nocaps: novel object captioning at scale
Supplementary Material

In Section 1, we provide the details about our data collection interface for **nocaps**. Further, in Section 2, we provide some qualitative examples from **nocaps** validation split. In Section 3, we provide additional details in relation to the **nocaps** benchmark. In Section 4, we provide implementation details for our baseline models and finally in Section 5, we provide examples of predicted captions on the three (**in-domain**, **near-domain** and **out-of-domain**) subsets of the **nocaps** validation set.

1. Data Collection Interface

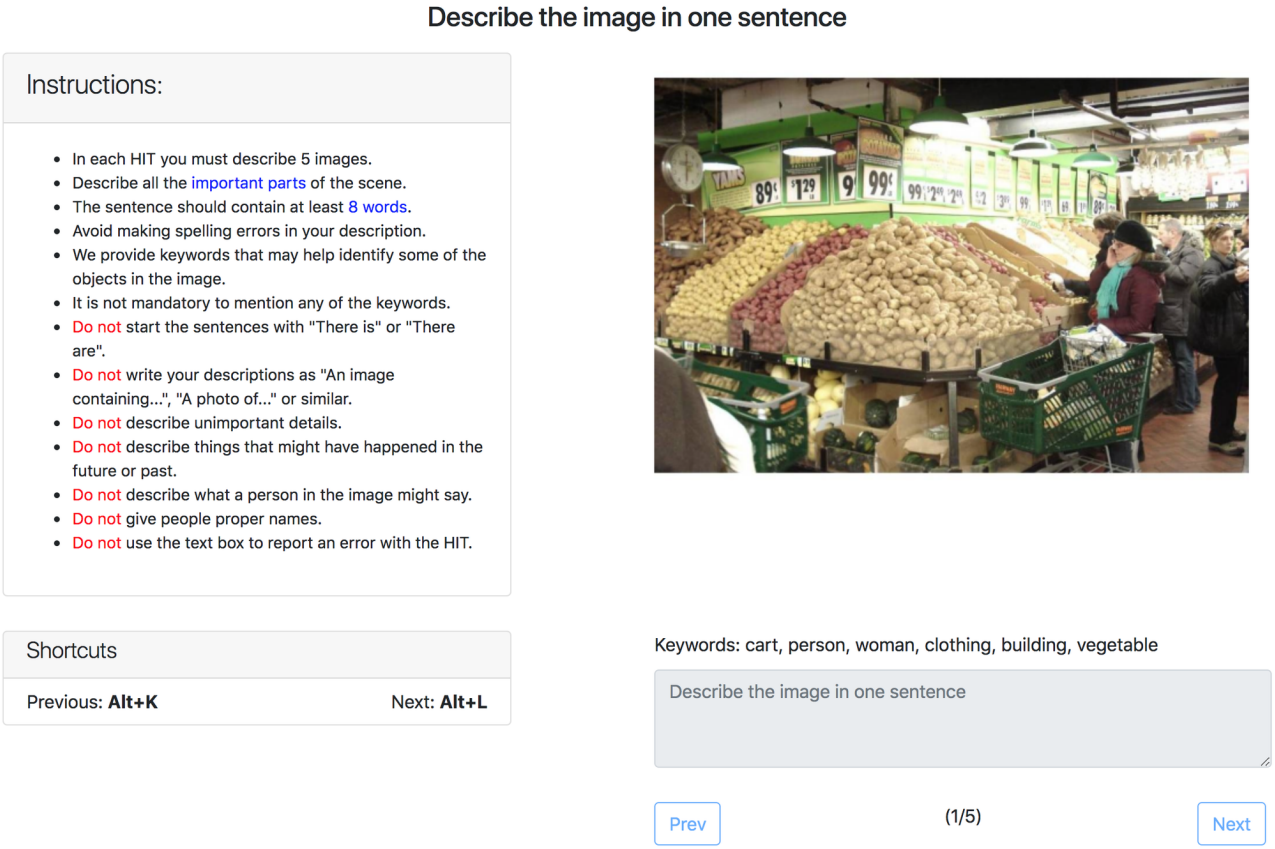


Figure 1: Amazon Mechanical Turk (AMT) user interface with priming for gathering captions. The interface shows a subset of object categories present in the image as keywords. Note that the instruction explicitly states that it is not mandatory to mention any of the displayed keywords. Other instructions are similar to the interface described in [5]

2. Example Reference Captions from nocaps

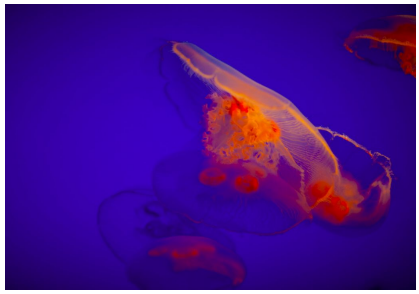
in-domain

1. Two hardcover **books** are on the **table**
2. Two magazines are sitting on a **coffee table**.
3. Two **books** and many crafting supplies are on this **table**.
4. a recipe **book** and sewing **book** on a craft **table**
5. Two hardcover **books** are laying on a **table**.
6. A **table** with two different **books** on it.
7. Two different **books** on sewing and cooking/baking on a **table**.
8. Two magazine **books** are sitting on a **table** with arts and craft materials.
9. A couple of **books** are on a **table**.
10. The **person** is there looking into the **book**.

near-domain

1. **Men** in military uniforms playing instruments in an orchestra.
2. Military officers play brass horns next to each other.
3. Two **men** in camouflage clothing playing the **trumpet**.
4. Two **men** dressed in military outfits play the french horn.
5. Two **people** dressed in camoflauge uniforms playing musical instruments.
6. A couple **people** in uniforms holding tubas by their mouths.
7. Two **people** in uniform are playing the tuba.
8. A couple of military **men** playing the french horn.
9. A **man** in uniform plays a French horn.
10. Two **men** are playing the **trumpet** standing nearby.

out-of-domain



1. Some red invertebrate **jellyfishes** in dark blue water.
2. orange and clear **jellyfish** in dark blue water
3. A red **jellyfish** is swimming around with other red **jellyfish**.
4. Orange **jellyfish** swimming through the water.
5. Bright orange and clear **jellyfish** swim in open water.
6. The fish is going through the very blue water.
7. A bright orange **jellyfish** floating in the water.
8. Several red **jellyfish** swimming in bright blue water.
9. An orange **jellyfish** swimming with a blue background
10. A very vibrantly red **jellyfish** is seen swimming in the water.

