Spot It: Translating Differences between Paired Pictures

Jinlin Fan, Qingsong Wang

June 21, 2019

1 Introduction

Deep learning is a popular framework with lots of applications in a wide range of fields. Deep learning methods are usually based on many flexible networks constructed by unit neurons. Of course, they get many breakthroughs in many tasks where needs interaction between human and electronic devices, which significantly increase the efficiency in handling those tasks. Especially, in computer vision(CV) and natural language processing(NLP), which are two biggest branches in deep learning tasks, lots of deep learning networks have been deployed and bring us quite impressive excellent performences, such as Auto-driving, AlphaGo, OpenAI dota2, Robots from Boston dynamics.

There are many methods and network structures in both CV and NLP. We human can simultaneously handle these two tasks, in other words, get and analyze visual and linguistic information by resorting to our brains. However, this ability is a challenge for machines or algorithms. Which cases need machine to handle visual and linguistic information at the same time?

Here is a toy example, which can show us the road from easy problems to hard problems, in three different stages.

Stage 1: When we see a picture, we can tell out what objects are in this picture. **Stage 2:** Furthermore, with several seconds for obseving, we can describe the relationship between objects in picture, including space relationship, direction relationship, classification relationship and so on. **Stage 3:** What is next? If we find out a man and a boy looks similar, we may infer that this man is father of this boy. We should say, we can get information which is not shown by the picture, which bring us to a inferential task.

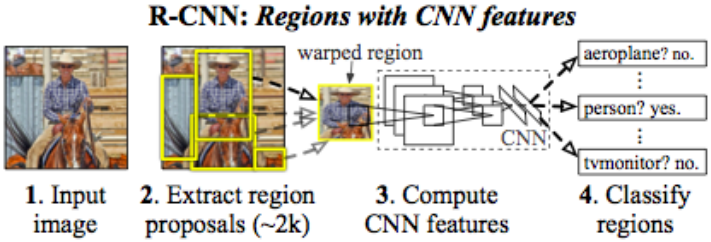
1.1 Image2text

In recent years, many researchers focus on these problems, building a birdge connecting CV and NLP.

At the beginning, when it comes to image classification and the image only includes one object, if all the classes have characteristic tags, it is actually a very naive kind of "translation" or "interpretation" of images. This task is one kind of image semantic analysis.

This try is also a start of Image2text, followed by image detection. Image detection task admits several objects appear in one picture, after feeding in a well-trained network, we can get all the tags of objects in image.

A good detection method is Region-based CNN(R-CNN), which is introduced by Jeff Donahue and Ross Girshick. R-CNN divide detection task into two seperate parts, one is region segmentation based on selective search or other methods, the other is object recognition which can be done resorting to CNN and multiple-class classifiers.



But we are still at **stage 1**, because machines can't explain or introduce relationship between objects in one image. However, we are close to **stage 2** somehow. The machine already has ability to detect all important objects in image, we need to teach it how to combine objects with some basic relationship rules.

1.2 Image2sentence

In this subsection, we will go through some typical methods for Image2sectence, which means translating images to sentence in human language.

Two kinds of methods are developed recently.

The first kind is following the idea we discussed at the end of last subsection, called pipeline method. Using detected objects, with additional language model trying to combine them, we can get a sentence. At last, we need to rank different sentences in output by reasonability. After all the steps, the top 1 ranking sentence will be most preferable among all the other. Microsoft has some contributions on such procedure. Pipeline methods consist

