Visual Clues: Bridging Vision and Language Foundations for Image Paragraph Captioning

Anonymous Author(s)
Affiliation
Address
email

Abstract

People say, "*A picture is worth a thousand words*". Then how can we get the rich information out of the image? We argue that by using *visual clues* to bridge large pretrained vision foundation models and language models, we can do so without any extra cross-modal training. Thanks to the strong zero-shot capability of foundation models, we start by constructing a rich semantic representation of the image (*e.g.*, image tags, object attributes / locations, captions) as a structured textual prompt, called *visual clues*, using a vision foundation model. Based on visual clues, we use large language model to produce a series of comprehensive descriptions for the visual content, which is then verified by the vision model again to select the candidate that aligns best with the image. We evaluate the quality of generated descriptions by quantitative and qualitative measurement. The results demonstrate the effectiveness of such a structured semantic representation.

1 Introduction

	This image is of a family celebrating Christmas. They are all gathered around a dinner table, with a turkey and other food on it. The family is smiling and seems to be enjoying themselves. There is a Christmas tree in the background and some Christmas lights on the walls.
	This image features a horse and buggy travelling down a road in the town of Holguin, Cuba. The horse is harnessed to the buggy and is pulling it along, while a farmworker rides behind. This image captures the everyday life of Cubans, with their traditional horse-drawn carts still in use.

Figure 1: Examples of generated image paragraph.

> "*Vision is a process that produces from images of the external world a* ***description*** *that is useful to the viewer and not cluttered with irrelevant information.*"
>
> —David Marr, *Vision, p31*

What makes a good "*description*" for vision? Over the past several decades, computer vision pioneers drew inspiration from neural science, cognitive science, and psychophysics (Marr, 2010), pointing us to the North Stars (Fei-Fei and Krishna, 2022), some among them being image classification and object detection. Despite the tremendous progress that has been made, much of these object-centric works remain a proxy for an eventual task or application that requires a holistic view of the visual content, involving concepts beyond objects: actions, attributes, and relations, to name a few.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

In our work, we argue that textual representation suffices such "description". It brings forth a more holistic visual representation than categorical labels. It allows machines to interpret visual signals through descriptive captions (Zhou et al., 2020b; Li et al., 2022), and perform more language-heavy tasks such as question-answers (Rajpurkar et al., 2016), or multi-round dialogues (Li et al., 2017). On the other hand, the access to abundant web multimodal language data (*e.g.*, image alt-text, video subtitles) provides us with the fuel for powering neural visual representations from contrastive language-image pre-training (CLIP, Yuan et al. (2021); Radford et al. (2021)). The marriage of the two renders a new computer vision system that is faithful, generic, and versatile.

We name this new computer vision system **BEST**, for Bridging with Explicit Structured Textural clues. We start by constructing a semantic representation of the image. This semantic representation, which we referred to as *visual clues*, comprises rich semantic components, from object and attribute tags to localized detection regions and region captions. Powered by the recent advances in vision foundation model Florence (Yuan et al., 2021), the visual clues are rich in open-vocabulary expressions, marking a major difference compared to existing symbolic approaches (*e.g.*, scene graphs Krishna et al. (2017)) with closed-set vocabularies.

The visual clues are interpretable, not only for humans, but also for machines. Take the generative language model GPT-3 (Brown et al., 2020). The visual clues could be digested by GPT-3, which in return produces crisp language descriptions that are sensible to the viewer while not cluttered with irrelevant information from the visual clues. Whereas this open-loop process could potentially suffer from object hallucination issues (Maynez et al., 2020; Zhou et al., 2020a) as the outputs from GPT-3 are not governed by any means, we further deploy a closed-loop verification procedure that grounds descriptions back to the original image.

To evaluate the quality of the language descriptions, we resort to an existing task named Image Paragraph Captioning (IPC), but with a twist. IPC aims to address the demand for generating long, coherent, and informative descriptions of the whole visual content of an image (Krause et al., 2017), which can eventually be used for many applications including poetry composition (Liu et al., 2018), automatic recipe generation (Salvador et al., 2019), visual storytelling (Huang et al., 2016), advertisement generation, or help blind or visually-impaired people see better. The existing metrics for IPC such as BLEU (Papineni et al., 2002), METEOR (Denkowski and Lavie, 2014), and CIDEr (Vedantam et al., 2015) encourage exact matching between semantics in generated captions and those in the reference. However, they over penalize visual details that are not annotated thus compromising their qualifications for measuring overall representation quality. Inspired by Anderson et al. (2016); Krishna et al. (2017), we propose to measure the accuracy on *scene graphs* extracted from generated text against human-annotated graphs, which, as suggested by Anderson et al. (2016), co-relates better with human judgment.

The contributions are twofold. First, we propose a general framework for semantic visual representation and showcase its application to image paragraph captioning. The framework is simple yet highly extendable, allowing new components to be plug-in and supporting other use scenarios that require a holistic view of the visual content. Second, we benchmark the effectiveness of the proposed model on its capacity for representing visual concepts (*e.g.*, scene graphs) and set new state-of-the-art results.

Notations. We denote $\langle \cdot, \cdot \rangle$ as inner product between two vectors, $|\mathcal{A}|$ as the cardinality of set $\mathcal{A}$.

2 Related Works

Image paragraph generation. The task of generating image paragraphs is first introduced by Krause et al. (2017). Conditioned on the visual features, they first train a sentence recurrent neural network (RNN) to output sentence topics, and then feed each of the topics into another RNN to generate the paragraphs. Liang et al. (2017) further improve the hierarchical RNN framework by introducing an adversarial discriminator for smoother transitions between sentences. Chatterjee and Schwing (2018) also address cross-sentence topic consistency by a global coherence vector. Melas-Kyriazi et al. (2018) add a repeat penalty to the optimization, to prevent the appearance of repeated sequences. Wang et al. (2019) use convolutional auto-encoder for topic modeling based on region-level image features. Along this line, many other works have been done (Dai et al., 2017; Luo et al., 2019; Mao et al., 2018; Xu et al., 2020; Guo et al., 2021; Shi et al., 2021). Most of the proposed models, however, are trained on *Stanford image-paragraph dataset* (Krause et al., 2017), which only contains 14 thousand of training paragraphs for its expensive nature to collect. Due to lack of data, the generated paragraphs usually lack coherence both locally and globally. Therefore, many of the above works aim to make the best use of data to improve the coherence. Yet nowadays,

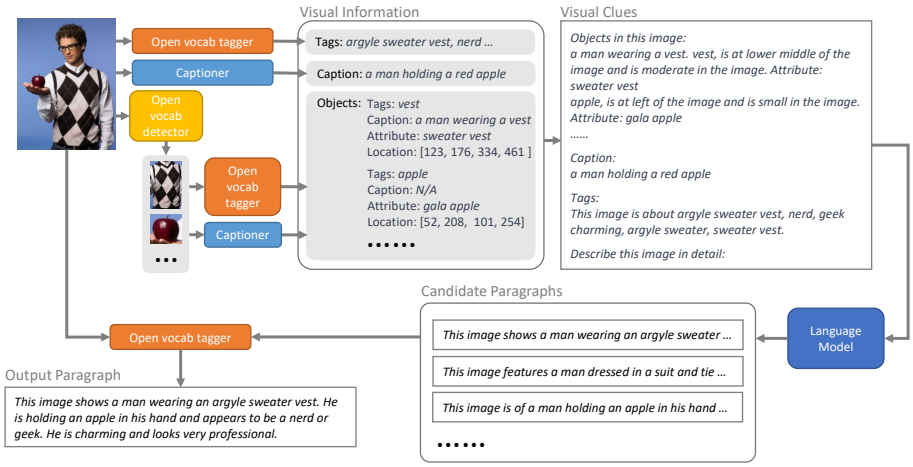


Figure 2: Framework demonstration. The orange *open vocab tagger* box corresponds to the image encoder $f_{\text{v}}(\cdot)$ and the text encoder as $f_{\text{t}}(\cdot)$. The blue *captioner* box is the caption model $c(\cdot)$.

large language models can generate long coherent paragraphs by default. Our work, leveraging recent progress of large pretrained models, focuses more on how to guide and constrain the generated text instead.

Constrained text generation In recent years, rapid progress has been made in vision-language pretraining (VLP). CLIP (Radford et al., 2021), ALIGN (Jia et al., 2021), and Florence (Yuan et al., 2021) are proposed to encode vision and language into a *joint* representation space for crossmodal alignment tasks, *e.g.*, zero-shot image classification. Another line of research, *e.g.*, SimVLM (Wang et al., 2021), FLAVA (Singh et al., 2021), BLIP (Li et al., 2022), CoCa (Yu et al., 2022) and many others (Cho et al., 2021; Wang et al., 2022; Zhu et al., 2021; Alayrac et al., 2022) adopt encoder-decoder models trained with generative losses. Those models are capable of performing image captioning in a zero-shot manner. A concurrent work, Socratic Models (SM, Zeng et al. (2022)), also use textual data to bridge the domain gap between vision-language models and language models. The model, however, is stronger in retrieval tasks than captioning tasks as we will show later. There are also other works leveraging large language models to solve vision tasks, *e.g.*, PICa (Yang et al., 2021) uses GPT-3 (Brown et al., 2020) to extract commonsense knowledge for visual question answering tasks, MAGIC (Su et al., 2022) uses a CLIP-induced score to regularize the language generation so that it is semantically related to the given image, and VisualGPT (Chen et al., 2022) employs a self-resurrecting encoder-decoder attention mechanism to adapt the language models with a small amount of in-domain image-text data.

3 Framework

Given an image I, our goal is to generate long and coherent descriptive text based on image inputs, leveraging only the existing pretrained models. Our framework can be divided into three stages:

1. Represent I with visual clues S, which contain the rich visual information;
2. Feed the visual clues into a language model to generate K candidate paragraphs $\{T_i\}_{i=1}^{K}$;
3. Select the best paragraph T^* from the candidates $\{T_i\}_{i=1}^{K}$.

The overall framework is illustrated in Figure 2. We will then elaborate on each of them.

3.1 Visual Clue Extraction

We leverage three state-of-the-art models with the open-vocabulary capability to extract the visual information, namely, the concise tags, the short captions, and the local descriptions.