1. Jockeys on **horses** racing around a track.
2. Several **horses** are running in a race thru the grass.
3. several **people** racing **horses** around a turn outside
4. Uniformed jockeys on **horses** racing through a grass field.
5. Several **horse** jockies are riding **horses** around a turn.
6. Six men and six **horses** are racing outside
7. A group of men wearing sunglasses and racing on a horse
8. Six **horses** with riders are racing, leaning over at an incredible angle.
9. Seveal **people** wearing goggles and helmets racing **horses**.
10. a row of **horses** and jockeys running in the same direction in a line

1. Two **people** in a fencing match with a **woman** walking by in the background.
2. Two **people** in masks fencing with each other.
3. Two **people** in white garbs are fencing while **people** watch.
4. Two **people** in full gear fencing on white mat.
5. A couple of **people** in white outfits are fencing.
6. Two fencers in white outfits are dueling indoors.
7. A couple of **people** doing a fencing competition inside.
8. Two **people** in white clothes fencing each other.
9. Two **people** in an room competing in a fencing competition.
10. Two **people** in all white holding swords and fencing.

1. A **panda** bear sitting beside a smaller **panda** bear.
2. The **panda** is large and standing over the plant.
3. Two **panda** are eating sticks from plants.
4. Two **panda** bears sitting with greenery surrounding them.
5. two **panda** bears in the bushes eating bamboo sticks
6. two **pandas** sitting in the grass eating some plants
7. two **pandas** are eating a green leaf from a plant
8. Two **pandas** are eating bamboo in a wooded area.
9. **Pandas** enjoy the outside and especially with a friend.
10. Two black and white **panda** bears eating leaf stems

Figure 2: Examples of images belonging to the **in-domain**, **near-domain** and **out-of-domain** subsets of the **nocaps** validation set. Each image is annotated with 10 reference captions, capturing more of the salient content of the image and improving the accuracy of automatic evaluations [1, 13]. Categories in **orange** are **in-domain** object classes while categories in **blue** are **out-of-domain** classes. Note that not all captions mention the ground-truth object classes consistent with the instructions provided on the data collection interface.

in-domain	near-domain	out-of-domain
		
1. A dog sitting beside a man walking on the lawn	1. A woman is sitting with a camera in front of her	1. Some decorations are have red lights you can see at night.
2. A small dog looking up at a person standing next to him.	2. A beautiful brown haired woman next to a camera.	2. A couple of red lanterns floating in the air.
3. A young dog looks up at their owner.	3. A woman poses to take a picture in a mirror.	3. Many red Chinese lanterns are hung outside at night
4. A little puppy looking up at a person.	4. A women with brown hair holding a camera.	4. Floating lighted lanterns on a dark night in the city.
5. A dog sitting on the grass next to a human.	5. A woman behind a camera on a tripod.	5. Dozens of glowing paper lanterns floating off into the sky.
6. A dog is looking up at the person who is wearing jeans.	6. A woman sits with her head leaned behind a camera.	6. A black night sky with red, bright floating lanterns.
7. The tan dog sits patiently beside the person.l	7. A girl looking pity behind a camera on a tripod.	7. Red lanterns floating up to the dark night sky.
8. The white dog is sitting in the grass by a person who is standing up.	8. A woman sits and tilts her head while behind a camera.	8. Chinese lanterns that are red are floating into the sky.
9. The tan dog happily accompanies the human on the grass.	9. A woman is sitting behind a camera with tripod.	9. This town has many lit Chinese lanterns hanging between the buildings.
10. A dog is on the grass is looking to a person	10. A woman with a camera in front of her.	10. The street is filled with light from hanging lanterns.
		
1. people are standing on the side of a food truck	1. A room with a hot tub and sauna.	1. Large silver tanks behind the counter at a restaurant.
2. A food truck parked with people standing in line.	2. A white hot tub is next to some wood.	2. Shiny metal containers with writing are beside each other.
3. people standing outside and ordering from a food truck in the daytime.	3. A jacuzzi sitting on rocks inside of a patio.	3. A brewery with big, silver, metal containers and a sign.
4. The food truck has a line of people in front of the window.	4. A hot tub sits in the middle of the room.	4. A brew station inside of a restaurant.
5. A woman standing in front of a food truck.	5. A jacuzzi sitting near some rocks and a sauna	5. Large steel breweries sit behind a chalkboard displaying different food and drink deals.
6. A food truck outside of a small business with several people eating	6. A hot tub in a room with wooden flooring.	6. A cabinetry with big tin cans and a chalkboard on the top
7. people stand in line to get food from a food truck.	7. A room is shown with a hot tub, decorative plants and some paintings ont he wall.	7. A man works on machinery inside a brewery.
8. A large metal truck serving food to people in a parking lot.	8. A room with a large hot tub and a sauna.	8. The many silver tanks are used for beverage making.
9. men and women speaking in front of a grey food truck that is open for business.	9. A water filled jacuzzi surrounded by smooth river rocks and a wooden deck.	9. A menu is hanging above a craft brewery.
10. woman wearing jeans in front of the truck	10. A white and grey jacuzzi around rock building	10. A man peers at a brewing tank while standing on a step ladder.

