2015; Anderson et al., 2018), we use a standard Transformer (Vaswani et al., 2017) encoder. Empirically, we find  $k = 2$  to be the best number of sampled objects.

## 5 Results

We present quantitative results in Table 1 and qualitative results in Figure 2. We evaluate the explicit, implicit and variational implicit models in a single-reference setup, as the chosen input concepts are meant to guide the model output towards one particular target reference.

### 5.1 Quantitative Results

Starting with the explicit variant, as seen in Table 1, we note that our image-only baseline model achieves a BLEU-4 score of 5.95. We test our model with different combinations of text features to identify which textual input is most influential to the reported metrics. We notice that the contribution of the category is the most important text input with respect to improving the score of the model, raising the BLEU-4 score by more than 11 points (image-category) over the aforementioned baseline.

Model	BLEU				CIDEr	METEOR	ROUGE	MSJ		
	1	2	3	4				3	4	5
	Comparative									
IMVQG (z-path) <sup>†</sup>	<b>50.1</b>	32.3	24.6	16.3	94.3	20.6	39.6	47.2	38.0	31.5
IMVQG (t-path)	47.4	29.0	19.9	14.5	86.0	18.4	38.4	53.8	44.1	37.2
WBS <sup>‡</sup>	42.1	22.4	14.1	9.2	60.2	14.9	29.1	63.2	55.7	49.7
DBN	40.7	-	-	-	-	22.6	-	-	-	-
C3VQG	41.9	22.1	15.0	10.0	46.9	13.6	42.3	-	-	-
image-only	25.9	15.9	9.8	5.9	41.4	13.5	27.8	52.2	42.8	36.0
	Explicit									
image-category	40.8	29.9	22.5	17.3	131	20.8	43.0	64.2	55.5	48.8
image-objects	34.7	25.0	19.1	15.0	130	19.4	36.9	67.4	59.2	52.7
image-guided	46.3	<b>36.4</b>	<b>29.5</b>	<b>24.4</b>	<b>214</b>	<b>25.2</b>	<b>49.0</b>	<b>71.3</b>	<b>63.6</b>	<b>57.3</b>
image-guided-random	23.6	12.1	5.75	2.39	17.6	10.8	24.2	62.3	52.6	45.0
	Implicit									
image-category	28.4	17.5	11.3	8.5	42.8	13.5	30.7	51.8	42.9	36.4
image-guided	33.8	24.0	18.3	14.2	123	19.1	35.9	66.7	58.9	52.5
image-guided-pred	25.3	14.9	9.1	6.3	27.3	11.6	27.3	52.0	44.0	38.1
image-guided-random	21.3	11.4	6.3	3.6	23.1	10.7	22.2	61.7	52.8	45.9
	Variational Implicit									
image-guided	33.9	23.5	16.8	12.6	113	18.8	35.6	64.2	56.3	49.8
image-guided-pred	22.6	12.5	6.9	4.1	24.3	11.2	23.0	58.6	49.3	42.4
image-guided-random	19.8	10.7	5.9	3.3	19.6	10.0	21.3	58.8	50	43.4

Table 1: Single reference evaluation results. “\*-guided” refers to the combination of category and objects. In the explicit variant only, objects refers to the subset of detected objects and caption keywords, filtered on the target QA pair. <sup>†</sup> indicates an unrealistic inference regime, using answers as input for question generation. <sup>‡</sup> WBS scores are from single reference evaluation based on the VQA1.0 pre-trained “Im. + Cap.” model provided by the authors.

However, whilst the BLEU-4 for the image-object variant is 2.3 points lower, it outperforms the image-category variant by 3.9 points on the diversity orientated metric MSJ-5 - indicating that the image-category variant creates more generic questions. As expected, the inclusion of both the category and objects (image-guided) outperforms either of the previously mentioned models, achieving a new state-of-the-art result of 24.4 BLEU-4. This combination also creates the most diverse questions, with an MSJ-5 of 57.3.

We also test our hypothesis that guiding produces questions that are relevant to the fed in concepts. This is tested with ‘image-guided-random’ variant. This variant is the same trained model as ‘image-guided’, but uses  $k = 2$  random concepts from a respective image instead of using the ground truth question to generate concepts. Our results show that guiding is an extremely effective strategy to produce questions related to conceptual information, with a BLEU-4 score difference of over 20 points. We refer the reader to Section 5.3 for human evaluation which again validates this hypothesis, and Section 3.1 for an explanation of why guiding is valid for evaluating VQG models.