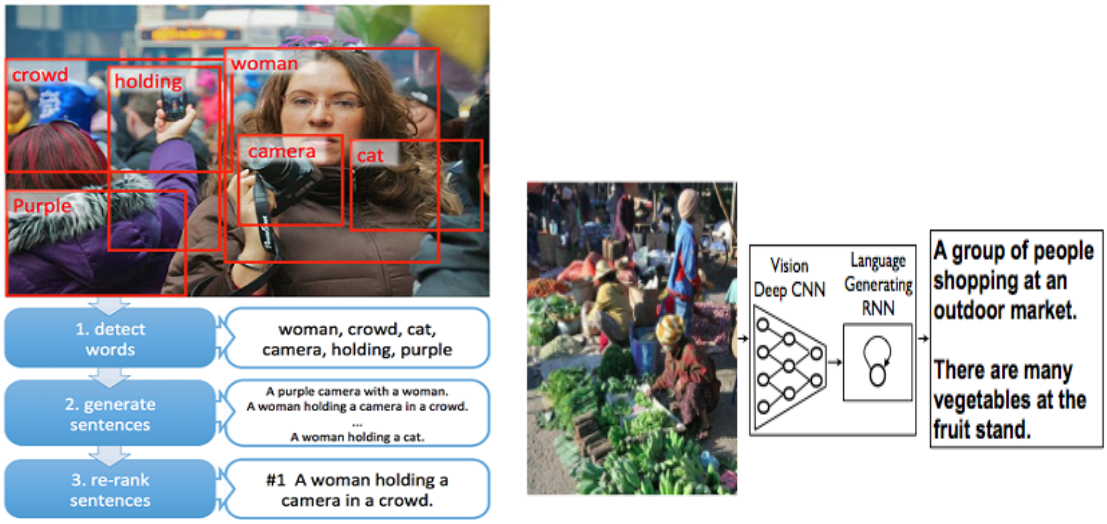


Figure 1: The left part is an example of pipeline method; the right part is an example of end-to-end method.

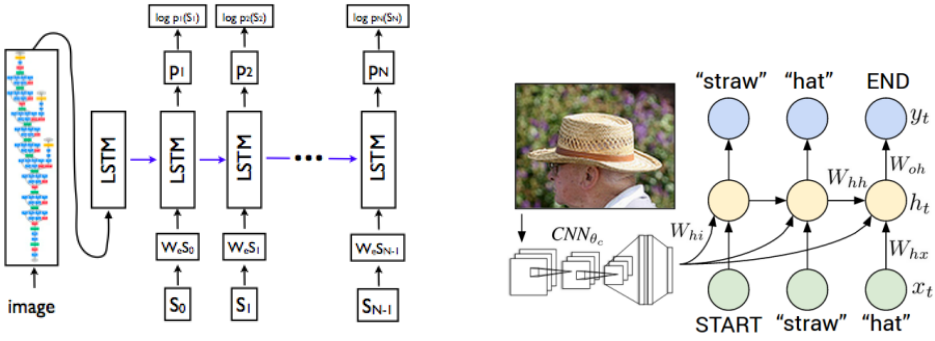


Figure 2: The network struture of ICG and Li-feifei's work.

of seperate parts and each part can be regarded as a independent structure, which accepts proper adjustment training.

The other kind of method is end-to-end method, which is more straightforward but needs more complicated network structure. Google developed Image Caption Generator based on combination of RNN and CNN. In machine translation tasks, we need a encoder 'RNN' to project information in original language sentence on a fixed dimension embedding space and a decoder 'RNN' to transform the embedded vector to target language sentence. Similarly, we need to translate a image into english now. Hence CNN, which is flexible enough to extract hierarchical information in image, substitute RNN as a new encoder part. Li Feifei also did similar work, which is called Visual-semantic alignments.

2 Spotting the Differences in Pictures

2.1 Basic Structure of Network

Until now, we can handle tasks in **Stage 2**. Some approaches can help machine to translate images into language, are the translated sentences comparable? Or does the transformation from image to sentence have some kind of continuity property.

In many cases, people need to compare different images, or to find out the differences in multiple images. For example, compare different but similar commondities or counterfeit detection. These tasks need structured network to spot the differences and amplified the different regions. To handle this problem, we give out our insights and designs of working network.

Because this problem also needs a translation procedure, similar to many state-of-the-art methods, maximizing the probability of the correct ranslation given an input in an end-to-end framework,

$$\theta^* = \arg\max_{\theta} \sum_{I,S} logp(S|I;\theta)$$

where θ are the parameters in our network, I is an image and S is corresponding target sentence. Because of the sequential structure, by the chain rule of conditional distribution,

$$logp(S|I,\theta) = \sum_{t}^{N} logp(S_t|I, S_0, ..., S_{t-1}, \theta)$$

It's natrual to model $p(S_t|I, S_0, ..., S_{t-1})$ with a sequnetial model such as RNN. It can take advantage of information contained in the preceding $t-1$ words, which is regarded as 'memory' of a network. This memory has a updating rule, computed by preceding memory and current state input,

$$h_t = f(h_{t-1}, x_t).$$

For the choice of RNN, we choose LSTM to handle sequential to sequential task. To compress rich information in images, we utilize CNN, especially ResNet to learn the mapping from images to embedding space.