Figure 3: More examples of images belonging to the **in-domain**, **near-domain** and **out-of-domain** subsets of the **nocaps** validation set. Each image is annotated with 10 reference captions, capturing more of the salient content of the image and improving the accuracy of automatic evaluations [1, 13]. Categories in **orange** are **in-domain** object classes while categories in **blue** are **out-of-domain** classes. Note that not all captions mention the ground-truth object classes consistent with the instructions provided on the data collection interface.

3. Additional Details about nocaps Benchmark

3.1. Evaluation Subsets

As outlined in Section 3.3 of the main paper, to determine the **in-domain**, **near-domain** and **out-of-domain** subsets of **nocaps**, we first classify Open Images classes as either **in-domain** or **out-of-domain** with respect to COCO. To identify the **in-domain** Open Images classes, we manually map the 80 COCO classes to Open Images classes. We then select an additional 39 Open Images classes that are not COCO classes, but are nonetheless mentioned more than 1,000 times in the COCO captions training set (e.g. 'table', 'plate' and 'tree'), and we classify all 119 of these classes as **in-domain**. The remaining classes are considered to be **out-of-domain**.

To put this in perspective, in Figure 4 we plot the number of mentions of both the **in-domain** classes (in orange) and the **out-of-domain** classes (in blue) in the COCO Captions training set using a log scale. As intended, the **in-domain** object classes occur much more frequently in COCO Captions compared to **out-of-domain** object classes. However, it is worth noting that the **out-of-domain** are not necessarily absent from COCO Captions, but they are relatively infrequent which makes these concepts hard to learn from COCO.

Open Images classes ignored during image subset selection: We also note that 87 Open Images classes were not considered during the image subset selection procedure to create **nocaps**, for one of the following reasons:

- **Parts:** In our image subset selection strategy (refer Section 3.1 of the main paper), we ignored 'part' categories such as 'vehicle registration plate', 'wheel', 'human-eye', which always occur with parent categories such car, person;
- **Super-categories:** Our image subset selection strategy also ignored super-categories such as 'sports equipment', 'home appliance', 'auto part' which are often too broad and subsumes both COCO and Open Images categories;
- **Solo categories:** Certain categories such as 'chime' and 'stapler' did not appear in images alongside any other classes, and so were filtered out by our image subset selection strategy; and
- **Rare categories:** Some rare categories such as 'armadillo', 'pencil sharpener' and 'pizza cutter' do not actually occur in the underlying Open Images val and test splits.

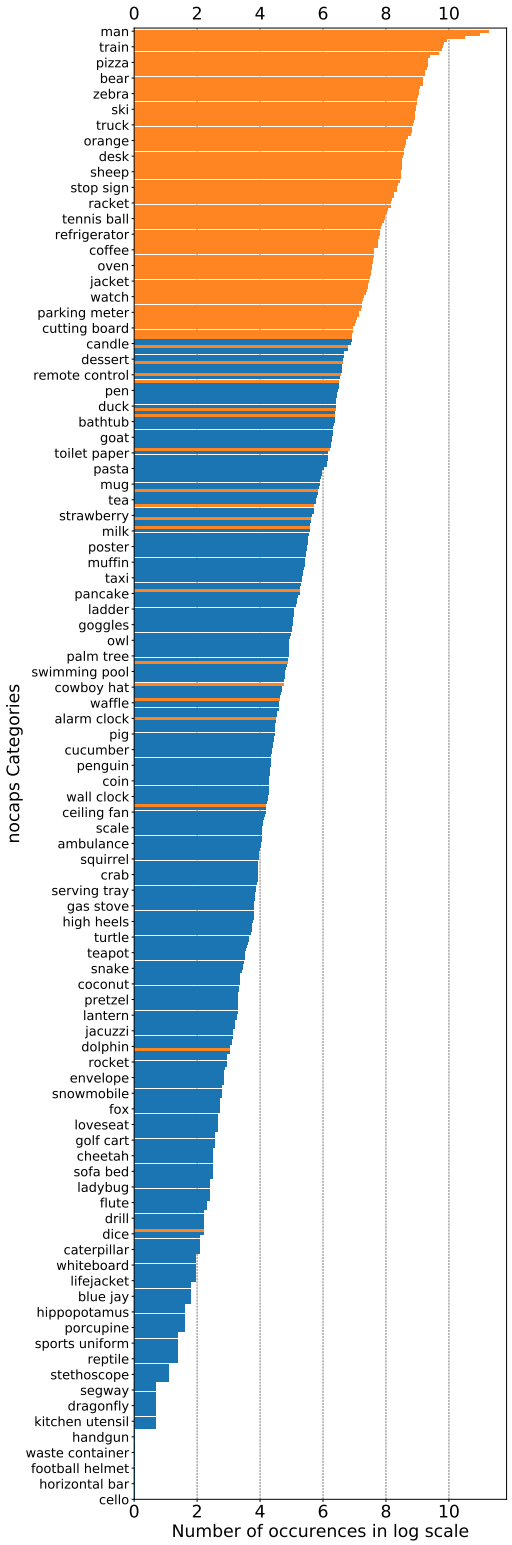


Figure 4: Histogram of mentions in the COCO Captions training set for various Open Images object classes. In **nocaps**, classes in orange are considered to be **in-domain** while classes in blue are classified as **out-of-domain**. *Zoom in for details.*

3.2. T-SNE Visualization in Visual Feature Embedding Space

Figure 5: T-SNE [12] plot comparing the visual similarity between object classes in COCO, **`in-domain`** and **`out-of-domain`** splits of **`nocaps`**. For each object class in a particular split, we extract bottom-up image features from the Faster-RCNN detector made publicly available by [3] and mean pool them to form a 2048-dimensional vector. We further apply PCA on the feature vectors for all object classes and pick the first 128 principal components. Using these feature vectors of reduced dimension, we compute the exact form T-SNE with perplexity 30. We observe that: (a) **`in-domain`** shows high visual similarity to COCO– green and brown points of same object class are close to each other. (b) Many **`out-of-domain`** classes are visually different from **`in-domain`** classes – large clusters of blue, far away from green and brown. (c) **`out-of-domain`** also covers many visually similar concepts to COCO– blue points filling the gaps between sparse clusters green/brown points.