We evaluate the implicit models as follows. The

implicit image-category variant does not predict any objects internally. It uses all image features and object embeddings alongside the category supervision signal as described in Equation 7. The implicit image-guided models use the ‘gold’ objects at inference (See Section 3.1). If these variants fit the ‘gold’ objects well, it indicates that their generative abilities are suitable for guiding/conditioning on predicted or random objects. The image-guided-pred variants are evaluated using internally predicted objects - and the model variant that would be used in a real inference setting. Finally, the image-guided-random variants are fed in random object labels at inference.

For implicit guiding to be a valid methodology, we need to validate two criteria: 1) Successfully conditioning the decoder on guiding information; 2) Better than random accuracy of object prediction/selection. Note that intuitively, the implicit model is expected to perform worse than the explicit model in terms of the language generation metrics. This is because of the inherently large entropy of the relevant answer category and the objects given an image. However, if the learned distributions over the categories and objects can capture the relevant concepts of different images,

they may benefit the question generation when compared with image-only.

According to Table 1, by predicting just an answer category and no objects (image-category), the proposed implicit model beats the image-only baseline. The BLEU-4 score difference is less than 1 with the best performing WBS model (Scialom et al., 2020) – which also generates questions without explicit guided information.

As mentioned above, we can evaluate the implicit model by either feeding the ‘gold’ objects obtained as described in Section 3.1, or by the internally predicted objects as described in Section 3.2. These form the variants image-guided and image-guided-pred respectively. For both the implicit and variational implicit models, image-guided is expected to perform the best. Results validate this, showing a performance of 14.2 and 12.6 BLEU-4 respectively. Importantly, the relatively high scores of these guided models (compared to the comparative approaches) show that these models can successfully be conditioned on guiding information.

We also notice that for both types of implicit models, image-guided-pred outperforms image-guided-random. Specifically for the non-variational implicit, we see a higher BLEU-4 score difference of 2.7. Interestingly, despite this BLEU-4 difference being higher than its variational counterpart, there is a trade-off for the diversity-orientated MSJ metric. This indicates that although generated questions are discretely ‘closer’ to the ground truth, similar phrasing is used between the generated questions. In fact, an acute case of this phenomena occurs for the image-category variant where the BLEU-4 variant is higher than image-guided-pred or image-guided-random. In this case, qualitative analysis shows us that the higher BLEU-4 score can be attributed to the generic nature of the generated question. Failure cases of automatic evaluation metrics in NLP is discussed further in (Caglayan et al., 2020).

To satisfy the ‘better than random accuracy of object prediction/selection’ criteria previously outlined, we measure the overlap of the  $k$  predicted objects vs  $k$  ‘gold’ object labels. These ‘gold’ object labels are obtained similarly to the explicit variant (Section 3.1), however the caption tokens are not fed to the filtering process. Random accuracy for selecting objects is 12.5%. Our overlap accuracy on implicit image-pred is 18.7% - outperforming random selection. Variational implicit image-pred

	Baseline	Implicit	V-Implicit	Explicit
Experiment 1	34.3% ± 0.1	47.1% ± 0.12	36.7% ± 0.08	44.9% ± 0.08
Experiment 2	95.9% ± 0.03	76.6% ± 0.16	89% ± 0.09	93.5% ± 0.06
Experiment 3	-	-	-	77.6% ± 0.09
Experiment 4	-	-	-	74.1%/40.0% ± 0.07/0.18

Table 2: Human evaluation results (and standard dev.)

failed to outperform random accuracy.

## 5.2 Qualitative Results

Qualitative results are shown in Figure 2 and Appendix D. Figure 2 depicts how outputs from different model variants compare to ground truth questions. Without any guiding information, the image-only variant is able to decode semantic information from the image, however this leads to generic questions. The implicit variant, for which we also report the predicted category and objects, mostly generates on-topic and relevant questions. Focusing on the explicit variant, we witness high-quality, interesting, and on-topic questions.

Appendix D depicts how well our explicit image-guided variant handles a random selection of detected objects given the image. This experiment intends to gauge the robustness of the model to detected objects which may fall on the low tail of the human generating question/data distribution. To clarify, humans are likely to ask commonsense questions which generally focus on obvious objects in the image. By selecting objects at random for the question to be generated on, the model has to deal with object permutations not seen during training, and categories that are invalid for an image.

