Supplementary Material for NLX-GPT: A Model for Natural Language Explanations in Vision and Vision-Language Tasks

1. Implementation Details

In this section, we provide implementation details for pretraining, concept detection and finetuning. Please note that for all three stages, the vision backbone is frozen and not fine-tuned at any time.

1.1. Pretraining

We choose 4 publicly available image-caption datasets for pre-training: MSCOCO [8], Visual Genome Region Descriptions [7], Flickr30K [12] and Image-Paragraph Stanford dataset [6]. The region descriptions of Visual Genome are of large-scale ($\sim$5M). However, they are short compared to COCO and Flickr30K, and each region description is associated to a small part of the image. We therefore combine the per-image region descriptions to form a paragraph, which acts as the caption associated to the image. The maximum length of the caption is set to 70. There are also other image-caption datasets that can be used for pretraining, such as Conceptual Captions of $\sim$3M pairs [14] as well as SBU of $\sim$1M pairs [10]. We leave these to future works since the results with the 4 datasets we mentioned already give satisfactory results when fine-tuned on the Natural Language Explanations (NLE) downstream task. We initialize the model with the Distilled GPT-2 weights[1]. The model is trained with the ADAM optimizer [5] with a batch size of 768 and a learning rate of 1e-4 which is linearly decayed to 0 over the total number of training steps. We evaluate the pre-trained model performance on the "Karpathy" test split [3] of COCO Captions [8].

1.2. Concept Detection

Let H, W, P, Y be the height, width, patch size and total number of patches of the image, respectively. In order to predict image concepts, the output representation of the vision backbone for all image patches is utilized. Let the output of the vision backbone be $X \in \mathbb{R}^{Y \times d}$ where d is the output dimension. In the case of ViT, we do not utilize the `CLS` reduced representation since it is mainly optimized for other objectives such as image classification or contrastive learning, and may not be optimal for concept learning which requires a much broader view of all the image patches. Therefore, we learn an attention reduced representation of all the Y image patches. We first feed X into 2 linear layers with a ReLU activation function in-between, followed by a residual [2] and layer normalization layer [1] to get an output $V \in \mathbb{R}^{Y \times d_k}$. We utilize an attention-summary layer implemented as $\mathbf{s} = \sum_{i=1}^{Y} \alpha_i \mathbf{x}_i$ where $\alpha_i = \frac{\exp(\mathbf{w}_x^T \mathbf{v}_i)}{\sum_{j=1}^{Y} \exp(\mathbf{w}_x^T \mathbf{v}_j)}$ in order to reduce X into to a single feature vector $\mathbf{s}$, where $w \in \mathbb{R}^{d_k \times 1}$ are learnable weights. $\mathbf{s}$ is then fed into a classification layer over all the concepts. We train the concept detection head on the Visual Genome 2.8M attributes dataset [7] with a batch size of 256 using the ADAM optimizer [5] with a learning rate of 2e-3 which is decayed by a factor of 0.8 every 3 epochs. We use dropout with a keeping probability of 0.9. In order to measure the accuracy, we take the top-K predicted concepts and check how many of them appear in the ground-truth concepts. For each sample in the test set, we count the number of elements in the set $PR \cap GT$, where PR indicates the top-K predicted concepts and GT are all the ground-truth concepts for the tested sample. We measure the acuracy at $K = 5, 10$ and 15. Our resulting accuracy scores are 83.0, 73.61 and 65.99, respectively.

1.3. Finetuning

We finetune our pretrained model on 4 NLE datasets: VQA-X, ACT-X, e-SNLI-VE and VCR (explained in the next section). The input sequence consists of tokens of the (question, answer, explanation) and each token is then fed to an embedding layer to get a representation of the word. In order allow the model to distinguish between the question, answer and explanation, we add the embeddings of the segment ID `<ques>`, `<ans>` and `<exp>` to all question, answer and explanation token embeddings, respectively. We also add a continuous positional embedding for the complete sequence starting from the question up until the explanation. We use the ADAM optimizer [5] with a learning rate of 1e-5 which is linearly decayed to 0 over the total number of training steps. For models which are finetuned from the pretrained image captioning model (models for VQA-X and ACT-X), we use a batch size of 32. For other models,

[1] https://github.com/huggingface/transformers

Table 1. Filtered Scores on VCR dataset. B, M, R-L, C, S, BS are short for BLEU, METEOR, ROUGE-L, CIDER, SPICE and BERT Score, respectively. Unfiltered Results for B1, B4, M, R-L, C, S, BERTScore are: 18.5, 3.3, 9.0, 19.9, 24.2, 12.4, 77.1