3.3. T-SNE Visualization in Linguistic Feature Embedding Space

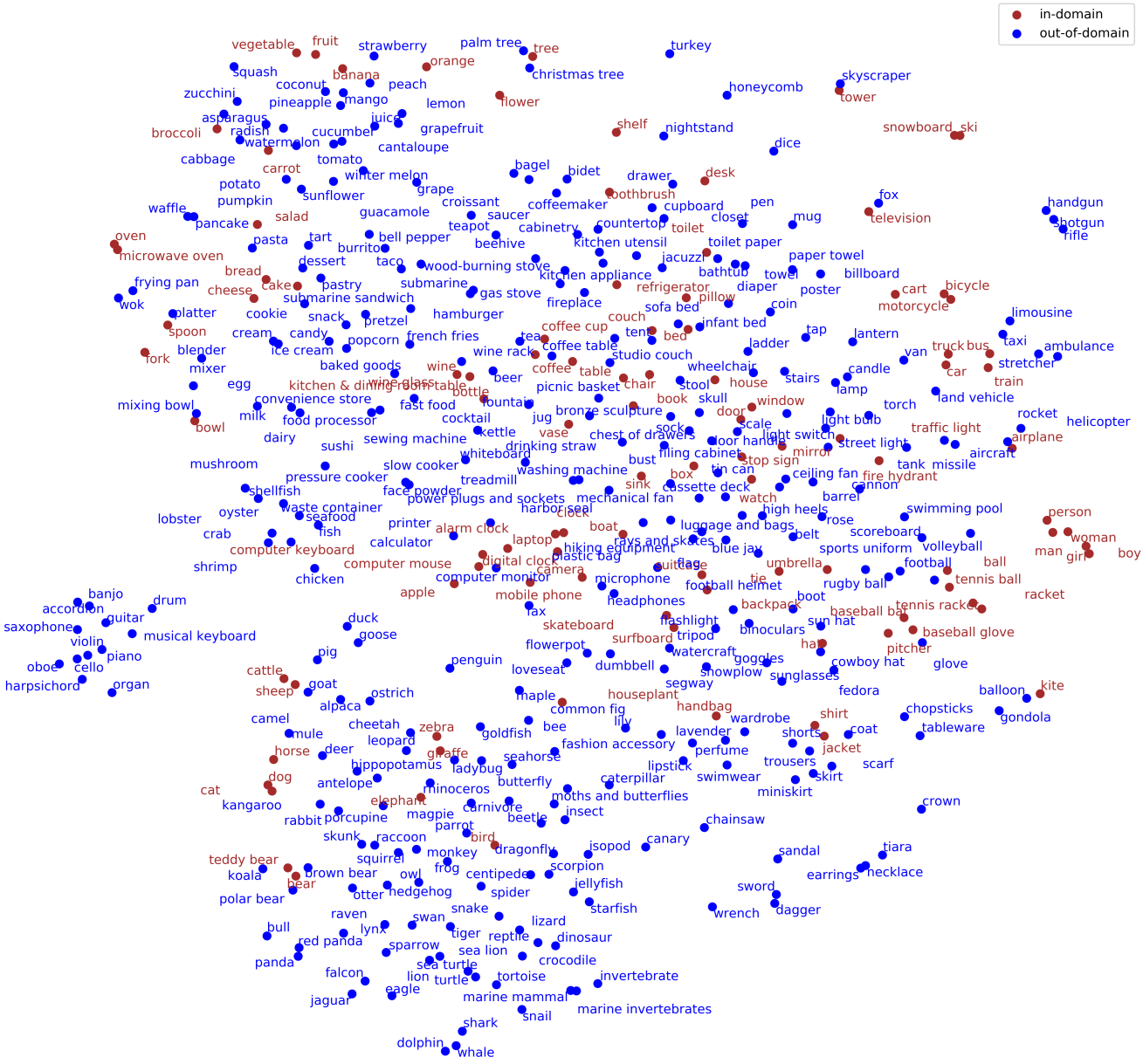


Figure 6: T-SNE [12] plot comparing the linguistic similarity between object classes in **`in-domain`** and **`out-of-domain`** splits of **`nocaps`**. For each object class in a particular split, we obtain 300-dimensional GloVe [9]. We further apply PCA on these GloVe vectors vectors for all object classes and pick the first 128 principal components. Using these feature vectors of reduced dimension, we compute the exact form T-SNE with perplexity 30. We observe that: (a) Many **`out-of-domain`** classes are linguistically different from **`in-domain`** classes – large clusters of blue points far away from brown points. (b) **`out-of-domain`** also covers many linguistically similar, fine-grained classes not present in **`in-domain`**– blue points filling gaps in sparse clusters of brown points.

3.4. Linguistic Similarity to COCO

Overall, our collection methodology closely follows COCO. However, we do introduce keyword priming to the collection interface (refer Figure 1) which has the potential to introduce some linguistic differences between **nocaps** and COCO. To quantitatively assess linguistic differences between the two datasets, we review the performance of COCO-trained models on the **nocaps** validation set while controlling for visual similarity to COCO. As a proxy for visual similarity to COCO, we use the average cosine distance in FC7 CNN feature space between each **nocaps** image and the 10 closest COCO images.