Concise tags. The first model we use is the contrastively trained vision-language models, *e.g.*, CLIP (Radford et al., 2021), Florence (Yuan et al., 2021). Such models are pretrained on image-text pairs $\{x_i, y_i\}$, and is composed of the image encoder $f_{\text{v}}(\cdot)$ and the text encoder $f_{\text{t}}(\cdot)$. Given a minibatch $\mathcal{B}$, the models are optimized by contrastive loss

$$\mathcal{L} = -\frac{1}{|\mathcal{B}|} \sum_{x_i, y_i \in \mathcal{B}} \left(\frac{\exp(\langle f_{\text{v}}(x_i), f_{\text{t}}(y_i) \rangle / \tau)}{\sum_{y_j \in \mathcal{B}, j \neq i} \exp(\langle f_{\text{v}}(x_i), f_{\text{t}}(y_j) \rangle / \tau)} + \frac{\exp(\langle f_{\text{v}}(x_i), f_{\text{t}}(y_i) \rangle / \tau)}{\sum_{x_j \in \mathcal{B}, j \neq i} \exp(\langle f_{\text{v}}(x_j), f_{\text{t}}(y_i) \rangle / \tau)} \right),$$

where τ is the temperature. This loss explicitly uses inner product $\langle \cdot, \cdot \rangle$ to measure the similarity between the encoded image $f_{\text{v}}(x_i)$ and encoded text $f_{\text{t}}(y_j)$, and higher similarities are encouraged if the images and texts are paired. Therefore, such a pretrained model is capable of selecting the tags that describe the image I from a set of customized tags by computing the similarities. Given a set of tags $\{t_i\}_{i=1}^N$, we compute the similarities between the input image I and the tags, and adopt the tags with top-M similarities,

$$\mathcal{T} = \{t_j^*\}_{j=1}^M = \underset{t_i, i=1,\cdots,N}{\arg\text{top-M}} \langle f_{\text{v}}(I), f_{\text{t}}(t_i) \rangle. \tag{1}$$

Short captions. The second model is a caption model $c(\cdot)$. We use it to generate an overall image description $c(I)$.

Local descriptions. The third model is an object detection model. We adopt a well-trained object detector, to provide us with the locations of the possible objects in the format of bounding boxes. The bounding boxes are processed with the non-maximum suppression technique to filter out repetitions. Denote the object proposals as $\{b_j\}_{j=1}^R$ and image regions cropped from corresponding boxes as $\{p_j\}_{j=1}^R$. We first select the indices of the bounding boxes with objects that can be named by our customized tag set,

$$\mathcal{P} = \{\ell_k\}_{k=1}^Q = \{j | \langle f_{\text{v}}(p_j), f_{\text{t}}(t_i) \rangle > \beta, i = 1, \cdots, N, j = 1, \cdots, R\}. \tag{2}$$

Here, β is a threshold certifying whether t_i is aligned with p_j. Given a set of customized attribute $\{a_i\}_{i=1}^V$, each selected proposal ℓ_k from $\mathcal{P}$ is then assigned to an attribute

$$a_{\ell_k}^* = \underset{a_i, i=1,\cdots,V}{\text{argmax}} \langle f_{\text{v}}(p_{\ell_k}), f_{\text{t}}(a_i) \rangle, \tag{3}$$

and the corresponding tags

$$\mathcal{O}_{\ell_k} = \{t_i | \langle f_{\text{v}}(p_{\ell_k}), f_{\text{t}}(t_i) \rangle > \beta, i = 1, \cdots, N\}. \tag{4}$$

In addition to the tags and attributes to the bounding boxes, we also use the caption model $c(\cdot)$ to provide some more descriptive texts $\{c(p_{\ell_k})\}_{k=1}^Q$.

In summary, we collect a tag set $\mathcal{T}$ and a caption $c(I)$ as global descriptions to the image, and a quadruple $(b_{\ell_k}, a_{\ell_k}^*, \mathcal{O}_{\ell_k}, c(p_{\ell_k}))$ as local descriptions for each selected bounding box.

3.2 Candidate Synthesis

We then format the collected visual information into the structured *visual clues*, which can be directly used as the prompt of the language model. Figure 2 shows an example of the visual clues. We observe that the information near the end of the prompt will have a more significant influence on the language model output. As the tags $\mathcal{T}$ are usually more informative and the local extractions are noisier, we input the visual clues with the order of local descriptions, caption, and tags.

To incorporate each local description, a naive way is to inject the coordinates of the bounding boxes directly into the prompt. However, we find the current language models still lack the capability to handle the inference task with numbers, especially in a zero-shot manner. Therefore, we reformat the bounding boxes b_{ℓ_k} into plain language by describing its location and size. Specifically, we adopt rule-based method to divide the locations into 9 classes {"*upper left*", "*upper middle*", "*upper right*", "*left*", "*middle*", "*right*", "*lower left*", "*lower middle*", "*lower right*"}, and divide the sizes into 3 classes {"*large*", "*moderate-sized*", "*small*"}, and incorporate these descriptions into the prompt.

The other visual clues are inputted straightforwardly in the format as showed Figure 2. The prompt is then fed into a large-scale language model to synthesize K candidate paragraphs $\{T_i\}_{i=1}^K$ full of descriptive details.

3.3 Candidate Selection

Finally, we use the vision-language model again, to select the candidate that aligns best with the image,

$$S = \underset{T_i, i=1,\cdots,K}{\text{argmax}} \langle f_{\text{v}}(I), f_{\text{t}}(T_i) \rangle. \tag{5}$$

To further rule out the unrelated concepts in S, we filter the output again in sentence level. This is because large language models sometimes have hallucination issues, i.e., it might generate unrelated

sentences in the paragraphs. For example, a paragraph beginning with "*A couple is hugging on the beach.*" is likely to be followed with "*It's a beautiful day and they're enjoying the sun and each other's company.*" even if there is no visual clue suggesting the weather. Therefore, we split it into sentences $(s_1, s_2, \cdots, s_U)$, and use a threshold γ to remove the sentences with lower similarities,

$$T^* = (s_i | \langle f_{\mathrm{v}}(I), f_{\mathrm{t}}(s_i) \rangle > \gamma, i = 1, \cdots, U). \tag{6}$$

In this way, we obtain the final output T^*.

4 Automatic Evaluation Metric: SPIPE

As indicated by Figure 1, the generated paragraphs of images can be very flexible. This makes the n-gram based metrics, e.g., BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), CIDEr (Vedantam et al., 2015), METEOR (Denkowski and Lavie, 2014), unsuitable for evaluating the generated text. Instead, we focus on the *semantic propositional content*. For example, given an image with content "*A man sitting in front of a blue snowboard*", a good evaluation metric for IPC should evaluate whether each of the semantic propositions is correct, namely, a). a man is sitting; b). a man is in front of a snowboard; c). the snowboard is blue, instead of the exact words used in the text. To do so, SPICE (Anderson et al., 2016) extracts the *scene graphs* (Johnson et al., 2015) from the generated texts and the reference texts, respectively, and computes an F-score between the graphs. SPICE targets image caption tasks, where there are usually multiple good references for each image, and the generation is less flexible. However, IPC tasks usually only have one reference (Krause et al., 2017), which is not enough to evaluate the flexible generation. Therefore, we propose to directly compare the scene graphs extracted from the generated text to human-annotated graphs. Figure 3 shows an example of the generated graph from text and the human-annotated graph for the image.

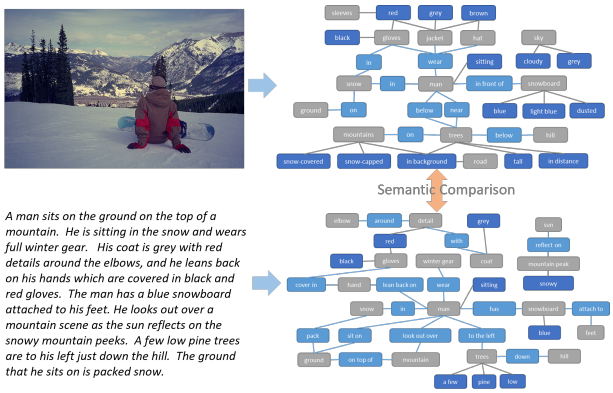


Figure 3: An example of the human-annotated graph and the text extracted graph.

Specifically, a scene graph consists of the objects, the attributes of the objects, and the relationships between the objects. To parse the generated text into a scene graph, we use a two-stage approach following Anderson et al. (2016). First, we use the pretrained dependency parser (Klein and Manning, 2003) to establish the synthetic dependency between the words. Then we map from the dependency trees to scene graphs using a rule-based system (Schuster et al., 2015). Given scene graphs extracted from the text and the human-annotated graphs (Krishna et al., 2017), our metric computes an F-score based on the synonym match[1] Denkowski and Lavie (2014) between the two graphs among the conjunction of three sets of concepts: (object), (object, attribute), and (object, relationship, subject). Paying homage to Anderson et al. (2016), we name our approach **SPIPE**, Semantic Propositional Image Paragraph Evaluation.

5 Empirical Analysis

The basic evaluation of the generated output should include three aspects:

1. Accuracy. Most of the contents appearing in the paragraph should be from the image;
2. Completeness. Most of the contents appearing in the image should be included in the paragraph;
3. Coherence. Paragraphs should be more than concatenating the sentences together.

We evaluate the accuracy and completeness of the generated descriptions using the proposed automatic evaluation metric SPIPE, and do human evaluation to quantify the coherence. We include 500 randomly sampled outputs in `output.html` in the **Supplement Materials** for readers to perform a qualitative study.

[1]Tuples are considered to be matched if their lemmatized word forms are equal or if they are found in the same WordNet (Miller, 1995) synset.

Table 1: Comparison between different methods using SPIPE metric on the test set of the *Stanford dataset* (Krause et al., 2017).