	B-1	B-2	B-3	B-4	M	R-L	C	S	BS
PJ-X [11]	21.8	11.0	5.9	3.4	16.4	20.5	19.0	4.5	78.4
FME [15]	23.0	12.5	7.2	4.4	**17.3**	22.7	27.7	**24.2**	79.4
RVT [9]	18.0	10.2	6.0	3.8	11.2	21.9	30.1	11.7	78.9
QA-only [4]	18.0	10.2	6.0	3.8	11.2	22.0	30.6	11.6	78.9
e-UG [4]	20.7	11.6	6.9	4.3	11.8	22.5	32.7	12.6	79.0
NLX-GPT	**24.7**	**15.0**	**9.6**	**6.6**	12.2	**26.4**	**46.9**	18.8	**80.3**

we initialize them from the Distilled GPT-2 weights and use a batch size of 64. At inference, we use greedy decoding. The maximum sequence length for VQA-X, ACT-X and e-SNLI-VE is set to 40, 30, and 40, respectively. For e-SNLI-VE, the maximum number of concepts fed at the input is 15.

2. VCR Setup and Experiments

Visual Commensense Reasoning (VCR) [16] is a new task introduced in which a model is given an image, question and a list of objects (regional features and bounding boxes) and is required to select one answer from a set of multiple-choice answers ($Q \rightarrow A$). After that, it is required to select a rationale (explanation) of *why* the answer it has selected is correct, from a set of multiple-choice rationales ($QA \rightarrow R$). The dataset consists of 290K samples of questions, answers and rationales. For the purpose of the NLE task, we follow previous NLE models [4, 9] and reformulate the explanation as a text generation task rather than a multiple-choice answering task. The train, validation and test splits for NLE are 191.6k, 21.3k, and 26.5k, respectively. Different from previous NLE vision-language tasks (VQA-X and e-SNLI-VE) where the input is an image and a question/hypothesis, VCR requires an additional input (detected objects) which can be represented by the regional features. Given the region proposal coordinates, NLE models implementing the re-formulated VCR [4, 9] first extract these regional features by performing Region-of-Interest (ROI) pooling or ROI Align on the output of a Faster R-CNN network [13]. Let a bounding box for a specific object be represented by x_1, y_1, x_2, y_2 which indicates the top-left and bottom-right coordinates. One approach we could take to represent objects for our NLX-GPT is to also perform ROI pooling on the grid-based vision backbone output. In the case of the vision transformer, we can reshape the output of shape $Y \times D$ to $H' \times W' \times D$, where $H' = H/P$ and $W' = W/P$ and D is the output dimension. After that, we can perform ROI pooling on that reshaped output using the given bounding box coordinates. However, we take a simpler approach. We first represent each bounding box with 8 values: $(\frac{x_1}{W}, \frac{y_1}{H}, \frac{x_2}{W}, \frac{y_2}{H}, \frac{x_1+x_2}{2W}, \frac{y_1+y_2}{2H}, \frac{x_2-x_1}{W}, \frac{y_2-y_1}{H})$, which are then projected to a high-dimensional representation equal to the dimension of the word and positional embeddings, in order to represent the object positional information. Since the question, answer or explanation may refer to specific detected objects in the image (e.g., *what are **person1** and **person3** doing?*, it becomes necessary to encode these objects along with their respective reference number. We therefore input to the Distilled GPT-2 tokens which consist of the objects, question, answer and explanation (as a single sequence). We embed these input tokens with a token embedding layer. We also embed the object reference number (ORN) with the same token embedding layer. We use `<no j>` (representing "no object") to represent the ORN for the question, answer and explanation tokens. Finally, we add the object positional information, token embeddings, ORN embeddings, positional embedding and segment embeddings together to form the Distilled GPT-2 input. The maximum number of objects is set to 20, and the maximum length of the (question, answer, explanation) sequence is set to 60. Figure 1 illustrates the complete process. For the purpose of our NLX-GPT, we also formulate the answer prediction as a text generation task. Unlike previous tasks discussed in the main paper (VQA-X, ACT-X and e-SNLI-VE) where the answer consists of one or a maximum of two words, the answer in VCR is usually much longer. It is therefore difficult to expect an identical correspondence between the generated and ground-truth answer. For example, the model may generate an answer: *no, this person does not live in this house* while the ground-truth answer is: *no, this person is a visitor*. In fact, this justifies why the authors of [16] formulated the VCR task as a multiple-choice task. At evaluation, only the test samples for which the predicted answer is correct are allowed to proceed to the second stage of providing the rationale ($QA \rightarrow R$), and thus test samples with wrong predicted answers should be filtered. To overcome the difficulty, we measure the context and semantic meaning of our predicted answers through the BERTScore [17] metric. We thus consider a predicted answer to be correct if its BERTScore referenced with its corresponding ground-truth is higher than or equal to 0.92. Table 1 shows our filtered scores and Figure 2 shows quali-