As illustrated in Table 1, the baseline UpDown model (trained using COCO) exceeds human performance on the decile of **nocaps** images which are most similar to COCO images (decile=1, avg. cosine distance=0.15), consistent with the trends seen in the COCO dataset. This suggests that the linguistic structure of COCO and **nocaps** captions is extremely similar. As the **nocaps** images become visually more distinct from COCO images, the performance of UpDown drops consistently. This suggests that no linguistic variations have been introduced between COCO and **nocaps** due to priming and the degradation in the performance is due to visual differences. Similar trends are observed for our best model (UpDown + ELMo + CBS) although the performance degradation with increasing visual dissimilarity to COCO is much less.

	nocaps test CIDEr scores										
Decile	1	2	3	4	5	6	7	8	9	10	Overall
Avg Cosine Dist from COCO	0.15	0.18	0.20	0.21	0.23	0.24	0.25	0.27	0.30	0.35	
UpDown	**82.6**	72.6	63.9	61.1	55.9	55.0	50.7	48.5	39.2	28.7	54.5
UpDown + ELMo + CBS	**81.8**	77.3	75.4	72.8	77.1	78.2	72.3	71.7	70.6	65.1	73.1
Human	77.8	**78.0**	**82.4**	**84.0**	**86.2**	**88.8**	**89.4**	**91.2**	**97.3**	**95.6**	**85.3**

Table 1: CIDEr scores on **nocaps** test deciles split by visual similarity to COCO (using CNN features). Our models exceed human performance on the decile of **nocaps** images that are most visually similar to COCO. This suggests that after controlling for visual variations the linguistic structure of COCO and **nocaps** captions is highly similar.

4. Additional Implementation Details for Baseline Models

4.1. Neural Baby Talk (NBT)

In this section, we describe our modifications to the original authors' implementation of Neural Baby Talk (NBT) [8] to enable the model to produce captions for images containing novel objects present in **nocaps**.

Grounding Regions for Visual Words

Given an image, NBT leverages an object detector to obtain a set of candidate image region proposals, and further produces a caption template, with slots explicitly tied to specific image regions. In order to accurately caption **nocaps** images, the object detector providing candidate region proposals must be able to detect the object classes present in **nocaps** (and broadly, Open Images). Hence, we use a Faster-RCNN [11] model pre-trained using Open Images V4 [6] (referred as **OI detector** henceforth), to obtain candidate region proposals as described in Section 4 of the main paper. This model can detect 601 object classes of Open Images, which includes the novel object classes of **nocaps**. In contrast, the authors' implementation uses a Faster-RCNN trained using COCO.

For every image in COCO train 2017 split, we extract image region proposals after the second stage of detection, with an IoU threshold of 0.5 to avoid highly overlapping region proposals, and a class detection confidence threshold of 0.5 to reduce false positive detections. This results in number of region proposals per image varies up to a maximum of 18.

Bottom-Up Visual Features

The language model in NBT (Refer Figure 4 in [8]) has two separate attention layers, and takes visual features as input in three different manners:

- The first attention layer learns an attention distribution over region features, extracted using ResNet-101 + RoI Align layer.
- The second attention layer learns an attention distribution over spatial CNN features from the last convolutional layer of ResNet-101 (7 x 7 grid, 2048 channels).
- The word embedding input is concatenated with FC7 features from ResNet-101 at every time-step.

All the three listed visual features are extracted using ResNet-101, with the first being specific to visual words, while the second and third provide the holistic context of the image. We replace the ResNet-101 feature extractor with the publicly available Faster-RCNN model pre-trained using Visual Genome (referred as **VG detector** henceforth), same as [3]. Given a set of candidate region proposals obtained from **OI detector**, we extract 2048-dimensional bottom-up features using the **VG detector** and use them as input to first attention layer (and also for input to the Pointer Network). For input to the second attention layer, we extract top-36 bottom-up features (class agnostic) using the **VG detector**. Similarly, we perform mean-pooling of these 36 features for input to the language model at every time-step.

Fine-grained Class Mapping

NBT fills the slots in each caption template using words corresponding to the object classes detected in the corresponding image regions. However, object classes are coarse labels (e.g. 'cake'), whereas captions typically refer entities in a fine-grained fashion (e.g. 'cheesecake', 'cupcake', 'coffeecake' etc.). To account for these linguistic variations, NBT predicts a fine-grained class for each object class using a separate MLP classifier. To determine the output vocabulary for this fine-grained classifier we extend the fine-grained class mapping used for COCO (Refer Table 5 in [8]), adding Open Images object classes. Several fine-grained classes in original mapping are already present in Open Images (e.g. 'man', 'woman' – fine-grained classes of 'person'), we drop them as fine-grained classes from original mapping and retain them as Open Images object classes.

Visual Word Prediction Criterion

In order to ensure correctness in visual grounding, the authors' implementation uses three criteria to decide whether a particular region proposal should be tied with a "slot" in the caption template. At any time during decoding, when the Pointer Network attends to a visual feature (instead of the visual sentinel), the corresponding region proposal is tied with the "slot" if:

- The class prediction threshold of this region proposal is higher than 0.5.
- The IoU of this region proposal with at least one of the ground truth bounding boxes is greater than 0.5.
- The predicted class is same as the object class of ground truth bounding box having highest IoU with this region proposal.

We drop the third criterion, as the **OI detector** can predict several fine-grained classes in context of COCO, such as 'man' and 'woman' (while the ground truth object class would be 'person'). Keeping the third criterion intact in **nocaps** setting would suppress such region proposals, and result in lesser visual grounding, which is not desirable for NBT. Relaxation of this criterion might introduce false positives from detection in the caption but prevents reduction in visual grounding.