	Name	F-score	Precision	Recall
Models	BLIP-large	7.6	38.0	4.4
	Socratic model	3.2	13.9	1.9
	BEST–general domain	8.8	15.3	6.6
	BEST–VG domain	**10.0**	17.5	7.6
Oracle	BEST with human extracted visual clues	22.9	32.8	19.0
Annotation	Stanford dataset	17.3	27.7	14.0
	Concatenation of VG captions	18.9	40.0	14.1

5.1 Model Specification

Models. We adopt Florence-H (Yuan et al., 2021) as the vision-language model, BLIP-large (Li et al., 2022) finetuned on COCO captions dataset (Chen et al., 2015) as the captioner with its default setting, and one-stage detector as a general object detector. To be more specific about the detector, we first omit the category information from COCO (Chen et al., 2015) datatset and train Dynamic Head (Dai et al., 2021) on the bounding boxes only to formulate a class-agnostic object detector. We then use non-maximum suppression (NMS) to select the top 100 object proposals.

We use GPT-3 (Brown et al., 2020) *Davinci-text-001* model as the language model. To enable more difference in the generated candidates, we adopt temperature as 0.8, as a higher temperature encourages the model to have more creative outputs. We adopt the frequency penalty as 0.5 and the maximum number of tokens as 100.

Customized sets. To construct a general domain tag set, we collect the most frequently searched 400 thousand queries in Bing Search as the tags $\{t_i\}_{i=1}^N$. We adopt the attribute set of the Visual Genome dataset (Krishna et al., 2017) as the attribute set $\{a_i\}_{i=1}^V$.

Parameters. We adopt number of tags $M = 5$, thresholds $\beta = \gamma = 0.2$, and number of candidates $K = 40$. Among $K = 40$ candidates, half of them are generated without caption information while the remaining half are with them. This is because we notice the caption model sometimes cannot output good captions due to too small bounding boxes. We also remove the bounding boxes that are smaller than $1/400$ of the image sizes.

5.2 Automatic Evaluation

In this section, we use SPIPE to benchmark the accuracy and completeness of our framework. We evaluate our framework on the test set of *Stanford dataset* (Krause et al., 2017). The dataset is a subset of Visual Genome (VG) dataset[2] (Krishna et al., 2017), and therefore we can obtain the human-annotated scene graphs from VG as well. We compare the following frameworks.

BLIP (Li et al., 2022). This is the BLIP-large model finetuned on COCO captions dataset.

Socratic model (Zeng et al., 2022). We adopt the image captioning code[3] without alternation.

BEST-general domain. This is our framework with the customized set listed above.

BEST-VG domain. With open-vocabulary capability, our framework can adapt to a specific domain. Here, we replace the customized tag set $\{t_i\}_{i=1}^N$ for the local objects as the object set of VG datasets.

The results are shown in Table 1. Our general domain framework significantly outperforms the BLIP model and the Socratic model. With the domain specified to VG, the performance is further boosted.

Figure 4 shows an example with a image cropped from the Socratic model paper (Zeng et al., 2022) directly. We find that caption generation does not require the complex prompt used in Socratic model. Our framework with only tagging information $\mathcal{T}$ can generate texts with a similar degree of detail. See Appendix D for more discussion.

[2]We remark that the VG caption data is included in the pretraining data of BLIP model. Therefore we do not claim our framework as a *zero-shot* method, despite that it can handle the images in the wild in a zero-shot way.

[3]`https://github.com/google-research/google-research/tree/master/socraticmodels`

BEST: *This image features a group of people standing under a tree with pink blossoms. The people are all dressed in various types of clothing, and some are carrying bags or backpacks. The tree is a sakura tree, and the blossoms are in full bloom. There is a bench in the background, and the ground is covered in fallen petals.*
SM: *People gather under a blossoming cherry tree, enjoying the beauty of natural together.*
BLIP: *A group of people standing around a blue table.*
BEST with only tags: *The image is of people gathered in a park, looking at cherry blossoms. The parks are full of people enjoying the festival and the beautiful view.*

Figure 4: An example cropped from Zeng et al. (2022) paper, with other outputs for comparison.

To evaluate the representation capability of our visual clues, we also compare it to a naive scene graph generation method. We use the vision-language model to assign objects, attributes, and relationships between the objects, using the object set, attribute set, and relationship set of VG. And then we compare the generated scene graph to the human-annotated graph. The F-score is 0.3, with precision 0.8 and recall 0.2. We discuss more on why this does not work in Appendix D.

We also build an oracle model to see the limit of our framework. The oracle model in Table 1 uses the ground truth objects with ground truth attributes to replace the corresponding concepts in the visual clues of our framework. It significantly outperforms the human annotation, either from Stanford dataset or from VG. This reveals the large potential of BEST with the development of object detectors.

5.3 Ablation

We perform an ablation study to see how each of the components contributes to the performance. Especially, we consider replacing the open-vocabulary object detector with YOLO v5 (Jocher, 2020), which is a closed-set detector trained with COCO classes. Table 2 shows the results. The performance of the YOLO v5 alternation is competitive compared to our general domain version. The precision is higher, which may be a consequence that YOLO models tend to recognize fewer objects (Zou et al., 2019). However, it is still inferior to our VG domain model.

Table 2: Ablation on each components. The metrics are F-score (F), Precision (P), and Recall (R).

Name	F	P	R
BEST-VG domain	10.0	17.5	7.6
Extraction with YOLOv5	9.0	19.0	6.3
Remove local information	8.0	14.4	6.0
Remove caption model $c(\cdot)$	8.7	15.0	6.6
Input tags $\mathcal{T}$ only	5.9	10.7	4.4
Smaller language model (*curie*)	8.9	15.9	6.7
Weaker tagger (CLIP *ViT-L/14*)	7.8	16.4	5.6

5.4 Human Evaluation

To further evaluate our framework, we perform human evaluation. We first compare BEST to human annotation. Specifically, we randomly sample 200 descriptions from the test set of the two sources. For each assignment, we present one image and two corresponding descriptions, and ask human evaluators to evaluate on accuracy, completeness, coherence, and ask an additional question "*which of the descriptions is written by human*" for the humanlikeness aspect. They will choose one answer from {*Description 1, Description 2, Cannot determine*}. For each assignment, we hire 5 workers using the Amazon Mechanical Turk platform. More details can be found in Appendix E. As the difference between the long texts can be subtle, we perform two statistical tests to see whether the difference is statistically significant. Please refer to Appendix E.2 for the hypotheses.

Table 3 shows the results. There is *no* significant evidence (p value ≈ 0.5) showing human annotation is better than BEST in terms of completeness and humanlikeness. However, BEST still falls behind in terms of accuracy and coherence. The failure cases are usually because the BEST outputs might contain small mistakes that cannot be easily filtered out, mostly from the hallucination of the language model. We show more examples in Appendix C.

We then compare BEST to BLIP and the Socratic model using similar hypothesis tests. The results show BEST are significantly better than BLIP and Socratic models under most of the metrics (p value < 0.05). Note that here accuracy is defined slightly different than the precision used in Table 1: In human evaluation, providing background information about concepts in the image is not viewed as inaccurate, while in Table 1 it will hurt the precision.

Table 3: Human evaluation. p-value 1 is with the binomial test, and p-value 2 is with Mann–Whitney U test. The blue regions in the voted proportion section represent the proportion that the descriptions from the first source are better than the second, while the orange ones represent the second are better than the first. The 1.2% and 0.6% in the middle of row 4 and 12 represent "*Cannot determine*".

Sources	Criteria	Voted proportion	p-value 1	p-value 2
Anno. / BEST	Accuracy	61.0% 39.0%	2×10^{-3}	9×10^{-6}
	Completeness	51.5% 48.5%	0.38	0.50
	Coherence	59.7% 40.3%	5×10^{-3}	8×10^{-4}
	Humanlikeness	54.7% 1.2% 44.2%	0.10	0.50
BEST / BLIP	Accuracy	57.2% 42.8%	0.03	3×10^{-3}
	Completeness	73.6% 26.4%	2×10^{-11}	2×10^{-28}
	Coherence	56.2% 43.8%	0.05	5×10^{-3}
	Humanlikeness	58.3% 41.7%	9×10^{-3}	5×10^{-4}
BEST / Socratic	Accuracy	59.4% 40.6%	6×10^{-3}	7×10^{-5}
	Completeness	71.3% 28.7%	2×10^{-9}	1×10^{-24}
	Coherence	68.7% 31.3%	3×10^{-7}	2×10^{-12}
	Humanlikeness	53.4% 0.6% 46.0%	0.18	0.35

6 Variants and Real-world Applications

The proposed framework opens up many creative real-world applications. For example, People with vision deficiencies may not be able to view images easily. BEST can help convert it into precise and comprehensive descriptions for general domain images.

Another example is the closed-loop training of the large models. The large-scale vision-language model and language model used in BEST are trained on tremendous amounts of data, and thus can memorize knowledge beyond human capability. We can use it to automatically annotate data, which is easy to scale up. Furthermore, it can incorporate commonsense knowledge into the text naturally. For example, in the second example of Figure 1, the text contains "*This image captures the everyday life of Cubans, with their traditional horse-drawn carts still in use.*" This is because our tags contain "*Cuba*" and "*buggy*", and the language model knows traditional horse-drawn carts are still in use in Cuba. We finetune a BLIP-large model on our BEST generated data. The training images are similar to Stanford dataset (Krause et al., 2017), which is around 15 thousands. After finetuning, the F-score improves more than 50%.

Table 4: Finetune BLIP-large on different data.

Name	F	P	R
No finetune	7.6	38.0	4.4
With Socratic generated data	3.9	17.5	2.3
With BEST generated data	11.6	23.1	8.2

With small modifications, the proposed framework enables us to free human labor for even more applications. To list a few examples,

Visual storytelling. As shown in Figure 5 (a), the framework can generate charming stories based on the input image. To do so, we simply change the end of the prompt to be "*Tell me a creative story:*".

Automatic ads generation: As shown in Figure 5 (b), with the framework, the merchants only need to upload an image, and make small modifications to the generated advertisement as wanted. As there is usually one product in an image, we adopt the number of input tags $M = 1$. We also change the end of the prompt to be "*Write a product description to sell in eBay or Amazon marketplace to get lots of engagement:*".

Social media post. As shown in Figure 5 (c), the framework can be a social media bot, which may alleviate the workload of internet celebrities. We change the end to be "*Social media post:*".

Background generation. As shown in Figure 5 (d), the framework can also be used to provide some background knowledge. To do so, we change the end of the prompt to be "*Textbook text:*".