Analysing the outputs, when viable categories and objects that are expected to fall in a commonsense distribution are sampled, the model can generate high quality questions. Interestingly, we observe that when the sampled objects are not commonsense (e.g. “ears arms” for the baby and bear picture), the model falls back to using the object features instead of the guiding information. This phenomenon is also witnessed when the sampled category does not make sense for the image (e.g. category ‘animal’ in image 531086). Despite the category mismatch, the model successfully uses the object information to decode a question.

## 5.3 Human Evaluation

We ask seven humans across four experiments to evaluate the generative capabilities of our models. *Experiment 1* is a visual Turing test: given an image, a model generated question and a ground truth question, we ask a human to determine which ques-






						
		85187	63958	447043	82259	38083
Ground Truth		is this in the usa?	is the zebra tall or short?	where is this building?	is this person airborne?	how many containers of strawberries are there?
Baseline	Generated Question:	is this a busy street?	what is the zebra doing?	is this a church?	what is the man doing?	what is the green vegetable?
Implicit	Predicted Category: Predicted Objects:	count street, statues	binary zebra, ground	other building, tower	colour person, people	binary strawberries, strawberry
	Generated Question:	how many red poles are there?	is the rock in the background?	what is the weather like?	what is the color of the skier?	how many carrots are there?
Variational Implicit	Predicted Category: Predicted Objects:	binary truck, man	binary legs, shadow	object statue, tower	binary ski pole, poles	count apple, leaves
	Generated Question:	is this a new truck?	is this a zoo?	what color is the clock?	is this a ski park?	how many different types of fruit are there?
Explicit	Given Category: Given Objects:	binary statues, pillar	attribute zebra, tail	other building, tower	binary snow, skis	count strawberries, strawberry
	Generated Question:	is this a rural setting?	what pattern is on the zebra's mohawk?	what is the condition of the building?	are these people going to ski down a mountain?	how many different kinds of produce are in the bowl?

Figure 2: Qualitative Examples. The ground truth is the target question for the baseline, implicit and explicit. The examples of explicit variant uses `image-guided` whereas the implicit is using the non-variational `image-pred`.

tion they believe is model generated. *Experiment 2* attempts to discern the linguistic and grammatical capabilities of our model by asking a human to make a binary choice about whether the generated question seems natural. *Experiment 3* shows a human an image alongside a model generated question (explicit variant). Then, we ask the human to make a choice about whether the generated question is relevant to the image (*i.e.* could an annotator have feasibly asked this question during data collection). Finally, *experiment 4* judges whether objects are relevant to a generated question. The experiment is set up with true-pairs and adversarial-pairs. True-pairs are samples where the shown objects are the ones used to generate the question. Adversarial-pairs show a different set of objects than those which generated the question. If more true-pairs are marked correct (*i.e.* if at least one of the objects is relevant to the generated question) than the adversarial-pairs, then our model successfully generates questions on guiding information.

In *experiment 1*, a model generating human-level questions should be expected to score 50%, as a human would not be able to reliably distinguish them from the manually created questions. Our results show the explicit and non-variational implicit model outperforming the variational implicit and baseline variants, fooling the human around 45% of the time. Whilst not yet at the ideal 50%, the explicit approach provides a promising step towards beating the visual Turing Test. *Experiment 2*

evaluates the grammatical plausibility of the generated questions. In general, all models perform extremely well in this experiment, with the baseline variant generating grammatically correct sentences 96% of the time. This is expected, as the baseline typically falls back to decoding easy/generic questions. *Experiment 3*, is evaluated on our best performing model (explicit image-guided). Here, 78% of the generated questions are marked as relevant/on-topic given an image. Finally, *experiment 4*'s results show true-pairs marked as correct vs adversarial-pairs (incorrectly) marked as correct. Since the former is larger than the latter - 72% vs 42%, the model can successfully use guiding/object information to create on-topic questions.

## 6 Conclusions

We presented a guided approach to visual question generation (VQG), which allows for the generation of questions that focus on specific chosen aspects of the input image. We introduced three variants for this task, the explicit, implicit, and variational implicit. The former generates questions based on an explicit answer category and a set of concepts from the image. In contrast, the latter two discretely predict these concepts internally, receiving only the image as input. The explicit model achieves SoTA results when evaluated against comparable models. Qualitative evaluation and human-based experiments demonstrate that both variants produce realistic and grammatically valid questions.

