Natural Language Person Retrieval

Tao Zhou
University of California, Los Angeles
taozhou@cs.ucla.edu

Jie Yu
SAIC Innovation Center
jerry.j.yu@gmail.com

Abstract

Following the recent progress in image classification and image captioning using deep learning, we developed a novel person retrieval system using natural language, which to our knowledge is first of its kind. Our system employs a state-of-the-art deep learning based natural language object retrieval framework to detect and retrieve people in images. Quantitative experimental results show significant improvement over state-of-the-art methods for generic object retrieval. This line of research provides great advantages for searching large amounts of video surveillance footage and it can also be utilized in other domains, such as human-robot interaction.

[1]Video surveillance cameras are everywhere—in small stores and apartments for indoor scenario and in parking lots and traffic lanes for wide-area observation. With increasingly ubiquitous security cameras, the challenge is not acquiring surveillance data but automatically recognizing what is valuable in the video. Understanding content from video alone, however, is extremely challenging due to factors such as low resolution, deformation, and occlusion. Therefore, it is highly desirable for a system to match objects of interest with a natural language description sentence. Here we employ a state-of-the-art deep learning framework (Hu et al. 2016, Hu, Rohrbach, and Darrell 2016) to retrieve people.

The first challenge of our project is the lack of a dataset for natural language person retrieval tasks. We turn to the Cityscapes dataset (Cordts et al. 2016), a large-scale benchmark dataset for pixel-level and instance-level semantic labeling. Since the focus of our project is on person retrieval rather than semantic segmentation, only segmentation masks belonging to 'person' and 'rider' categories are transformed into ground truth bounding boxes based on the masks' maximum and minimum value of (x, y) coordinates. Specifically, the (x_{MAX}, y_{MAX}) location is treated as the bottom-right corner of the bounding box while the (x_{MIN}, y_{MIN}) location is treated as the top-left corner. To avoid small persons, bounding box size larger than 5000 are selected for further annotation via Amazon Mechanical Turk (AMT). Given a person inside a bounding box, the AMT workers need to describe the person and select attributes best matching the appearance.

The region proposal network (RPN) in Faster R-CNN (Ren et al. 2015) is adopted to generate dozens bounding boxes with different confidence which might contain a person. The higher the confidence, the more likely it is for the bounding box to contain a person (Figure 1A). Since most bounding boxes with low confidence do not include a person's entire body, the bounding boxes are filtered by setting the threshold of the confidence to 0.5. Additionally, the minimum size of the bounding box is set to 5,000 in order to avoid small persons (Figure 1B). Due to the limited number of persons, the dataset is augmented for training purposes by randomly selecting 3 shifted region proposals whose IOU with ground truth bounding boxes are larger than 0.5 (Figure 1C). The region proposals without augmentation (Figure 1B) and a description "An elderly man on the right riding a bike" are provided as input to the model for person retrieval (green region proposal, Figure 1D, E). The ground truth bounding box is shown in red.

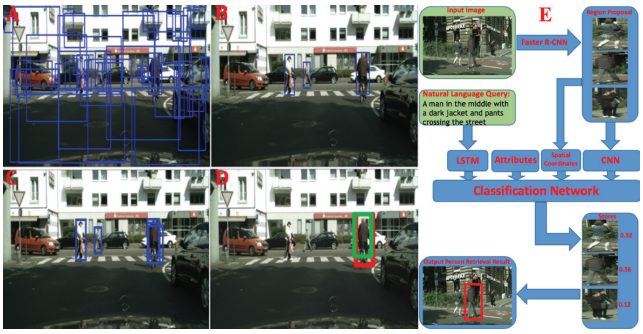

Figure 1. Procedures on region proposals generation and overview of natural language person retrieval framework.

During the training phase, a positive training instance is comprised of one region proposal, the spatial configuration, its corresponding description and the label (true)