(a)		*As the Hogwarts Express made its way through the Scottish countryside, a young wizard named Harry peered out the window at the stunning Glenfinnan Viaduct. He had always dreamed of traveling on the venerable steam train, and now he was finally doing it! The locomotive chugged along, its great iron wheels clanking over the rails, as passengers in every car gazed in wonder at the beautiful scenery.*
(b)		*Looking for a stylish and practical coat to keep you warm this winter? Look no further than our long parka! Made from high quality beige fabric, this coat is sure to keep you warm and comfortable all winter long. It features a stylish and flattering fit, as well as a warm and cozy hood, making it perfect for cold days out.*
(c)		*What do you think of this Chevy Impala Lowrider? I spotted it at the Sema Car Show and it's definitely a head-turner!*
(d)		*A tench is a fish that lives in fresh water. It is a common fish that can be found in many lakes and rivers. The tench is a large fish, and can grow to be quite large. It has a greenish-brown body, and is covered in scales. The tench feeds on insects, plants, and small fish.*
(e)		*The image is of a cover of The Economist magazine. The headline is "March of the machines". The cover features a drawing of the Maschinenmensch, or machine man, in the center. The background is art deco in style. This image represents the rise of robots and artificial intelligence, as they become increasingly important in our society.*

Figure 5: Examples for real-world applications with variants of the proposed framework.

Applications with scene texts. As shown in Figure 5 (e), when the input image has rich scene text, we plug in an OCR (Optical Character Recognition) model (Farley, 2022), and insert the OCR output into the prompts with prefix "*This image contains text:*" before the captions.

7 Limitations and Further Improvements

Prompt tuning. As suggested in Brown et al. (2020), language models can infer better when they are shown examples in the prompt. In our experiments, however, this results in model directly copying sentence pieces from example paragraphs, introducing unnecessary noise. We suspect this prompt tuning approach may work better if the input examples are similar to the generated one. This may be a promising direction as we can better control writing style.

Visual clues as a scene graph. Our visual clue extraction process is motivated by the fact that an image can be comprehensively represented by a scene graph (Johnson et al., 2015). As mentioned in Section 4, a scene graph contains objects, attributes of objects, and the relationship between objects. In BEST, however, we do not include the relationships, as we observe in our initial study that the current vision-language models, although powerful, are not good at inferring relationships (echoing findings from Thrush et al. (2022)). Yet, relationships among the objects are important components of an image. This can be plugged into our framework if better vision-language models are developed.

A well-trained filter model. We find that the current filtering strategy (6) is not immune to certain types of mistakes. As also mentioned in Thrush et al. (2022), the vision-language model cannot accurately associate attributes to their corresponding objects. For example, in the second image of Figure 1, if there is a sentence "*The man wears a black shirt.*", it will lead to a high image-text relevance score, since there is a man, a shirt, and dark bush in the image. To handle this issue, we crop the image into local regions and pair each region with an attribute. Still, if it is the language model who hallucinates new attributes and the attributes happen to be in the image, these captions cannot be filtered out. We suspect an adversarially trained filter is needed to perform the task.

Broader Societal Impact. BEST inherits the risks of large vision and language models. BEST can potentially output offensive language and propagate social biases and stereotypes. For real applications, we can use rule-based methods or train a specific filter to filter out the offensive text. This is an area that we plan to explore to gain more insights further.

References

ALAYRAC, J.-B., DONAHUE, J., LUC, P., MIECH, A., BARR, I., HASSON, Y., LENC, K., MENSCH, A., MILLICAN, K., REYNOLDS, M. ET AL. (2022). Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*.

ANDERSON, P., FERNANDO, B., JOHNSON, M. and GOULD, S. (2016). Spice: Semantic propositional image caption evaluation. In *European conference on computer vision*. Springer.

BROWN, T., MANN, B., RYDER, N., SUBBIAH, M., KAPLAN, J. D., DHARIWAL, P., NEELAKANTAN, A., SHYAM, P., SASTRY, G., ASKELL, A. ET AL. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, **33** 1877–1901.

CHATTERJEE, M. and SCHWING, A. G. (2018). Diverse and coherent paragraph generation from images. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

CHEN, J., GUO, H., YI, K., LI, B. and ELHOSEINY, M. (2022). Visualgpt: Data-efficient adaptation of pretrained language models for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

CHEN, X., FANG, H., LIN, T.-Y., VEDANTAM, R., GUPTA, S., DOLLÁR, P. and ZITNICK, C. L. (2015). Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*.

CHO, J., LEI, J., TAN, H. and BANSAL, M. (2021). Unifying vision-and-language tasks via text generation. In *International Conference on Machine Learning*. PMLR.

DAI, B., FIDLER, S., URTASUN, R. and LIN, D. (2017). Towards diverse and natural image descriptions via a conditional gan. In *Proceedings of the IEEE international conference on computer vision*.

DAI, X., CHEN, Y., XIAO, B., CHEN, D., LIU, M., YUAN, L. and ZHANG, L. (2021). Dynamic head: Unifying object detection heads with attentions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

DENKOWSKI, M. and LAVIE, A. (2014). Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the ninth workshop on statistical machine translation*.

FARLEY, P. (2022). What is optical character recognition? - azure cognitive services. `https://docs.microsoft.com/en-us/azure/cognitive-services/computer-vision/overview-ocr`

FEI-FEI, L. and KRISHNA, R. (2022). Searching for computer vision north stars. *Daedalus*, **151** 85–99.

GOYAL, Y., KHOT, T., SUMMERS-STAY, D., BATRA, D. and PARIKH, D. (2017). Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

GUO, D., LU, R., CHEN, B., ZENG, Z. and ZHOU, M. (2021). Matching visual features to hierarchical semantic topics for image paragraph captioning. *arXiv preprint arXiv:2105.04143*.

HE, P., LIU, X., GAO, J. and CHEN, W. (2020). Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

HUANG, T.-H., FERRARO, F., MOSTAFAZADEH, N., MISRA, I., AGRAWAL, A., DEVLIN, J., GIRSHICK, R., HE, X., KOHLI, P., BATRA, D. ET AL. (2016). Visual storytelling. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

HUDSON, D. A. and MANNING, C. D. (2019). Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.

JIA, C., YANG, Y., XIA, Y., CHEN, Y.-T., PAREKH, Z., PHAM, H., LE, Q., SUNG, Y.-H., LI, Z. and DUERIG, T. (2021). Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*. PMLR.

JOCHER, G. (2020). ultralytics/yolov5: v3.1 - Bug Fixes and Performance Improvements. `https://github.com/ultralytics/yolov5`. `https://doi.org/10.5281/zenodo.4154370`

JOHNSON, J., KRISHNA, R., STARK, M., LI, L.-J., SHAMMA, D., BERNSTEIN, M. and FEI-FEI, L. (2015). Image retrieval using scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

KLEIN, D. and MANNING, C. D. (2003). Accurate unlexicalized parsing. In *Proceedings of the 41st annual meeting of the association for computational linguistics*.

KRAUSE, J., JOHNSON, J., KRISHNA, R. and FEI-FEI, L. (2017). A hierarchical approach for generating descriptive image paragraphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

KRISHNA, R., ZHU, Y., GROTH, O., JOHNSON, J., HATA, K., KRAVITZ, J., CHEN, S., KALANTIDIS, Y., LI, L.-J., SHAMMA, D. A. ET AL. (2017). Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, **123** 32–73.

LI, J., LI, D., XIONG, C. and HOI, S. (2022). Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. *arXiv preprint arXiv:2201.12086*.

LI, Y., SU, H., SHEN, X., LI, W., CAO, Z. and NIU, S. (2017). Dailydialog: A manually labelled multi-turn dialogue dataset. *arXiv preprint arXiv:1710.03957*.

LIANG, X., HU, Z., ZHANG, H., GAN, C. and XING, E. P. (2017). Recurrent topic-transition gan for visual paragraph generation. In *Proceedings of the IEEE international conference on computer vision*.

LIN, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*.

LIU, B., FU, J., KATO, M. P. and YOSHIKAWA, M. (2018). Beyond narrative description: Generating poetry from images by multi-adversarial training. In *Proceedings of the 26th ACM international conference on Multimedia*.

LUO, Y., HUANG, Z., ZHANG, Z., WANG, Z., LI, J. and YANG, Y. (2019). Curiosity-driven reinforcement learning for diverse visual paragraph generation. In *Proceedings of the 27th ACM International Conference on Multimedia*.

MAO, Y., ZHOU, C., WANG, X. and LI, R. (2018). Show and tell more: Topic-oriented multi-sentence image captioning. In *IJCAI*.

MARINO, K., RASTEGARI, M., FARHADI, A. and MOTTAGHI, R. (2019). Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*.

MARR, D. (2010). *Vision: A computational investigation into the human representation and processing of visual information*. MIT press.

MAYNEZ, J., NARAYAN, S., BOHNET, B. and MCDONALD, R. (2020). On faithfulness and factuality in abstractive summarization. *arXiv preprint arXiv:2005.00661*.

MELAS-KYRIAZI, L., RUSH, A. M. and HAN, G. (2018). Training for diversity in image paragraph captioning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

MILLER, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, **38** 39–41.

PAPINENI, K., ROUKOS, S., WARD, T. and ZHU, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*.

RADFORD, A., KIM, J. W., HALLACY, C., RAMESH, A., GOH, G., AGARWAL, S., SASTRY, G., ASKELL, A., MISHKIN, P., CLARK, J. ET AL. (2021). Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*. PMLR.

RAJPURKAR, P., ZHANG, J., LOPYREV, K. and LIANG, P. (2016). Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

SALVADOR, A., DROZDZAL, M., GIRÓ-I NIETO, X. and ROMERO, A. (2019). Inverse cooking: Recipe generation from food images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

SCHUSTER, S., KRISHNA, R., CHANG, A., FEI-FEI, L. and MANNING, C. D. (2015). Generating semantically precise scene graphs from textual descriptions for improved image retrieval. In *Proceedings of the fourth workshop on vision and language*.

SHI, Y., LIU, Y., FENG, F., LI, R., MA, Z. and WANG, X. (2021). S2td: A tree-structured decoder for image paragraph captioning. In *ACM Multimedia Asia*. 1–7.