## Acknowledgments

Lucia Specia, Zixu Wang and Yishu Miao received support from MultiMT project (H2020 ERC Starting Grant No. 678017) and the Air Force Office of Scientific Research (under award number FA8655-20-1-7006).

## References

- Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. VQA: Visual Question Answering. In *International Conference on Computer Vision (ICCV)*.
- Loïc Barrault, Fethi Bougares, Lucia Specia, Chiraag Lala, Desmond Elliott, and Stella Frank. 2018. Findings of the third shared task on multimodal machine translation. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 304–323, Belgium, Brussels. Association for Computational Linguistics.
- Ozan Caglayan, Pranava Madhyastha, and Lucia Specia. 2020. Curious Case of Language Generation Evaluation Metrics: A Cautionary Tale. pages 2322–2328.
- Ozan Caglayan, Pranava Madhyastha, Lucia Specia, and Loïc Barrault. 2019. Probing the need for visual context in multimodal machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4159–4170, Minneapolis, Minnesota. Association for Computational Linguistics.
- Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. 2019. Meshed-Memory Transformer for Image Captioning. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 10575–10584.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.
- Desmond Elliott, Stella Frank, Loïc Barrault, Fethi Bougares, and Lucia Specia. 2017. Findings of the second shared task on multimodal machine translation and multilingual image description. In *Proceedings of the Second Conference on Machine Translation*, pages 215–233, Copenhagen, Denmark. Association for Computational Linguistics.
- Zhihao Fan, Zhongyu Wei, Siyuan Wang, Yang Liu, and Xuanjing Huang. 2018. A Reinforcement Learning Framework for Natural Question Generation using Bi-discriminators. Technical report.
- William Fedus, Ian Goodfellow, and Andrew M. Dai. 2018. MaskGAN: Better Text Generation via Filling in the \_\_\_\_\_. *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*.
- Alex Graves, Greg Wayne, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwińska, Sergio Gómez Colmenarejo, Edward Grefenstette, Tiago Ramalho, John Agapiou, Adrià Puigdomènech Badia, Karl Moritz Hermann, Yori Zwols, Georg Ostrovski, Adam Cain, Helen King, Christopher Summerfield, Phil Blunsom, Koray Kavukcuoglu, and Demis Hassabis. 2016. Hybrid computing using a neural network with dynamic external memory. *Nature*, 538(7626):471–476.
- Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017. Toward Controlled Generation of Text. Technical report.
- Unnat Jain, Ziyu Zhang, and Alexander Schwing. 2017. Creativity: Generating Diverse Questions using Variational Autoencoders. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, 2017-January:5415–5424*.
- Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical Reparameterization with Gumbel-Softmax. *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*.
- Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical reparameterization with gumbel-softmax.
- Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Diederik P. Kingma and Max Welling. 2014. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*. International Conference on Learning Representations, ICLR.
- Krasser and Stumpf. 2020. fairseq-image-captioning. <https://github.com/krasserm/fairseq-image-captioning>.
- Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. 2019. Information Maximizing Visual Question Generation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019-June:2008–2018.
- Alon Lavie and Abhaya Agarwal. 2007. METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 228–231, Prague, Czech Republic. Association for Computational Linguistics.

- Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. 2020. [Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks](#). *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12375 LNCS:121–137.
- Yikang Li, Nan Duan, Bolei Zhou, Xiao Chu, Wanli Ouyang, Xiaogang Wang, and Ming Zhou. 2018. [Visual question generation as dual task of visual question answering](#). *CVPR*.
- Chin-Yew Lin. 2004. [ROUGE: A package for automatic evaluation of summaries](#). In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.
- Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. 2016. [The Concrete Distribution: A Continuous Relaxation of Discrete Random Variables](#). *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*.
- Yishu Miao, Lei Yu, and Phil Blunsom. 2016. [Neural variational inference for text processing](#). In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1727–1736, New York, New York, USA. PMLR.
- Andriy Mnih and Karol Gregor. 2014. [Neural Variational Inference and Learning in Belief Networks](#). *31st International Conference on Machine Learning, ICML 2014*, 5:3800–3809.
- Andriy Mnih and Danilo J. Rezende. 2016. [Variational inference for Monte Carlo objectives](#). *33rd International Conference on Machine Learning, ICML 2016*, 5:3237–3248.
- Ehsan Montahaee, Danial Alihosseini, and Mahdiah Soleymani Baghshah. 2019. [Jointly measuring diversity and quality in text generation models](#).
- Edwin G Ng, Bo Pang, Piyush Sharma, Radu Soricut, and Google Research. 2020. [Understanding Guided Image Captioning Performance across Domains](#).
- Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. 2017. [Neural Discrete Representation Learning](#). *Advances in Neural Information Processing Systems*, 2017-December:6307–6316.
- Liangming Pan, Wenqiang Lei, Tat-Seng Chua, and Min-Yen Kan. 2019. [Recent Advances in Neural Question Generation](#).
- 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.
- Badri N. Patro, Sandeep Kumar, Vinod K. Kurmi, and Vinay P. Namboodiri. 2018. [Multimodal differential network for visual question generation](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018*, pages 4002–4012. Association for Computational Linguistics.
- Badri N. Patro, Vinod K. Kurmi, Sandeep Kumar, and Vinay P. Namboodiri. 2020. [Deep Bayesian Network for Visual Question Generation](#). *Proceedings - 2020 IEEE Winter Conference on Applications of Computer Vision, WACV 2020*, pages 1555–1565.
- Nils Reimers and Iryna Gurevych. 2019. [Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks](#). *EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference*, pages 3982–3992.
- Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. [Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks](#). Technical report.
- Thomas Scialom, Patrick Bordes, Paul-Alexis Dray, Jacopo Staiano, and Patrick Gallinari. 2020. [What BERT Sees: Cross-Modal Transfer for Visual Question Generation](#).
- Lucia Specia, Stella Frank, Khalil Sima'an, and Desmond Elliott. 2016. [A shared task on multimodal machine translation and crosslingual image description](#). In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 543–553, Berlin, Germany. Association for Computational Linguistics.
- Shagun Uppal, Anish Madan, Sarthak Bhagat, Yi Yu, and Rajiv Ratn Shah. 2020. [C3VQG: Category Consistent Cyclic Visual Question Generation](#). *arXiv*.
- 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. L. Zitnick, and Devi Parikh. 2015. [Cider: Consensus-based image description evaluation](#). *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4566–4575.
- Oriol Vinyals, A. Toshev, S. Bengio, and D. Erhan. 2015. [Show and tell: A neural image caption generator](#). *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3156–3164.
- Ronald J. Williams. 1992. [Simple statistical gradient-following algorithms for connectionist reinforcement learning](#). *Machine Learning*, 8(3-4):229–256.

- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. [Transformers: State-of-the-art natural language processing](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.
- Jiayuan Xie, Yi Cai, Qingbao Huang, and Tao Wang. 2021. [Multiple Objects-Aware Visual Question Generation](#). *MM 2021 - Proceedings of the 29th ACM International Conference on Multimedia*, pages 4546–4554.
- Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. [Show, attend and tell: Neural image caption generation with visual attention](#). In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 2048–2057, Lille, France. PMLR.
- Xing Xu, Tan Wang, Yang Yang, Alan Hanjalic, and Heng Tao Shen. 2021. [Radial graph convolutional network for visual question generation](#). *IEEE Transactions on Neural Networks and Learning Systems*, 32(4):1654–1667.
- Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2016. [SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient](#). *31st AAAI Conference on Artificial Intelligence, AAAI 2017*, pages 2852–2858.
- Shijie Zhang, Lizhen Qu, Shaodi You, Zhenglu Yang, and Jiawan Zhang. 2016. [Automatic Generation of Grounded Visual Questions](#). *IJCAI International Joint Conference on Artificial Intelligence*, pages 4235–4243.
- Yue Zheng, Yali Li, and Shengjin Wang. 2018. [Intention Oriented Image Captions with Guiding Objects](#). *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019-June:8387–8396.

## A Training, testing and inference

Here, using an example, we clarify the inputs to our explicit model (Section 3.1) in the training, testing and inference setups.

### Training

- Ground truth question: What is the labrador about to catch?
- Answer: Frisbee
- Category: Object
- Image:  $i \in \mathbb{R}^{k_o \times d}$
- {Caption}: A man throwing a frisbee to a dog
- {Objects}: person, dog, frisbee, grass

N.B. {Caption} and {Objects} are both model generated, requiring only an image as input. These inputs are thus available at inference time.