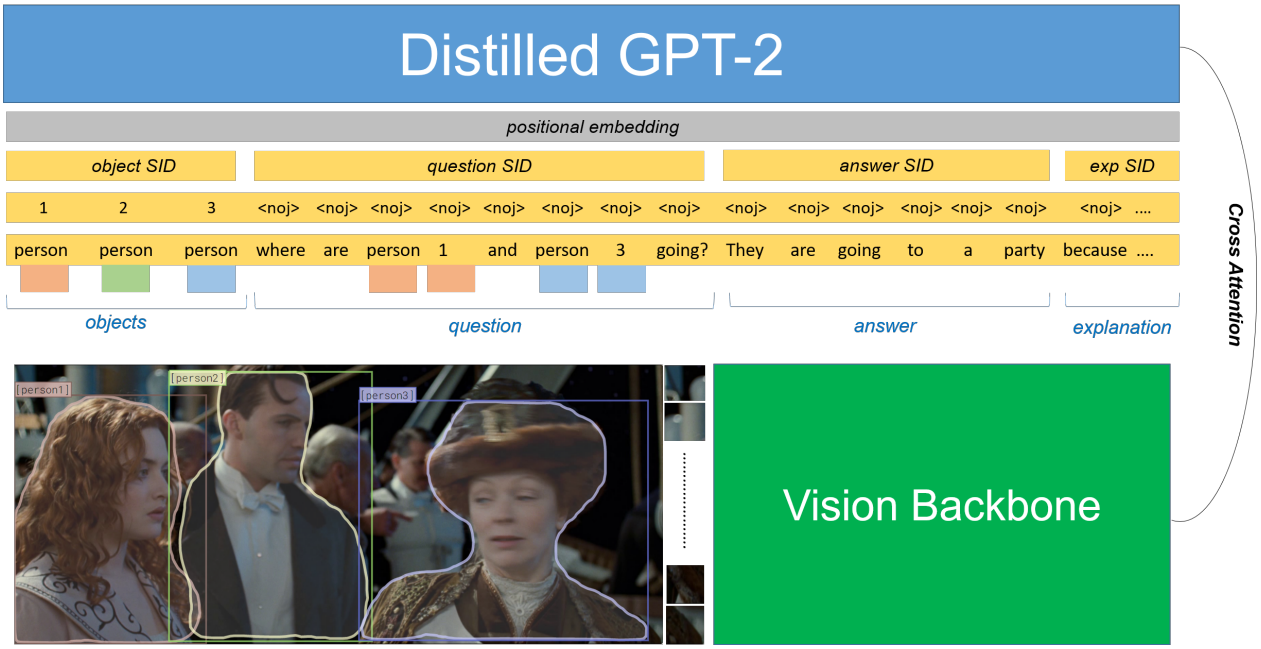


Figure 1. Our proposd NLX-GPT for the VCR Task. The word embedding, object reference number embedding as well as the segment ID embeddings all share the same layer (orange). The red, green and blue squares represent the projections of the bounding box coordinates for person 1,2 and 3, respectively.

tative examples.

3. Explain-Predict Implementation Details

The explain-predict evaluation framework is trained on the ground-truth explanations. That is, we feed the question and ground-truth explanation to the Distilled-BERT model during training. We initilze the model with the Distilled-BERT weights[2]. The model is trained with the ADAM optimizer with a batch size of 16 and a learning rate of 2e-5 which is linearly decayed to 0 over the total number of training steps. For VQA-X and ACT-X, we train a multi-label classifier with soft targets over the ground-truth answers using binary cross-entropy loss. For e-SNLI-VE, we train a a classifier with the hard targets over the ground-truth answers using cross-entropy loss. We measure the accuracy as the evaluation criteria. It is important to note that for VQA-X, we find a total of 37 test samples for which their answers are never seen in the training set. For other NLE models [4, 11, 15], that is not a problem since they employ a pretrained VQA model (trained on the full VQAv2 dataset). However, our NLX-GPT is trained from scratch and only on the VQA-X dataset (which is much smaller than VQAv2). We therefore exclude these 37 samples from the explain-predict accuracy calculation.