We use the same optimization hyper-parameters as the authors' implementation. We encourage the reader to refer the authors' implementation for further details. We will release code for our modifications.

4.2. Constrained Beam Search (CBS)

Determining Constraints

When using constrained beam search (CBS) [2], we decoded the model in question while forcing the generated caption to include words corresponding to object classes detected in the image. For object detection, we use the same Faster-RCNN [11] model pre-trained using Open Images V4 [6] (**`OI detector`**) that is used in conjunction with NBT. However, not all detected object classes are used as constraints. We perform constraint filtering by removing the 39 object classes listed in Table 2 from the constraint set, as these classes are either object parts, or classes that we consider to be either too rare or too broad. We also suppress highly overlapping objects as described in Section 4 of the main paper.

Parts	Too Rare or Too Broad
Human Eye	Clothing
Human Head	Footwear
Human Face	Fashion Accessory
Human Mouth	Sports Equipment
Human Ear	Hiking Equipment
Human Nose	Mammal
Human Hair	Personal Care
Human Hand	Bathroom Accessory
Human Foot	Plumbing Fixture
Human Arm	Tree
Human Leg	Building
Human Beard	Plant
Human Body	Land Vehicle
Vehicle Registration Plate	Person
Wheel	Man
Seat Belt	Woman
Tire	Boy
Bicycle Wheel	Girl
Auto Part	
Door Handle	
Skull	

Table 2: Remove Class List for object filtering

To quantify the impact of this simple constraint filtering heuristic, in Table 3 we report the results of the following ablation studies:

- Using all the object classes for constraints (w/o class),
- Using overlapping objects for constraints (w/o overlap), and
- Using no filtering heuristic at all (w/o both).

Note that in all cases we rank objects based on confident score for detected objects and pick the top-3 as the constraints. We report results for three models, the baseline model (UpDown), the baseline model using Glove [9] and dependency-based [7] word embeddings (UpDown + GD) and our ELMo-based model (UpDown + ELMo +CBS). Table 3 shows that removing the above 39 classes significantly improves the performance of constrained beam search and removing overlapping objects can also slightly improve the performance. This conclusion is consistent across the three models.

Finite State Machine

Constrained Beam Search implements constraints in the decoding process using a Finite State Machine (FSM). In all experiments we use a 24 state FSM. We use 8 states for standard three single word constraints D_1, D_2 and D_3. As shown in Figure 9, the outputs of this FSM are the captions that mention at least two constraints out of three. Each D_i (i = 1,2,3) represents a set of alternative constraint words (e.g., bike, bikes). D_i can also be multi-word expressions. Our FSM dynamically support two-word or three-word phrases in D_i by extending additional one states (see Figure 7) or two states (see Figure 8) for two-word or three-word phrases respectively. Since D_1, D_2 and D_3 are all used 4 times in the base eight-state FSM, we need to allocate 4 states for a single two-word expression and 8 states for a single three-word expression.

	In-Domain		Near-Domain		Out-of-Domain		Overall	
	CIDEr	SPICE	CIDEr	SPICE	CIDEr	SPICE	CIDEr	SPICE
UpDown + CBS w/o both	73.4	11.2	68.0	10.9	65.2	9.8	68.2	10.7
UpDown + CBS w/o class	72.8	11.2	68.6	10.9	65.5	9.7	68.6	10.8
UpDown + CBS w/o overlap	80.6	12.0	73.5	11.3	66.4	9.8	73.1	11.1
UpDown + CBS	80.0	12.0	73.6	11.3	66.4	9.7	73.1	11.1
UpDown + GD + CBS w/o both	72.8	11.2	68.4	10.8	66.3	9.8	68.6	10.7
UpDown + GD + CBS w/o class	72.3	11.2	68.6	10.9	66.9	9.7	68.8	10.7
UpDown + GD + CBS w/o overlap	77.0	12.0	73.5	11.4	67.2	9.7	72.8	11.1
UpDown + GD + CBS	77.0	12.0	73.6	11.4	69.5	9.7	73.2	11.1
UpDown + ELMo + CBS w/o both	73.3	11.5	68.6	10.9	70.0	10.8	69.6	10.8
UpDown + ELMo + CBS w/o class	73.5	11.5	69.2	11.0	69.9	9.9	70.0	10.9
UpDown + ELMo + CBS w/o overlap	79.8	12.3	73.7	11.4	72.0	9.9	74.2	11.2
UpDown + ELMo + CBS	**79.3**	**12.4**	**73.8**	**11.4**	**71.7**	**9.9**	**74.3**	**11.2**
Human	**83.3**	**13.9**	**85.5**	**14.3**	**91.4**	**13.7**	**87.1**	**14.1**

Table 3: We investigate the effect of different object filtering strategies in Constrained Beam Search and report the model performance in **nocaps** val. We find that using both strategies with the ELMo model performs best.