SINGH, A., HU, R., GOSWAMI, V., COUAIRON, G., GALUBA, W., ROHRBACH, M. and KIELA, D. (2021). Flava: A foundational language and vision alignment model. *arXiv preprint arXiv:2112.04482*.

SU, Y., LAN, T., LIU, Y., LIU, F., YOGATAMA, D., WANG, Y., KONG, L. and COLLIER, N. (2022). Language models can see: Plugging visual controls in text generation. *arXiv preprint arXiv:2205.02655*.

THRUSH, T., JIANG, R., BARTOLO, M., SINGH, A., WILLIAMS, A., KIELA, D. and ROSS, C. (2022). Winoground: Probing vision and language models for visio-linguistic compositionality. In *CVPR*.

VEDANTAM, R., LAWRENCE ZITNICK, C. and PARIKH, D. (2015). Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

WANG, J., PAN, Y., YAO, T., TANG, J. and MEI, T. (2019). Convolutional auto-encoding of sentence topics for image paragraph generation. *arXiv preprint arXiv:1908.00249*.

WANG, P., YANG, A., MEN, R., LIN, J., BAI, S., LI, Z., MA, J., ZHOU, C., ZHOU, J. and YANG, H. (2022). Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. *arXiv preprint arXiv:2202.03052*.

WANG, Z., YU, J., YU, A. W., DAI, Z., TSVETKOV, Y. and CAO, Y. (2021). Simvlm: Simple visual language model pretraining with weak supervision. *arXiv preprint arXiv:2108.10904*.

XU, C., LI, Y., LI, C., AO, X., YANG, M. and TIAN, J. (2020). Interactive key-value memory-augmented attention for image paragraph captioning. In *Proceedings of the 28th International Conference on Computational Linguistics*.

YANG, Z., GAN, Z., WANG, J., HU, X., LU, Y., LIU, Z. and WANG, L. (2021). An empirical study of gpt-3 for few-shot knowledge-based vqa. *arXiv preprint arXiv:2109.05014*.

YU, J., WANG, Z., VASUDEVAN, V., YEUNG, L., SEYEDHOSSEINI, M. and WU, Y. (2022). Coca: Contrastive captioners are image-text foundation models.

YUAN, L., CHEN, D., CHEN, Y.-L., CODELLA, N., DAI, X., GAO, J., HU, H., HUANG, X., LI, B., LI, C. ET AL. (2021). Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*.

ZENG, A., WONG, A., WELKER, S., CHOROMANSKI, K., TOMBARI, F., PUROHIT, A., RYOO, M., SINDHWANI, V., LEE, J., VANHOUCKE, V. ET AL. (2022). Socratic models: Composing zero-shot multimodal reasoning with language. *arXiv preprint arXiv:2204.00598*.

ZHOU, C., NEUBIG, G., GU, J., DIAB, M., GUZMAN, P., ZETTLEMOYER, L. and GHAZVININEJAD, M. (2020a). Detecting hallucinated content in conditional neural sequence generation. *arXiv preprint arXiv:2011.02593*.

ZHOU, L., PALANGI, H., ZHANG, L., HU, H., CORSO, J. and GAO, J. (2020b). Unified vision-language pre-training for image captioning and vqa. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34.

ZHU, X., ZHU, J., LI, H., WU, X., WANG, X., LI, H., WANG, X. and DAI, J. (2021). Uni-perceiver: Pre-training unified architecture for generic perception for zero-shot and few-shot tasks. *arXiv preprint arXiv:2112.01522*.

ZOU, Z., SHI, Z., GUO, Y. and YE, J. (2019). Object detection in 20 years: A survey. `https://arxiv.org/abs/1905.05055`

References

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [Yes]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] As much as possible. Some of the models and data are proprietary.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] The proposed method does not involve training.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No] The proposed method does not involve training. The training used in application section is not significant. The overall usage of computing resources is not significant.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [No]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [Yes]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [Yes]

A More Examples of Generated Paragraphs and Visual Clues

BEST: *This image features a doughnut and a cup of coffee on a table. The doughnut is chocolate and glazed and the coffee is black. There is also a daypack visible in the image, as well as some straps that may be used to hold the backpack in place.*
SM: *Home office sink with book and coffee.*
BLIP: *A doughnut and a cup of coffee on a table.*
BEST with only tags: *This image is of a coffee and donut in a daypack. The donut is labelled the "donut of shame" as it is a bit battered and not as fresh looking as the coffee.*
Human Annotation: *A table has some items on it. The table is made of wood. There is a donut on a plate. The donut is made of chocolate. The plate is white. The plate is round. There is a cup of coffee on the table. The cup is white with a blue rim. The is a backpack by the cup and plate. The backpack is brown and white. There is a book on the backpack. The book is a paperback. The table is by a wall. The table has a chair by it.*

BEST: *This image shows a train traveling over a bridge over a street in Las Vegas. The train is a monorail, and is traveling next to tall buildings and the Stratosphere casino. There is a blinking red light on the train, and a blinking green light on the bridge.*
SM: *A photo of an indoor parking garage.*
BLIP: *A train traveling over a bridge over a street.*
BEST with only tags: *This image is of the Las Vegas Monorail. It is a high-speed, driverless railway system that runs on a single beam. The Monorail connects the Las Vegas Strip to the Las Vegas Convention Center and other major hotels, including the Stratosphere, MGM Grand, and Sahara.*
Human Annotation: *There are two skyscrapers. There is also a train going around a track above the road. The traffic lights are lit to red and green. There are palm trees to the left.*

BEST: *This image captures a surfer performing a cutback maneuver at the Superbank surf competition. The surfer is positioned in the middle of the frame and is relatively small in comparison to the surrounding waves. The waves are large and crashing, providing an impressive backdrop for the surfing action.*
SM: *A person is surfing on a wave at a medina.*
BLIP: *A man riding a wave on top of a surfboard.*
BEST with only tags: *This image captures a surfer cutting back in a competition at the Superbank. The surfer is Bruna Surfistinha and the competition is the Chris Ward Surf Invitational. The waves are huge and the surfer is doing a great job of carving through them.*
Human Annotation: *A man is in a black wetsuit. The man is riding a white surfboard. There is a large wave in the water the man is surfing on.*

Figure 6: Examples of generated image paragraph.

	Prompt: *Objects in this image:* *a donut on a plate next to a cup of coffee. coffee and donut, is at middle of the image and is large in the image. Attribute: donut* *a donut on a plate next to a cup of coffee. coffee and donut, is at middle of the image and is large in the image. Attribute: doughnut* *a donut on a plate next to a cup of coffee. coffee and donut, is at lower left of the image and is moderate-sized in the image. Attribute: chocolate glazed* *Caption:* *a doughnut and a cup of coffee on a table* *Tags:* *This image is about coffee and donuts, daypack, the donut of shame, dohnut, randys donuts* *Describe this image in detail:*
	Prompt: *Objects in this image:* *a train traveling through a city next to tall buildings. monorail, is at middle of the image and is moderate-sized in the image. Attribute: monorail* *a black and white photo of a clock. golden sword, is at lower middle of the image and is small in the image. Attribute: gold-tipped* *Caption:* *a train traveling over a bridge over a street* *Tags:* *This image is about las vegas monorail, las vegas metro, monorail, high roller stratosphere, high roller* *Describe this image in detail:*
	Prompt: *Objects in this image:* *a dog is sitting under an umbrella in the snow. leg on surfboard, is at middle of the image and is small in the image. Attribute: wake boarding* *a man riding a wave on top of a surfboard. black surfers, is at middle of the image and is small in the image. Attribute: concentrated surfing* *a man riding a wave on top of a surfboard. black surfers, is at middle of the image and is moderate-sized in the image. Attribute: concentrated surfing* *a man holding a surfboard on top of a wave. on surfboard, is at middle of the image and is small in the image. Attribute: surfboarding* *a man riding a wave on top of a surfboard. man on surfboard, is at middle of the image and is small in the image. Attribute: concentrated surfing* *Caption:* *a man riding a wave on top of a surfboard* *Tags:* *This image is about bruna surfistinha, chris ward surfer, superbank, cutback, surf competition* *Describe this image in detail:*

Figure 7: Examples of generated prompts of images in Figure 6. We can see the prompts may contain minor errors. But language model tends to ignore the concepts that are too irrelevant to the scene.

BEST: *This image shows a kitchen with white appliances and wooden cabinets. The stove is in the middle of the image and the refrigerator is on the right. There is also a sink on the left.*
SM: *A photo of a kitchen, with a stove, kitchen range, and range hood.*
BLIP: *A kitchen with white appliances and wooden cabinets.*
BEST with only tags: *This image is of a kitchen that has been remodeled. It is a galley style kitchen with an oak cabinet. There is a sink, stove, and refrigerator in the kitchen. The countertops are granite and the flooring is wood.*
Human Annotation: *A kitchen that has a brown wooden floor,wooden brown cabinets with a white stove and white refrigerator. the sink is empty and clean along with the counter which is white in color with a window above the sink which has brown blinds.*

BEST: *This image is of a group of people who appear to be waiting to enter a running room. The people are standing on a sidewalk and most are wearing athletic clothing. In the background, there is a yellow school bus parked on the side of the road.*
SM: *Runners in a fastfood restaurant.*
BLIP: *A group of people standing outside of a running room.*
BEST with only tags: *This image is of a marathon coach and two runners. The marathon coach is giving instructions to the runners and they are getting ready to run. There is a banner in the background that says "Run Around".*
Human Annotation: *A store named the Running Room is to the left of the road. A yellow school bus is parked next to the store. There are people in running clothes lined up the side of the building. Large windows surround the first level of the store. Some bikes are leaned up against the front of the building. There are clouds in the sky and electrical wires are on poles above the road.*

BEST: *This image is a red fire hydrant that is sitting on the side of the road. The fire hydrant has a black hat on top of it and it is made out of metal. The hydrant is red and the hat is black. The hat might be used to keep the hydrant from freezing in the winter.*
SM: *The fire hydrant stands out against the snowy background.*
BLIP: *A red fire hydrant sitting on the side of a road.*
BEST with only tags: *This image shows a firefighter in front of a fire hydrant. The firefighter is wearing a fire hat and has a woodford hydrant. The freeze proof hydrant is behind the woodford hydrant.*
Human Annotation: *A red fire hydrant with a white top and a black knit cap on top of it sits on a sidewalk at the side of a parking lot. The hydrant is casting a shadow on the sidewalk where leaves are scattered around it. There is a manhole cover a bit down from the hydrant. Behind the manhole cover is a small area with small green shrubs and mostly covered by leaves. At the end of the sidewalk are rows of cars with a store in the far back. A couple of people are near their cars in the parking lot. At the end of one row of cars is a wooden fence with a brick building behind it.*

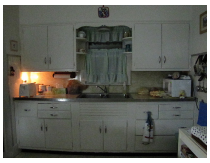
Figure 8: Examples of generated image paragraph.