[2]https://github.com/huggingface/transformers

4. More Qualitative Examples

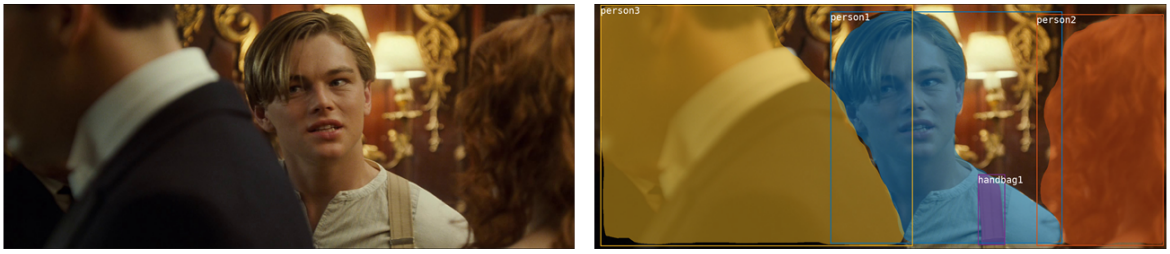
We include more qualitative examples from VQA-X, ACT-X and e-SNLI-VE in Figures 3, 4, and 5. Figure 6 depicts the retrieval-based attack evaluation results visually for two test samples from the ACT-X dataset when $K = 5$. As shown, very similar images have a low intra-distance, and thus our model has a low susceptibility to correlations and bias in the dataset.

5. Human Evaluation Process

For VQA-X and e-SNLI-VE, the human evaluation process is identical to [4]. We randomly select 300 test samples with correctly predicted answers. The evaluation is performed by 3 different annotaters and the results are then averaged. The annotaters mainly have to select one out of 4 choices (yes, weak yes, weak no, and no) as a response for whether the explanation justifies the answer. The 4 scores are numerically mapped to 1, 2/3 , 1/3 , and 0, respectively. The numerical scores are then averaged among all test samples to get a final score. For ACT-X, we follow the main procedure introduced in the paper [11]. 300 test samples with correctly predicted answers are randomly chosen, and a human annotater is asked to determine whether a generated explanation is superior to, inferior to, or equal to the ground truth explanation. The percentage of the generated explanations which are equal to or superior to the ground truth explanations are reported.

what era are person1 and person2 from?

they are from the nineteen fifties *because* *the style of clothing is the style of the time period*



who is person1 talking to?

person1 is talking to person3 *because* *person1 is looking at person3 and is speaking to him*

what is person2 going to do in the store?

person2 is going to buy some food *because* *person2 is standing in front of a counter with a basket of food in front of him*

where are person3 and person2 right now?

they are in a hospital *because* *there are hospital beds*

Figure 2. Qualitative examples from our model on the VCR task

Should the cars be stopped at this light?
yes because the light is red

Does the printer have paper?
yes because there is a white piece of paper on the top of the printer

What is this person riding?
skateboard because the main is riding on a board with four wheels

Figure 3. More qualitative examples from our model on the VQA-X task

water walking because he is inside of a bubble and is moving across the top of it

carpentry because he is standing in a workshop and working on a piece of wood

swimming because he is in a pool and is moving his arms and legs

Figure 4. More qualitative examples from our model on the ACT-X task

6. Failure Cases

We include failure cases of our model in Figure 7. We realize that in some cases, the answer is predicted wrong but the explanation is correct or vice-versa. For cases where the explanation is wrong but the answer is correct, we hypothesize the problem to be in the equal weighing of all words to be generated (including the answer), which treats the answer as all other words. To alleviate this problem, we tried several solutions. One solution is to assign more weight to any word in the answer vocabulary during the loss calculation, so that wrong wrongly predicted answers incur more loss. Unfortunately, none of the solutions we tried gave positive effects. For e-SNLI-VE, we observe two major failure cases. Firstly, the model justifies the prediction by simply repeating the hypothesis. Secondly, the model is sometimes biased towards the tone the human explainer justifies the prediction, as many samples in the dataset are of the form: (.....*is the same as*.....) or (*just because*.....*does not mean*.....).