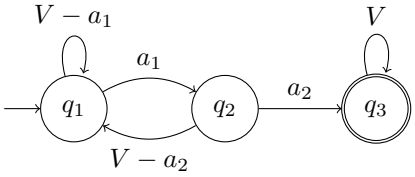


Figure 7: FSM for a two-word phrase $\{a_1, a_2\}$ constraint

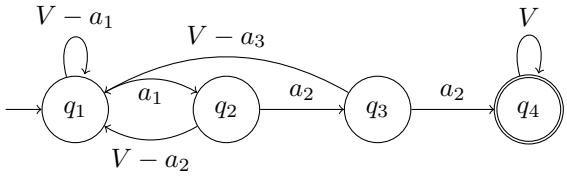


Figure 8: FSM for a three-word phrase $\{a_1, a_2, a_3\}$ constraint

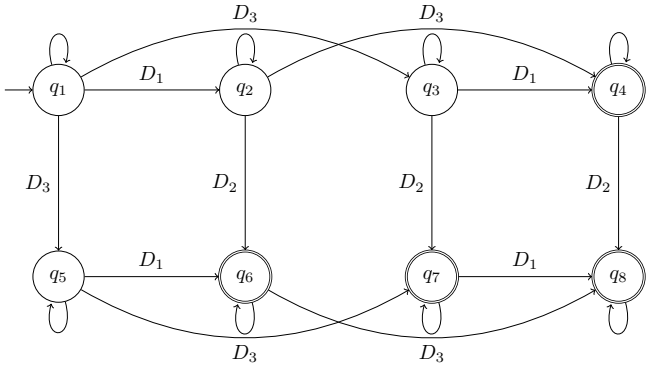


Figure 9: FSM for D_1, D_2, D_3 constraint

Integrating UpDown Model with ELMo

When using ELMo [10], we use a dynamic representation of w_c, $\bar{h}_t^1$ and $\bar{h}_t^2$ as the input word embedding w_{ELMo}^t for our caption model. w_c is the character embedding of input words and $\bar{h}_t^i$ $(i = 1, 2)$ is the hidden output of i^{th} LSTM layer of ELMo. We combine them via:

$$w_{ELMo}^t = \gamma_0 \cdot w_c + \gamma_1 \cdot \bar{h}_t^1 + \gamma_2 \cdot \bar{h}_t^2 \tag{1}$$

where γ_i (i=0, 1, 2) are three trainable scalars. When using w_{ELMo}^t as the external word representation of other models, we fixed all the parameters of ELMo but γ_i (i=0, 1, 2).

In addition, to handle unseen objects in training data, following [2], we initialize the softmax layer matrix (W_p, b_p) using word embedding and keep this layer fixed during training. This allow our caption model to produce similar logits score for the words that share similar vectors and values in W_p and b_p. We have:

$$W_p = W_{ELMo} \tag{2}$$

$$b_p = b_{ELMo} \tag{3}$$

where W_{ELMo} and b_{ELMo} is the softmax layer in original ELMo language model. To align the different dimension in softmax layer and LSTM hidden state, we add an additional fully connected layer with a non-linearity function $tanh$. We have:

$$v_t = tanh(W_t h_t^2 + b_t) \tag{4}$$

$$P(y_t | y_{1:t-1}, I) = softmax(W_p v_t + b_p) \tag{5}$$

where $W_t \in \mathbb{R}^{H \times E}$, $b_t \in \mathbb{R}^E$, H is LSTM hidden dimension, E is the word embedding dimension, $W_p \in \mathbb{R}^{E \times D}$, $b_p \in \mathbb{R}^D$ and D is the vocabulary size.

Other details of using ELMo

In our experiment, we use the full tensorflow checkpoint trained on 1 Billion Word Language Model Benchmark[1] from official ELMo tensorflow implementation project[2].

When selecting vocabularies for our model, we first extract all words from COCO captions and open image object labels. We then extend the open image object labels to both singular and plural word forms. Finally, we remove all the words that are not in ELMo output vocabularies. This allow us to use ELMo LM prediction for each decoding step.

Our UpDown + ELMo model is optimized by SGD [4]. We conduct hyper-parameter tuning the model and choose the model based on its performance on **nocaps** val. Table 4 shows the chosen hyper-parameters for the UpDown Model in the paper.

Parameter	Value	Parameter	Value
Batch Size	150	Attention Size	768
LSTM Hidden Size	1200	Word Dropout	0.2
Image Feature	2048	ELMo Embedding	512
Learning Rate	0.015	Momentum	0.9
Clip Gradients	12.5	Weight Decay	0.001

Table 4: Hyper-parameters for UpDown Model

[1] http://www.statmt.org/lm-benchmark/

[2] https://github.com/allenai/bilm-tf/

5. Example Model Predictions

Method	in-domain 	near-domain 	out-of-domain 
UpDown	A man in a white shirt is playing baseball.	A couple of men standing on top of a truck.	A group of vases sitting on top of a table.
UpDown + ELMo	A group of people standing around a blue table.	A couple of men standing next to a truck.	Two vases sitting next to each other on a table.
UpDown + ELMo + CBS	A group of people standing near a blue table.	A couple of men standing on top of a **tank**.	A **teapot** sitting on top of a table next to a **vase**.
UpDown + ELMo + CBS + GT	A group of people standing around a blue table.	A couple of men standing on top of a **tank**.	A couple of **kettle jugs** sitting next to each other.
NBT	A group of **men** standing in a field.	A **man** standing on the back of a **tank**.	A couple of **kettles** are sitting on a table.
NBT + CBS	A couple of **men** standing on a tennis court.	A **man** standing on top of a **tank** with a truck.	A close up of a **kettle** on a table.
NBT + CBS + GT	A group of **men** are standing in a field.	A man standing on top of a **tank plant**.	Two **kettles** and **teapot jugs** are sitting on a table.
Human	Two people in karate uniforms spar in front of a crowd.	Two men sitting on a tank parked in the bush.	Ceramic jugs are on display in a glass case.