BEST: *This image shows a group of people gathered around luggage carts at an airport. There is a mix of different types of luggage, including suitcases, backpacks, and duffel bags. There are also a few people in the image who are not carrying any luggage.*
SM: *People waiting in an airport terminal.*
BLIP: *A group of people standing around carts with luggage.*
BEST with only tags: *The image shows several people waiting in a line at an airport baggage claim. The airport is likely in China, as both Pudong and Shenzhen airports are located there. Beijing airport is also located in China. All three airports are major international hubs, with flights arriving from all over the world.*
Human Annotation: *There are people standing in a room. There are signs on the walls behind them. There are three scooters that have luggage on top of them in front of the people.*

BEST: *This black and white photo shows an old train station that is now abandoned. The train itself looks quite old and is in the middle of the image, taking up most of the space. There are other objects in the image such as a funicular and cog railroad, but the old train is the main focus.*
SM: *Abandoned railcar at a forgotten railway platform.*
BLIP: *A black and white photo of an old train.*
BEST with only tags: *This image features an abandoned train station. The station is located in the canfranc international railway station, and is part of the durango and silverton narrow gauge railroad. The funiculaire can also be seen in the background, as well as the cog railroad.*
Human Annotation: *A worn wooden building is on the side of a set of train tracks. The building has many windows with screens over the windows. Many of the screens are torn or missing. The paint on the building is severely thinned and peeling. A small foliage bush is at the front of the building. Tall mountains are in the far distance. A large metal structure is suspended in the air above the building and the train tracks.*

BEST: *This image features a group of people, all of whom are standing on a tennis court. The people in the image are all dressed in appropriate tennis gear, and each is holding a racquet of some kind. In the background, there is a fence and beyond that, some trees.*
SM: *A group of people enjoying a game of tennis.*
BLIP: *A group of people standing on top of a tennis court.*
BEST with only tags: *This image features the Bollettieri Tennis Academy, which is a well-known junior tennis training center. The academy is situated in Bradenton, Florida and has produced many top tennis players, such as Andre Agassi, Monica Seles, and Jim Courier. The academy has a kids' court that is visible in this image, as well as a grand slam tennis court in the background.*
Human Annotation: *A group of kids and teenagers are posing for a picture on a tennis court. An adult instructor is standing at the far right of the kids. A net is behind them and the court ground is dark green with white painted lines. There are tennis balls on the ground in front of the kneeling kids. A chain link fence is behind the court and a group of trees is in the background. Everyone is wearing shorts, t-shirts and sneakers.*

Figure 9: More examples of generated image paragraph.



BEST: *This image is of a kitchen with white cabinets and a sink. The kitchen has a retro feel to it, with the white cabinets and old-fashioned sink. There is a small amount of light in the image, coming from what may be a window in the upper left corner of the photograph. The rest of the image is fairly dark, with only a small amount of light illuminating the countertop and floor.*
SM: *A person in a kitchen, cooking food on a stove.*
BLIP: *A kitchen with white cabinets and a sink.*
BEST with only tags: *This image is of a retro kitchen in the night. It is a small, first kitchen with white cabinets and appliances. There is a light over the sink and a window with a curtain.*
Human Annotation: *White cabinets are in a kitchen. There is a window above the sink with white curtains. There is a white microwave on the counter.*

BEST: *This is a black and white photo of a toilet. The toilet is white and it looks like it is sitting outside of a building next to a bush.*
SM: *A photo of a toilet.*
BLIP: *A toilet sitting outside of a building next to a bush.*
BEST with only tags: *This image is of a toilet throne. It is a Universal Rundle toilet with a Portland Loo in the background. The American Standard toilet is to the right of the Portland Loo. The Toto Drake is to the right of the American Standard toilet.*
Human Annotation: *A white toilet with no lid is sitting outside of a brick building on the concrete ground. The toilet is white and is placed against the brick wall of a building. It is in a corner. A tree is on one side of the toilet. A window is in the wall just above the toilet. There are cracks in the concrete on the ground. A door with a doormat is beside the tree.*

BEST: *This image captures the traffic on the Subic-Clark-Tarlac Expressway in the Klang Valley region of Malaysia. The expressway is a 4-lane, controlled-access highway that connects the two major metropolitan areas of the country. The image shows several cars and trucks travelling on the highway, as well as an ambulance and a police motorcycle.*
SM: *A divided highway with many cars and people.*
BLIP: *A busy city street filled with lots of traffic.*
BEST with only tags: *This image is of the Tarlac-Pangasinan-La Union Expressway (TPLEX), a toll expressway in the Philippines. The expressway is seen here in the Klang Valley area of Selangor, Malaysia, and is crowded with cars and carreolas (two-wheeled carts).*
Human Annotation: *This is a photo of a busy roadway. Many different vehicles can be seen on the road. A long narrow grass strip is separating the different sides of the roads. Tall lamp poles are lining the roadway. A large blue and white sign is standing on the left side of the roadway. Tall green trees and a long steep hill can be seen in the background of the photo.*

Figure 10: More examples of generated image paragraph.

B Other Automatic Evaluation Metrics

In Table 5 we compare different zero-shot methods using the metrics inherit from image captioning tasks. We can see that our framework BEST still performs the best.

Table 5: Comparison between different methods using captioning metrics on the test set of the *Stanford dataset* (Krause et al., 2017).

Name	Bleu-1	Bleu-2	Bleu-3	Bleu-4	METEOR	ROUGE-L	CIDEr
BLIP-large	0.4	0.2	0.1	0.1	5.4	15.1	0.6
Socratic model	0.6	0.2	0.1	0.0	3.0	9.3	0.5
BEST–general domain	30.9	15.9	8.1	4.2	11.9	23.7	11.3
BEST–VG domain	31.8	16.5	8.5	4.4	12.5	24.3	11.9

C Negative Examples

Figure 11 shows a few failure cases of BEST generation, which may highlight the direction for further improvement. The first example demonstrates two issues. The first one is the vision-language model cannot accurately match attributes to the objects, as we discussed in Section 7. The second issue is the language model sometimes repeat the input sentences. This issue will be alleviated if we adopt better language models, e.g., replacing *Davinci-text-001* by its upgraded version *Davinci-text-001*.

The second example suggests the Optical Character Recognition (OCR) capability of the vision-language model is not well-trained. So if for datasets with images involving text photos, we should explicitly plug in an OCR model.

The third example shows the current VL model cannot differentiate the relationship between the objects. This may be because the training loss, e.g., contrastive loss, does not require the model to differentiate the complex relationships – only recognizing the objects is enough.

In this image, we see a young boy ***riding a bicycle with a backpack on****. To the right, we see another boy skateboarding. In the background, we see people walking and running.* ***This image is about streetsurfing, street surfing, street sesh, skatejoring, and longboarding.***

This image captures a homeless man taking a much-needed nap in the middle of the day. The bright red fire hydrant is a stark contrast to his muted red clothing, and stands out in the image. The man is lying on his back, with his arms and legs sprawled out, and appears to be asleep. He is surrounded by empty bottles and cans, as well as a few pieces of trash. ***There is an inscription on the wall behind him that reads "Siesta".***

This image features Andy Roddick serving during the US Open Tennis Championships in 2009. Roddick is a former world number one tennis player and is known for his powerful serve. In the image, he is shown hitting a forehand shot. ***The ball is seen speeding towards the opponent's court, and the racquet appears to be at the top of its arc.*** *The umpire can be seen in the background, and the top of the racquet is visible in the foreground.*

Figure 11: Negative examples of generated image paragraph.

D More Details on Baselines

Scene Graph Generation Baseline As mentioned earlier, to evaluate the representation capability of our visual clues, we also compare it to a naive scene graph generation method. We use the vision-language model to assign objects, attributes, and relationships between the objects, using the object set, attribute set, and relationship set of VG. Algorithm 1 shows the complete scene graph generation algorithm. And then we compare the generated scene graph to the human-annotated graph.

Algorithm 1 Algorithm for the Scene Graph Generation Baseline

```
Select the bounding boxes in the same way as BEST
Associate each of the bounding boxes with one object name and one attribute
for each pair of bounding boxes do
    Compute the minimum bounding box covering the union of the two bounding boxes
    Assign a relationship to the minimum bounding box using the open-vocabulary tagger
    Choose from (object 1, relationship, object 2) and (object 2, relationship, object 1) with tagger
    Add the chosen relationship with its object and subject to the output
end for
```

The F-score is 0.3, with precision 0.8 and recall 0.2. We show an example of the output graphs in Figure 12. The particularly bad performance is majorly due to the sets from VG are noisy, so the synonym match between nodes yields very low score.

Why Socratic model cannot perform well? The major issue of Socratic model is that its prompt contains inaccurate or useless information, and is not informative enough. Its prompt is

"*I am an intelligent image captioning bot. This image is a {img_type}. There {num_people}. I think this photo was taken at a {place1}, {place2}, or {place3}. I think there might be a {object1}, {object2}, {object3},... in this {img_type}. A creative caption I can generate to describe this image is:*"

There are three issues with this prompt:

1. Useless information. We find a much shorter prompt, *e.g.*, "*This is a {img_type} taken at {place1} containing {object1}, {object2}, {object3}... Generate a caption:*" has comparable or even better performance as the one above.
2. Inaccurate information. As we discussed in Section 7, VL models trained with contrastive loss cannot differentiate some details in the sentence. We observe that the *num_people* obtained by Socratic model is basically a random guess.
3. Lacking information. The *img_type* only contains four categories, photo, cartoon, sketch, painting. The object list is a list of common objects, not as comprehensive as ours. The place information usually redundant given the object information.