7. Removing a VL-model is advantageous

In the paper we give two intuitions behind this: In short, 1) it eliminates the high memory requirements of the VL-model and reduces the inference time. 2) it eliminates the independence and dissociation of the VL-model and explanation model, in the sense that the explanation is intrinsic, internally affiliated and connected to the reasoning process made to predict the answer. There are other reasons as well: Training the task and explanation jointly in one model allows us to have faithful explanations, which is what FME [15] discusses, however in a much simpler and model-intrinsic way, and without any external operations. Also, if we had a separate VL-model and a separate explanation model, finetuning the VL-model along with the explanation model is advantageous, but a difficult step (in most works, this step is avoided) due to the extra memory requirements to fine-tune the VL-model as well as the careful and correct consideration of hyperparameters and optimization procedure required. In our work, this is completely avoided since both are jointly trained in one model.

Hypothesis: The dogs are outside.
Detected Concepts: [grass, green, tree, black, white, mouth, leg, trees, brown, ear, running, tail, horse, dog, field]
Prediction: ***entailment*** *because grass is outside*

Hypothesis: The man is grilling hotdogs for the people during lunch hour.
Detected Concepts: [man, shirt, hand, black, white, person, hair, red, orange, shorts, arm, head, pants]
Prediction: *neutral because just because a man is grilling hotdogs does not mean it is during lunch hour*

Hypothesis: Older women are reading books about art
Detected Concepts: [woman, white, shirt, blue, table, black, person, paper, book, wall, glasses, lady, red, brown, booklet]
Prediction: ***neutral*** *because not all books are about art*

Figure 5. More qualitative examples from our model on the e-SNLI-VE task

Figure 6. Retrieval-based attack evaluation results visually for two test samples from the ACT-X dataset

References

[1] Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *ArXiv*, abs/1607.06450, 2016. 1

[2] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 1

[3] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:664–676, 2017. 1

[4] Maxime Kayser, Oana-Maria Camburu, Leonard Salewski, Cornelius Emde, Virginie Do, Zeynep Akata, and Thomas Lukasiewicz. e-vil: A dataset and benchmark for natural language explanations in vision-language tasks. *ArXiv*, abs/2105.03761, 2021. 2, 3

[5] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015. 1

[6] Jonathan Krause, Justin Johnson, Ranjay Krishna, and Li Fei-Fei. A hierarchical approach for generating descriptive image paragraphs. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3337–3345, 2017. 1

[7] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123:32–73, 2016. 1

[8] Tsung-Yi Lin, M. Maire, Serge J. Belongie, James Hays, P. Perona, D. Ramanan, Piotr Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 1

[9] Ana Marasović, Chandra Bhagavatula, Jae Sung Park, Ronan Le Bras, Noah A. Smith, and Yejin Choi. Natural language rationales with full-stack visual reasoning: From pixels to semantic frames to commonsense graphs. In *FINDINGS*, 2020. 2

[10] Vicente Ordonez, Girish Kulkarni, and Tamara L. Berg. Im2text: Describing images using 1 million captioned pho-

Question: What food is pictured?
Prediction: pizza because it is round and has cheese and sauce on it

Question: Is it daytime?
Prediction: yes because the man is standing in front of a kitchen stove

Question: Is it raining?
Prediction: no because the ground is wet and the sky is grey

Hypothesis: A blond woman in a yellow clown car
Prediction: contradiction because a woman can't be wearing a yellow shirt and a yellow clown car at the same time

Hypothesis: A man is reaching for something on the window
Prediction: entailment because a man is reaching for something on the window is the same as a man is reaching for something on the window

Hypothesis: A group of people look at a bridge.
Prediction: entailment because a group of people are looking at a bridge

Figure 7. Failure cases on the VQA-X and e-SNLI-VE tasks

tographs. In *NIPS*, 2011. 1

[11] Dong Huk Park, Lisa Anne Hendricks, Zeynep Akata, Anna Rohrbach, Bernt Schiele, Trevor Darrell, and Marcus Rohrbach. Multimodal explanations: Justifying decisions and pointing to the evidence. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8779–8788, 2018. 2, 3

[12] Bryan A. Plummer, Liwei Wang, C. Cervantes, Juan C. Caicedo, J. Hockenmaier, and S. Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. *International Journal of Computer Vision*, 123:74–93, 2015. 1

[13] Shaoqing Ren, Kaiming He, Ross B. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149, 2015. 2

[14] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*, 2018. 1

[15] Jialin Wu and Raymond J. Mooney. Faithful multimodal explanation for visual question answering. *ArXiv*, abs/1809.02805, 2019. 2, 3, 5

[16] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6713–6724, 2019. 2

[17] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. *ArXiv*, abs/1904.09675, 2020. 2