Firstly, we create a set of *candidate\_concepts* (see eq. 1) from the caption and objects: [person, dog, frisbee, grass, man, throwing] ( $\in \mathbb{R}^6$ ). These words are individually embedded. Secondly, we concatenate and embed the set of question and answer tokens ( $\in \mathbb{R}^7$ ).

Then, we construct a matrix which gives us cosine similarity scores for each *candidate\_concepts* token to a QA token ( $\in \mathbb{R}^{6 \times 7}$ ). We choose  $k = 2$  tokens from the *candidate\_concepts* which are most similar to the words from the QA. Here, “dog” and “frisbee” are likely chosen. Our input to the model is then  $\langle i, \text{“object”}, \text{“dog”}, \text{“frisbee”} \rangle$ .

Notice that it is possible for these words to be in the QA pair (e.g. “frisbee”). Importantly, these words have not been fed from the QA pair - they have been fed in from model-obtained concepts ({Object} and {Caption}). Philosophically similar, Krishna et al. (2019) constructed inputs based on target information for use in training and benchmarking.

**Testing.** Imagine a data labeler creating questions based on an image. They would look at the image, and decide on the concepts to create the question for. Our testing methodology follows this intuition using the strategy outlined above: the  $k = 2$  selected objects from *candidate\_concepts* is a programmatic attempt for selecting concepts which *could* generate the target question. Note that there can be many questions generated for a subset of

concepts (e.g. ‘is the dog about to catch the frisbee?’, ‘what is the flying object near the dog?’ etc.). As outlined above, we are not taking concepts from the target. Rather we use information from the target to emulate the concepts an actor would think of to generate the target question. Because there can be different concepts questions are based on for one image (see ground-truth questions in Appendix D), our strategy allows us to generate questions which might be similar to a singular target question. This leads to an evaluation which fairly uses information a human has access to to generate a question.

**Inference.** However, in the real world, there is no ‘ground-truth’ question. In this case, we simply feed image features, and actor selected concepts to our question generator model. The selection process of the actor may be random - in which case a human agent does not need to be involved in the question generation process. The  $k \leq 2$  selected concepts here are a subset of *candidate\_concepts*, which are fully generated from models.

## B Hyperparameters and training details

Batch size	128
Learning rate	1e-5
Text model layers	12
Text model dimension	768
Image encoder layers	6
Image encoder dimension	768
Image encoder heads	8

Table 3: Hyperparameters for our model variants.

Empirically, for both variants, we find  $k = 2$  to be the best number of sampled objects. All experiments are run with early stopping (patience 10; training iterations capped at 35000) on the BLEU-4 metric. Scores reported (in Section 5) are from the highest performing checkpoint. We use the PyTorch library and train our model on a V100 GPU (1.5 hours per epoch).

## C Impact of model size on results

Model	BLEU				CIDEr	METEOR	ROUGE
	1	2	3	4			
image-category	38.6	28.4	21.4	16.2	118	19.9	40.1
image-guided	44.5	34.4	27.4	22.1	197	24.6	47

Table 4: Truncated models single reference evaluation results.

Our models use the heavier Transformers than previous SoTA we compare to. For example, (Krishna et al., 2019) use ResNet and RNNs for their image encoder and question generator ( $\sim 18\text{M}$  parameters). Our models have between 200-300M parameters. To validate that our results are not purely attributable to model size, we train a truncated version of image-category and image-guided (explicit only). We truncate our models by using only the first and last layers of our BERT based encoders and decoders ( $\sim 36\text{M}$  parameters). Our closest model to theirs is the (truncated) explicit image-category, which achieves a BLEU-4 of 16.2 as seen in Table 4 - an improvement of 1.7 BLEU-4 over IMVQG’s *t-path*. Even if we attribute 100% of this score improvement to the pre-trained nature of the BERT models we use, our methodology still introduces a 5.9 BLEU-4 increase over the image-category combination (truncated image-guided achieves a BLEU-4 of 22.1).

## D More Qualitative Examples.

Examples can be seen in Figure 2 (next page). When examined, we see that the generated question accurately uses the guiding category when the category is valid for the given image. For example, 531086/1 has animal as the sampled category. Because no animal is present in the image, this category isn’t valid for the image. The generated question then correctly relies on the object labels and visual modality to generate a valid question given the image. Similarly for 490505/2.