In contrast, our prompt is much more informative and well-constructed.

Comparison with traditional IPC methods: It is unfair to compare methods with complete training data and training pipeline to our nearly zero-shot framework. Not only because the training cost is not comparable, but also for the trained models, the language style is aligned with the test data. For example, human annotators without domain specific knowledge tend to say, "The man is riding a white surfboard", while our framework outputs "This image captures a surfer performing a cutback maneuver". The difference in language styles significantly lowers the scores of our framework. As a matter of fact, our BEST scores in Table 5 are not as good as IPC methods. For example, the scores for paper Melas-Kyriazi et al. (2018) are BLEU-4 10.58, METEOR 17.86, CIDEr 30.63.

As a matter of fact, our framework and traditional IPC methods are essentially different pipelines. Our framework has unique advantages that traditional IPC methods do not have:

1. **Easier implementation.** Our framework involves no training and no data for training. Therefore, the cost of building such a pipeline is much lower than traditional frameworks.

   Traditionally to build a machine learning application, researchers need to 1). Collect a dataset; 2). Select a training framework; 3). Train the model with repeated hyper-parameter tuning. And then we can have a model that is specifically designed for such an application.

Model Generation
Objects: *plastic cups, woman eats pizza, spinach on pizza, enjoying a sandwich, eating contest, plastic cup, ribs, case is brown, griddle, foot pedal, parsley on plate*
Attributes:
(object: *plastic cups*, attr: *solo cup*),
(object: *woman eats pizza*, attr: *eating italian*),
(object: *spinach on pizza*, attr: *looking at pizza*),
(object: *enjoying a sandwich*, attr: *getting her lunch*),
(object: *plastic cup*, attr: *solo cup*),
(object: *ribs*, attr: *bacon*) ...
Relationships:
(object: *woman eats pizza*, subject: *plastic cups*, rel: *eating pizza*),
(object: *spinach on pizza*, subject: *plastic cups*, rel: *eating pizza*),
(object: *plastic cups*, subject: *enjoying a sandwich*, rel: *eating pizza*),
(object: *case is brown*, subject: *plastic cups*, rel: *cup on*),
(object: *griddle*, subject: *plastic cups*, rel: *table has items*)...

Human Annotation
Objects: *"bracelet", "glasses", "face", "cup", "phone", "table", "pizza", "hand", "tray", "neon sign", "wall", "napkin holder", "picture", "paper plate", "women", "pizza", "cups", "paper napkins", "woman", "tanktop", "woman", "blue shirt", "napkin", "plate", "some food", "advertisement", "cell phone", "short hair", "white napkins", "toppings", "ale", "plate and napkin", "dark hair", "wrist"*
Attributes:
(object: *bracelet*, attr: *blue*),
(object: *glasses*, attr: *black*),
(object: *cup*, attr: *plastic*),
(object: *pizza*, attr: *large*),
(object: *cell phone*, attr: *black*) ...
Relationships:
(object: *face*, subject: *glasses*, rel: *on*),
(object: *table*, subject: *phone*, rel: *on*),
(object: *hand*, subject: *pizza*, rel: *in*),
(object: *table*, subject: *tray*, rel: *on*),
(object: *wall*, subject: *picture*, rel: *on*)...

Figure 12: One examples of the generated scene graphs.

If the business need is adjusted, the process needs to be repeated to accommodate the shifted application domain.

In contrast, what we propose is a light-weighted solution: The pretrained models are either available via API service (e.g., GPT3, captioner), or will be available soon (e.g., Florence tagger). To build our framework, researchers only need to plug in the APIs. As suggested by Figure 5, it can handle various scenarios with only minor modification.

One may argue that some applications require fast inference, which our large pretrained model cannot handle. Yet, using the synthetic data generated by BEST (around 20K), we have successfully replaced GPT-3 by a DeBERTa-large model (He et al., 2020) with comparable performance. Generating data using BEST is faster and more stable than recruiting human laborers.

2. **Versatile to different application scenarios**, as suggested by Figure 5.

3. **Robust to domain shift.** As suggested Figure 1 of Melas-Kyriazi et al. (2018), for a black and white image which the model is not trained on, the inference of the well-trained model does not make sense. One example is:

"*A man is skateboarding on a skateboard. He is wearing a black shirt and black pants. He is wearing a black cap and a black hat. A man is wearing a black cap and a black shirt. A*

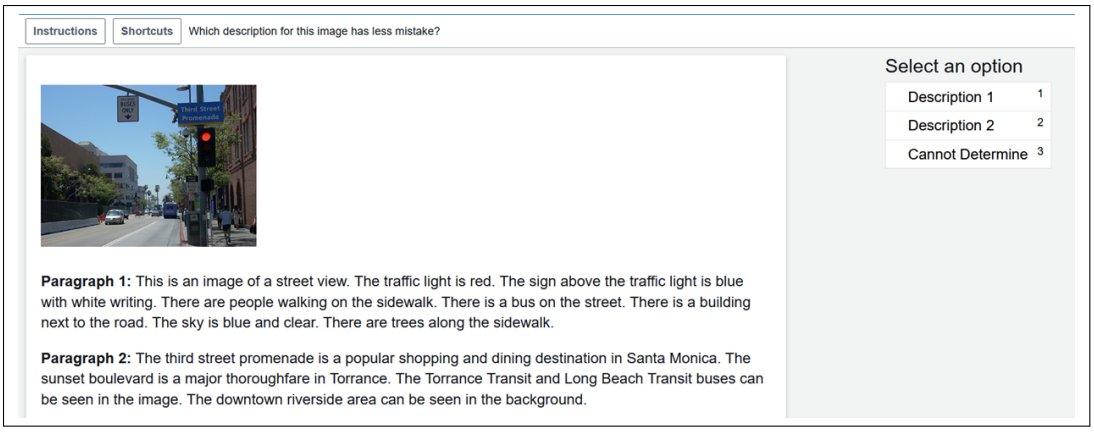


Figure 13: An example screenshots given to participants.

man is wearing a black shirt and a black pants. A man is wearing a black shirt and a black pants. A man is wearing a black shirt and a black pants..... A man is wearing a black shirt and a black pants."

Yet the man actually wears a white hat. This is a common issue of models trained on small language datasets. In comparison, our Figure 9 and 10 include some black and white examples.

4. **Better language quality.** The large language models are trained on tremendous amount of data, leading to much better capability in text generation. Figure 6, 8, 9, and 10 show a few examples of the smooth, coherent paragraphs. In contrast, here is a successful example from Figure 1 of Melas-Kyriazi et al. (2018):

   "*Two people are sitting on a bench. The elephant is sitting on the dirt. The man is sitting on top of the elephant. The woman is wearing a white shirt. The man is wearing a black shirt. There is a tree behind the elephant. There are trees on the ground. There are trees in the background.*"

   We cannot even tell how many people are there according to the text (as a matter of fact, there are two), due to lack of coherence among the sentences.

5. **Awareness of commonsense knowledge.** As suggested by Figure 1, our framework output background knowledge like "*This image captures the everyday life of Cubans, with their traditional horse-drawn carts still in use*", which smoothly complete the paragraph. Traditional IPC methods cannot have enough training data to enable such capability.

E More Details on Human Evaluation

E.1 Instructions

Figure 13 shows an example screenshot we give to the human evaluator. For each metric, we ask different questions, and give different instructions.

Accuracy:

- Question:

  Which of the descriptions has less mistakes describing the image?

- Detailed instruction:

  We would like to evaluate which one of the two image descriptions is more accurate.

  Definition of "accurate":

  Accurate means all concepts appeared in the text are from the image. It is OK to include background knowledge for the objects appeared in the image. It is not OK to include irrelevant objects. For example, if a family is celebrating Christmas eve, it is not OK to say it is a sunny day, but it is OK to say celebrating Christmas is a tradition in western countries.

Comprehensive:

- Question:
  Which description for this image includes more details?
- Detailed instruction:
  We would like to evaluate which one of the two image descriptions is more comprehensive.
  Definition of "comprehensive":
  Comprehensive means all concepts appeared in the image are included in the description.

Coherent:

- Question:
  For the two image descriptions, which one is more coherent?
- Detailed instruction:
  We would like to evaluate which one of the two image descriptions is more coherent.
  Definition of "coherent":
  Coherence describes the way that the elements in our paragraphs hang together to produce meaning. The sentences should be logically connected, instead of just simply concatenating individual sentences together.
  The coherence also include the coherence between the text and image. An image description can include related objects that is not clearly visible in the image, but should not include objects that violates the scene. For example, if the image is a family celebrating Christmas in door, and the background is blurred, it is OK to say there are Christmas decorations behind, but it is not OK to say there is a bird flying in the sky.

Humanlikeness:

- Question:
  Which description for this image is written by human?
- Detailed instruction:
  For the two image descriptions, which one is written by human?
  Only one of the two descriptions is written by human, the other one is from a machine learning model.

E.2 Hypothesis Tests

First, we adopt the voted results of the 5 workers, and perform a binomial test, treating the 200 outcomes as *i.i.d* samples. The null hypothesis here is

Given an image, the probability that human annotation is better than BEST output is 0.5.

To further utilize the detailed outcomes for each image, we perform a Mann–Whitney U test with the null hypothesis as

The number of people that think (the human annotation is better than the BEST output) is similar to the number of people that think (the BEST output is better than the human annotation).

E.3 Budget

We pay \$0.01 to the workers and \$0.01 to Amazon Mechanical Turks for each question answered by one worker. In total we spend \$240 for conducting the experiments reported in Table 3, and another \$80 to test our instructions in the initial exploration stage.

F Applications to Visual Question Answering

The visual clues is a faithful and detailed description of the image, which can be used to answer visual questions leveraging the question answering ability of language models. Specifically, we replaced the

ending of the prompt to be the question, *e.g.*, we replace the "*Describe the image in detail:*" by "*What is the man holding?*". We benchmark its performance in two Visual Question Answering (VQA) datasets – we use the GQA (Hudson and Manning, 2019) dataset for probing the capability of scene understanding, and the OK-VQA (Marino et al., 2019) dataset for the awareness of the commonsense knowledge.