while a negative training one includes a region proposal, the spatial configuration, an unrelated description and the label (false). Each training batch contains 150 samples with positive and negative training samples randomly shuffled. The model takes as input the region proposals, and output 1,000-dimensional features by first resizing them to 224 × 224 and then extracting visual features from the resized cropped image using a VGG-16 network with batch normalization (Simonyan and Zisserman 2014, Ioffe and Szegedy 2015) pretrained on the ILSVRC classification task. For the natural language description on an image region, each word is embedded into a vector through a word embedding matrix, and then use a recurrent Long-Short Term Memory (LSTM) (Hochreiter and Schmidhuber 1997) network with 1000 dimensional hidden state to scan through the embedded word sequence. After the LSTM network have seen the whole text sequence, we max pooling all the hidden state in the network as the encoded vector representation of the expression. Besides, relative coordinates of the region proposal are applied to reason about its spatial position in the image such as 'man in the middle'. The relative center and the relative length of the width and height of the region proposal are also incorporated. Thus, the 8-dimensional spatial coordinates are $[x_{r\text{-}min}, y_{r\text{-}min}, x_{r\text{-}max}, y_{r\text{-}max}, x_{r\text{-}center}, y_{r\text{-}center}, w_{r\text{-}box}, h_{r\text{-}box}]$, where r means relative quantity. The text feature (1,000D), visual feature (1,000D) and spatial coordinates (8D) are concatenated as input to a multilayer perceptron classifier which outputs scores for region proposals. The loss function during training is defined as the average loss over instances:

$$\text{Loss} = \frac{1}{N}\sum_{i=1}^{N} L(v_i, T_i)$$

where N is the total number of training instances, v_i is the score for the ith region proposal and T_i is its corresponding label, L is the sigmoid cross entropy loss as follows:
$-T_i * \log(\text{sigmoid}(v_i)) - (1 - T_i) * \log(1 - \text{sigmoid}(v_i))$.

During the retrieval phase, an image containing several people and a phrase description is introduced as input (green blocks in Figure 1E). The target person is finally retrieved based on the rank of the scores from the classification network.

Overall, our Natural Language Person Retrieval framework, which is based on a UC Berkeley model, leads to a 10% increase as compared to random selection, as shown in Table 1. In fact, the model pretrained on ReferIt and tested on CITYSCAPES (column 3) is even worse than random selection (column 2), while the model trained on CITYSCAPES increased the accuracy by 5% (column 4). Except for modifying the batch size for improvement, batch normalization (BN) was also employed. It is not surprising that VGG with BN leads to a 2.3% increases; however, LSTM with BN deteriorates by 8% even with careful initialization (Cooijmans et al. 2016). This might be due to the non-logic dependence nature of our description expression. Indeed, one can describe the upper body first, then the lower body, and vise versa. In this regard, LSTM cannot predict the next word based on the previous phrase. Thus, instead of predicting the next word (Hu, Rohrbach, and Darrell 2016), we model sentence semantics by max pooling across all hidden states. This adjustment boosts the performance by 1.5%.

	Random	UCBerkeley	UCBerkeley	Ours
Trained on		ReferIt	CITYSCAPES	CITYSCAPES
Tested on	CITYSCAPES	CITYSCAPES	CITYSCAPES	CITYSCAPES
REC@1	38%	37.3%	43.7%	48.8%
REC@2	60%	59.7%	65.3%	70.5%

Table 1. Performance of our method compared with random selection in CITYSCAPES dataset. "Rec@1" is the recall of the highest scoring region proposal (the percentage of the highest scoring region proposals being correct), and "Rec@2" is the percentage of at least one of the top 2 highest scoring proposals being correct.

In summary, we presented what is to our knowledge the first natural language person retrieval system. A large-scale benchmark dataset was constructed using crowdsourcing. A new deep-learning-based framework was furthermore designed to match visual and textual representations. Comparing to the state-of-the-art object retrieval method, a substantial increase in performance was observed due to our novel end-to-end training system, the introduction of batch normalization on VGG, and max pooling on LSTM.

References

Cooijmans, T., Ballas, N., Laurent, C., Gülçehre, Ç., & Courville, A. 2016. Recurrent Batch Normalization. arXiv preprint arXiv: 1603.09025.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Schiele, B. 2016. The Cityscapes Dataset for Semantic Urban Scene Understanding. In CVPR.

Hochreiter, S., & Schmidhuber, J. 1997. Long Short-Term Memory. *Neural Comput.*, *9*(8), 1735–1780.

Hu, R., Rohrbach, M., & Darrell, T. 2016. Segmentation from Natural Language Expressions. In ECCV.

Hu, R., Xu, H., Rohrbach, M., Feng, J., Saenko, K., & Darrell, T. 2015. Natural Language Object Retrieval. In CVPR.

Ioffe, S., & Szegedy, C. 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. arXiv preprint arXiv:1502.03167

Ren, S., He, K., Girshick, R., & Sun, J. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In CVPR.

Simonyan, K., & Zisserman, A. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. In CVPR.