There are some cases where a sampled object/concept is not valid given an image. For example, at least one of the objects in 22929/1, 41276/1, 531086/2, 281711/1, 490505/1 is not valid. In this case the model usually relies on the other available guiding information, prioritising the category information (e.g. 531086/2). In rare cases, the model has failure cases where some of the valid sampled objects may not be used in the generated question (e.g. 293705/2 and 490505/2).

The concept extractor utilises a pre-trained image captioning model and object detector model. This may lead to an accumulation of downstream errors, especially if the data fed into the pre-trained models are from a significantly different data generating distribution than those used to train the model. In this erroneous case, the model will likely fallback to rely on the image modality and category information to produce a generic question

(e.g. 22929/1, 22929/2, 531085/1, 293705/2).

## E Responsible NLP Research

### E.1 Limitations

Our approach claims to achieve SoTA in Visual Question Generation. However, we are only able to train and test our model on one dataset because it is the only existing dataset which contains answer categories. It is possible that our work may be suitable for use in a zero-shot setting, but we have not evaluated or tested our model in this setup.

### E.2 Risks

Our model could be used to generate novel questions for use in Visual Question Answering. This may have a knock-on effect which leads to training more VQA models, thus having a negative impact on the environment.

Our model could be used in downstream tasks such as language learning. There may be incorrectness in the generated questions which has a knock on effect to a user using this model (e.g. the user may gain a wrong understanding of a concept because of a question the model has generated)




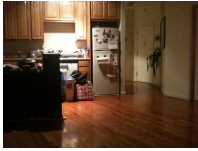


		 22929	 41276	 531086
Ground Truths (truncated @ 5)		is the bear bigger than the baby? is the baby showing the bear love? <b>what</b> is the child sitting with?	<b>how</b> many planes are there? is this a boeing 737? is the sky blue? <b>how</b> many propellers on the plane? is this a single engine aircraft? is the landing gear visible?	<b>do</b> the cabinets have handles? <b>what</b> room is this? <b>do</b> they wash dishes by hand? <b>are</b> there any magnets on the fridge? is there a coffee maker in the photo?
Explicit	<b>1. Sampled Category:</b> <b>1. Sampled Objects:</b> <b>1. Generated Question:</b>	spatial baby jacket which of the two bears's arms is closer to the camera?	material cockpit tail what is the landing gear made of?	animal door wall what is the only colorful object on the wall?
	<b>2. Sampled Category:</b> <b>2. Sampled Objects:</b> <b>2. Generated Question:</b>	activity ears arm what is the baby doing?	count wings sky how many clouds are in the sky?	binary book door is there a dishwasher in the picture?
	Valid category for image? On topic with category? On topic with objects? Valid question for image?	✓ ✓ ✓ ✓ ✗ ✗ ✓ ✓	✓ ✓ ✓ ✓ ✗ ✓ ✗ ✓	✗ ✓ ✗ ✓ ✓ ✗ ✓ ✓
		 281711	 490505	 293705
Ground Truths (truncated @ 5)		<b>where</b> are the paper towels hanging at? is this a museum? is there a plant? <b>what</b> type of flooring do you see?	<b>what</b> color is not included in the roses? <b>has</b> the envelope been opened? <b>what</b> color is the envelope?	<b>is</b> the electric bill being paid? <b>would</b> you pay money for staging? <b>what</b> color is the sofa?
Explicit	<b>1. Sampled Category:</b> <b>1. Sampled Objects:</b> <b>1. Generated Question:</b>	attribute television door is the refrigerator door open or closed?	location ground flowers where is the vase?	attribute living room, books what pattern is on the couch?
	<b>2. Sampled Category:</b> <b>2. Sampled Objects:</b> <b>2. Generated Question:</b>	location counter magnet what is on top of the counter?	food flower counter what is the red and white item?	shape counter leg what shape is the rug?
	Valid category for image? On topic with category? On topic with objects? Valid question for image?	✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓	✓ ✗ ✓ ✗ ✓ ✗ ✗ ✗	✓ ✓ ✓ ✓ ✓ ✗ ✓ ✓

Figure 3: Qualitative outputs from explicit variant being fed random guiding information. Failure cases are also shown.