Since no training is performed, BEST generated answer usually have different formatting from the ground truth, causing difficulty in evaluation. For example, for question "*Is the ground blue or brown?*", the ground truth answer in GQA is "*brown*", but the BEST answer is "*The ground in the image is brown.*". Therefore, we use GPT-3 model again to reformat the answer. We refer this evaluation method as *Generative*. Furthermore, for the GQA dataset, the answers in the training set and test set have a large overlap. So we adopt the nearest embedding from the training answers as the final answer, and refer the method as *Discriminative*.

Specifically, for generative evaluation, we show two made-up examples to GPT-3, to let it know the level of detailness we are looking for. For example, for the question "*Is the ground blue or brown?*" and the BEST answer "*The ground in the image is brown.*", we feed the following text into GPT-3:

Question: What is this bird called?

Long answer: The bird in this image is called a cockatoo.

Short answer: Cockatoo.

Question: Is the chair on the left or on the right of the desk?

Long answer: The chair is on the left of the desk.

Short answer: Left.

Question: Is the ground blue or brown?

Long answer: The ground in the image is brown.

Short answer:

The two examples are fixed for all the evaluations.

For discriminative evaluation, we collect the possible answers $\{a_i^{\mathrm{t}}\}$ in the training set, and encode them with text encoder $f_{\mathrm{t}}(\cdot)$. We then encode the generative short answer a^{g} generated as above, and adopt the answer with highest similarity as the final answer a^{f},

$$a^{\mathrm{f}} = \underset{a_i^{\mathrm{t}}}{\operatorname{argmax}} \langle f_{\mathrm{t}}(a_i^{\mathrm{t}}), f_{\mathrm{t}}(a^{\mathrm{g}}) \rangle.$$

Table 6: The VQA accuracy on GQA and OK-VQA datasets.

Method	Evaluation	GQA	OK-VQA
Socratic	Generative	24.95	16.50
	Discriminative	26.89	–
Visual Clues	Generative	37.00	**28.89**
	Discriminative	**39.93**	–
BLIP	Exact Match	47.58	43.62

Table 6 shows the evaluation results. BEST outperforms Socratic models significantly, suggesting our visual clues are better image representations. We also benchmark the accuracy on BLIP (finetuned on VQA v2 dataset (Goyal et al., 2017) and Visual Genome dataset (Krishna et al., 2017)) for reference, which is not directly comparable since its pretrain and finetune datasets have a significant overlap with the evaluation datasets. Figure 14 and Figure 15 show some success and failure cases of from the VQA datasets.

Question: *What material is the crosswalk in front of the stores?*
GT: *Concrete*
BEST: *Concrete*

Question: *What ingredient is missing from the picture to make a pb and j sandwich?*
GT: *Jelly*
BEST: *Jelly*

Figure 14: Examples of the success cases from the GQA (left) and the OK-VQA (right).

Question: *What place is pictured?*
GT: *Shore*
BEST: *Africa*

Question: *Where in the world would you find this structure?*
GT: *California*
BEST: *Los Angeles, California*

Figure 15: Examples of the failure cases from the GQA (left) and the OK-VQA (right).

G Pseudo Code

The following is the key function in our code with `PyTorch`. Here, `model` is the Florence model, `od_model` is the object detection model, `text_features_for_tag` is the embedding of the customized tags, `text_features` is the embedding of the VG tags, and `text_features_for_attributes` is the embedding of the VG attributes.

```
def predict(img):
    threshold = 0.2
    min_keep = 5
    num_each_answer = 20
    topn = 5
    image = Image.open(img).convert('RGB')
    image_for_tag = preprocess(image).unsqueeze(0).to(device1)
    with torch.no_grad():
        image_features_whole = model.encode_image(image_for_tag)
        logits_per_image = model.logit_scale.exp() * image_features_whole @ text_features_for_tag
        probs = logits_per_image.softmax(dim=-1).cpu().numpy()

    predictions = [(float(probs[0][i]), label) for i, label in enumerate(labels_for_tag)]
    predictions = sorted(predictions, reverse=True)
    tags = []
    for i in range(topn):
        pred = predictions[i][1].split(';')[0]
        tags.append(pred)
    tags = ', '.join(tags)

    regions, boxes = detect_object(od_model, od_preprocess, image)

    if len(regions) != 0:
        images = torch.stack([preprocess(im) for im in regions]).to(device1)
        with torch.no_grad():
            image_features = model.encode_image(images)
            logits_per_image = image_features @ text_features
            probs = logits_per_image.cpu().numpy()

            logits_per_image_for_attributes = model.logit_scale.exp() * image_features @ text_features_for_attributes
            probs_for_attributes = logits_per_image_for_attributes.softmax(dim=-1).cpu().numpy()

        width, height = image.size
        image_size = width*height
        results = []
        caption_for_regions = []
        predictions_for_resort = []

        for j in range(len(probs)):
            size = (boxes[j][2]-boxes[j][0])*(boxes[j][3]-boxes[j][1])
            if size<image_size/400:
                continue
            best_attributes = max([(float(probs_for_attributes[j][i]), label) for i, label in enumerate(labels_for_attributes)])

            predictions = [(float(probs[j][i]), label, j, best_attributes[1]) for i, label in enumerate(labels)]
            predictions = sorted(predictions, reverse=True)

            predictions_for_resort.extend(predictions[:min_keep])
            num_keep = 0
            while num_keep<topn and predictions[num_keep][0]>threshold:
                num_keep += 1
            num_keep = max(0, num_keep-1)
            if num_keep>0:
```

```
                pred_labels = [pred[1] for pred in predictions[:num_keep]]
                results.append((boxes[j], pred_labels, size, best_attributes[1]))

                try:
                    caption = blip_captioner(caption_model, regions[j], device=device2)
                    caption_for_regions.append(caption)
                except:
                    caption = ''
                    caption_for_regions.append(caption)
        if len(results)<min_keep:
            results = []
            caption_for_regions = []
            predictions_for_resort = sorted(predictions_for_resort, reverse=True)
            for pred in predictions_for_resort[:min_keep]:
                j = pred[2]
                size = (boxes[j][2]-boxes[j][0])*(boxes[j][3]-boxes[j][1])
                results.append((boxes[j], [pred[1]], size, pred[3]))
                try:
                    caption = blip_captioner(caption_model, regions[j], device=device2)
                    caption_for_regions.append(caption)
                except:
                    caption = ''
                    caption_for_regions.append(caption)

        prompt = 'Objects in this image:\n{}\n\nTags:\nThis image is about {}\n\nDescribe this image in detail:'

        input_texts = set()
        input_texts_caption = set()
        for result, caption in zip(results, caption_for_regions):
            input_text = ''
            input_text += ', or '.join(result[1]) + ' '
            location_x = (result[0][0]+result[0][2])/2
            location_y = (result[0][1]+result[0][3])/2
            if location_x / width < 1/3:
                if location_y/height<1/3:
                    location = 'upper left'
                elif location_y/height>2/3:
                    location = 'lower left'
                else:
                    location = 'left'
            elif location_x / width > 2/3:
                if location_y/height<1/3:
                    location = 'upper right'
                elif location_y/height>2/3:
                    location = 'lower right'
                else:
                    location = 'right'
            else:
                if location_y/height<1/3:
                    location = 'upper middle'
                elif location_y/height>2/3:
                    location = 'lower middle'
                else:
                    location = 'middle'
            size_percentage = result[2]/image_size
            if size_percentage > 0.4:
                size_text = 'large'
            elif size_percentage < 0.05:
                size_text = 'small'
            else:
                size_text = 'moderate-sized'
            input_text += 'is at '+ location + ' of the image and is ' + size_text + ' in the image. Attribute: ' + result[3]

            input_text_caption = caption + '. ' + input_text
            input_texts.add(input_text)
            input_texts_caption.add(input_text_caption)


        input_texts = list(input_texts)
        input_text = '\n'.join(input_texts)

        input_texts_caption = list(input_texts_caption)
        input_text_caption = '\n'.join(input_texts_caption)
        input_text_caption = input_text_caption.split(' ')
        if len(input_text_caption)>1000:
            input_text_caption = input_text_caption[:1000] # for GPT-3 length limit
        input_text_caption = ' '.join(input_text_caption)

        captions = blip_captioner(caption_model, image, device=device2)
        input_text_caption += '\n\nCaption: \n' + captions

        prompt1 = prompt.format(input_text, tags)
        prompt2 = prompt.format(input_text_caption, tags)

    else:
        prompt = '{}Tags:\nThis image is about {}\n\nDescribe this image in detail:'
        captions = blip_captioner(caption_model, image, device=device2)
        prompt1 = prompt.format('', tags)
        prompt2 = prompt.format('Caption:\n'+captions+'\n\n', tags)

    labels_new = []
    answers = get_answer([prompt1]*num_each_answer)['choices']
```

```
    answers = [ans['text'].strip() for ans in answers]
    labels_new.extend(answers)

    answers = get_answer([prompt2]*num_each_answer)['choices']
    answers = [ans['text'].strip() for ans in answers]
    labels_new.extend(answers)

    text = model.tokenizer(labels_new, padding="max_length", truncation=True, max_length=77, return_tensors='pt')
    text = {k: v.to(device1) for k,v in text.items()}

    with torch.no_grad():
        text_features_new = model.encode_text(text)
        logits_per_image = model.logit_scale.exp() * image_features_whole @ text_features_new.t()
        probs = logits_per_image.cpu().numpy()

    max_id = np.argmax(probs[0])
    desc = labels_new[max_id]
    desc_sen = sent_tokenize(desc)

    text = model.tokenizer(desc_sen, padding="max_length", truncation=True, max_length=77, return_tensors='pt')
    text = {k: v.to(device1) for k,v in text.items()}

    with torch.no_grad():
        text_features_new = torch.nn.functional.normalize(model.encode_text(text), dim=-1)
        image_features_whole_norm = torch.nn.functional.normalize(image_features_whole, dim=-1)
        logits_per_image = image_features_whole_norm @ text_features_new.t()
        probs = logits_per_image.cpu().numpy()

        candidate_index = probs[0]>threshold
        desc = '␣'.join([sen for sen, whether_in in zip(desc_sen, candidate_index) if whether_in])
    return desc
```