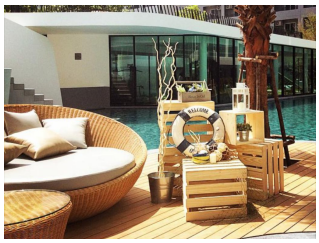
Method	in-domain 	near-domain 	out-of-domain 
UpDown	A woman riding a bike with a statue on her head.	A couple of chairs sitting in front of a building.	A bird sitting on the ground in the grass.
UpDown + ELMo	There is a woman that is riding a bike.	A room that has a lot of furniture in it.	A dog laying on the ground next to a stuffed animal.
UpDown + ELMo + CBS	There is a woman that is riding a bike.	Two **pillows** and a table in the **house**.	A dog laying on the ground next to a **tortoise**.
UpDown + ELMo + CBS + GT	There is a woman that is riding a bike.	Two **couches** and a table in a **house**.	A dog laying on the ground next to a **tortoise**.
NBT	A man is riding a **clothing** on a bike.	A table with a **couch** and a table.	A **tortoise** is laying on top of the ground.
NBT + CBS	A woman is riding a **clothing** in the street.	A couple of **pillows** on a wooden table in a **couch**.	A **tortoise** that is sitting on the ground.
NBT + CBS + GT	A man is riding a **clothing** on a **person**.	A **house** and a **studio couch** of **couches** in a room.	A **tortoise** is laying on the ground in the grass.
Human	People are performing in an open cultural dance.	On the deck of a pool is a couch and a display of a safety ring.	Three tortoises crawl on soil and wood chips in an enclosure.

Figure 10: Some challenging images from **nocaps** and corresponding captions generated by existing approaches. The constraints given to the CBS are shown in **blue**. The visual words associated with NBT are shown in **red**.

Method	in-domain	near-domain	out-of-domain
			
UpDown	A group of people are playing a game.	A large white sign on a city street.	A bear laying in the grass near a tree.
UpDown + ELMo	A woman in a pink dress is holding a child.	A large white bus parked on the side of a road.	A bear that is laying down in the grass.
UpDown + ELMo + CBS	A woman in a pink dress is holding a child.	A **billboard** that has a **street light** on it.	A **red panda** is walking through the grass.
UpDown + ELMo + CBS + GT	A woman in a pink dress is holding a child.	A large white bus parked next to a **billboard**.	A **red panda** is walking through the grass.
NBT	A group of **man** are standing in a field.	A **billboard** sign on the side of a building.	A brown **red panda** is laying on the grass.
NBT + CBS	A group of **man** are playing a baseball game.	A picture of **billboard** sign on the **street light**.	A **tree** and a brown **red panda** in a field.
NBT + CBS + GT	A group of **man** are standing on a field.	A **billboard** sign on the side of a building.	A brown **red panda** lying on top of a field.
Human	Two sumo wrestlers are wrestling while a crowd of men and women watch.	A man is standing on the ladder and working at the billboard.	The red panda trots across the forest floor.

Method	in-domain	near-domain	out-of-domain
			
UpDown	A woman wearing a white shirt and a white shirt.	A person riding a yellow bike in the field.	A close up of a cat looking at a bird.
UpDown + ELMo	A woman wearing a white shirt and white shirt.	A woman sitting on a yellow bike.	A close up of a bird with its mouth open.
UpDown + ELMo + CBS	A woman wearing a white **suit** and a white shirt.	A person sitting on a bicycle with a **wheelchair**.	A **sea lion** standing with its mouth open.
UpDown + ELMo + CBS + GT	A woman wearing a white **suit** and a white shirt.	A person sitting on a **bicycle** with a **wheelchair**.	A **sea lion** standing next to a **harbor seal**.
NBT	A woman wearing a white shirt is wearing a hat.	A man sitting on a **wheelchair** with a **bike**.	A close up of a **sea lion** and **harbor seal** with its head.
NBT + CBS	A **suit** of woman wearing a white suit.	A man sitting on a **wheelchair** and a **bike**.	A close up of a **harbor seal** of a **sea lion**.
NBT + CBS + GT	A **suit** of woman wearing a white shirt.	A **bicycle** sitting on a **wheelchair** with a **bike**.	A close up of a **harbor seal** of a **sea lion**.
Human	The man has a wrap on his head and a white beard.	A person sitting in a yellow chair with wheels.	A brown and gray sea lion looking at the photographer.

Figure 11: Some challenging images from **nocaps** and corresponding captions generated by existing approaches. The constraints given to the CBS are shown in **blue**. The visual words associated with NBT are shown in **red**.

References

[1] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. SPICE: Semantic Propositional Image Caption Evaluation. 2016. 2, 3

[2] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Guided open vocabulary image captioning with constrained beam search. 2017. 9, 11

[3] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. 2018. 5, 8

[4] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In Proceedings of COMPSTAT'2010, pages 177–186. Springer, 2010. 11

[5] Xinlei Chen, Tsung-Yi Lin Hao Fang, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollar, and C. Lawrence Zitnick. Microsoft COCO Captions: Data Collection and Evaluation Server. arXiv preprint arXiv:1504.00325, 2015. 1

[6] Ivan Krasin, Tom Duerig, Neil Alldrin, Vittorio Ferrari, Sami Abu-El-Haija, Alina Kuznetsova, Hassan Rom, Jasper Uijlings, Stefan Popov, Andreas Veit, Serge Belongie, Victor Gomes, Abhinav Gupta, Chen Sun, Gal Chechik, David Cai, Zheyun Feng, Dhyanesh Narayanan, and Kevin Murphy. Openimages: A public dataset for large-scale multi-label and multi-class image classification. Dataset available from https://github.com/openimages, 2017. 8, 9

[7] Omer Levy and Yoav Goldberg. Dependency-based word embeddings. 2014. 9

[8] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Neural baby talk. 2018. 8

[9] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. GloVe: Global Vectors for Word Representation. 2014. 6, 9

[10] Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 2227–2237. Association for Computational Linguistics, 2018. 11

[11] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. 2015. 8, 9

[12] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. Journal of Machine Learning Research, 9:2579–2605, 2008. 5, 6

[13] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. CIDEr: Consensus-based image description evaluation. 2015. 2, 3