The choice of f in updating memory is governed by its ability to deal with vanishing and exploding gradients issues, which usually happen in RNN and CNN training. This is also another motivation to choose LSTM. What's more, LSTM model has a core cell encoding knowledge at every time step of historical observation.

2.2 Difference Extraction

In this project, the dataset includes three parts, target images, reference images and difference captions. The captions are sentences descibing the differences in refernce images from respective target images.

Figure 3: Here are two pairs of original images and difference images

The core part of this work is 'How to tell the machine where the different is in the pictures'. If we input the original target images and reference images, by experiments, it is inefficient to collect information about differences for images. Thus, we constrcut a new 'image1' by combining two images horizontally and then change their position to get 'image2', finally, take the difference of 'image1' and 'image2'. After changing like this, the differences between two images are amplified and be doubled, both appearing in the left and right regions, which is a benefitial feed for training the network.

2.3 Training

For the training procedure, we use pretrained ResNet with trainable parameters and set batchsize to be 5, 50 epoches, learning rate to be 0.0001 with Adam optimizer. We choose LSTM with 0.1 dropout proportion for the sequential structure, of course, GRU is also available. We first input a image into the ResNet to get a 1024-dimension features, after transforming by a linear layer with batch-normalization and relu activation, its dimension is reduced to 512. Then put this feature in RNN with properly embedded captions. We also output some sampled sentences to verify the reasonability of our networks, and keep watch on the loss on validation set in avoid of overfitting by early stopping. In Figure 4, we have plots of loss function on training data and validation data, we use early stopping strategy to avoid overfitting and choose the stopping point by a heuristic method.

3 Conclusion

We now give out some of our outputs, from these sentences in our result we can find out that, in some cases, our network can only focus on some obvious and big differences, some tiny differences cannot be spotted. Here are some examples of predictions in Figure 5. In this figure, we can see that all the sample output are partially correct and some descriptions in details disappear. And in Figure 6, we can find out that our network sometimes can only detect very grant differences and ignore many local information.

For further improvememt of our networks, we give out several comments based on our experience on this project.

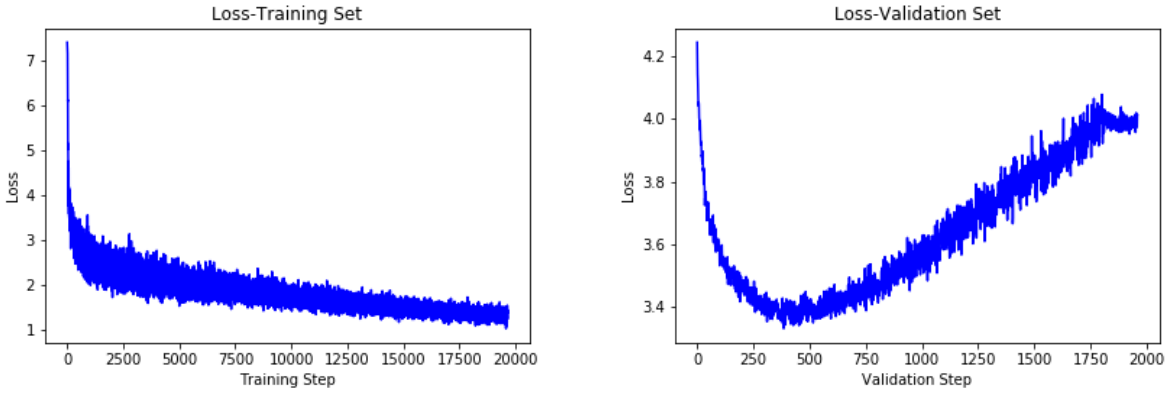


Figure 4: Traing loss is on the left; Validation loss is on the right

Figure 5: Some samples with reasonable captions.

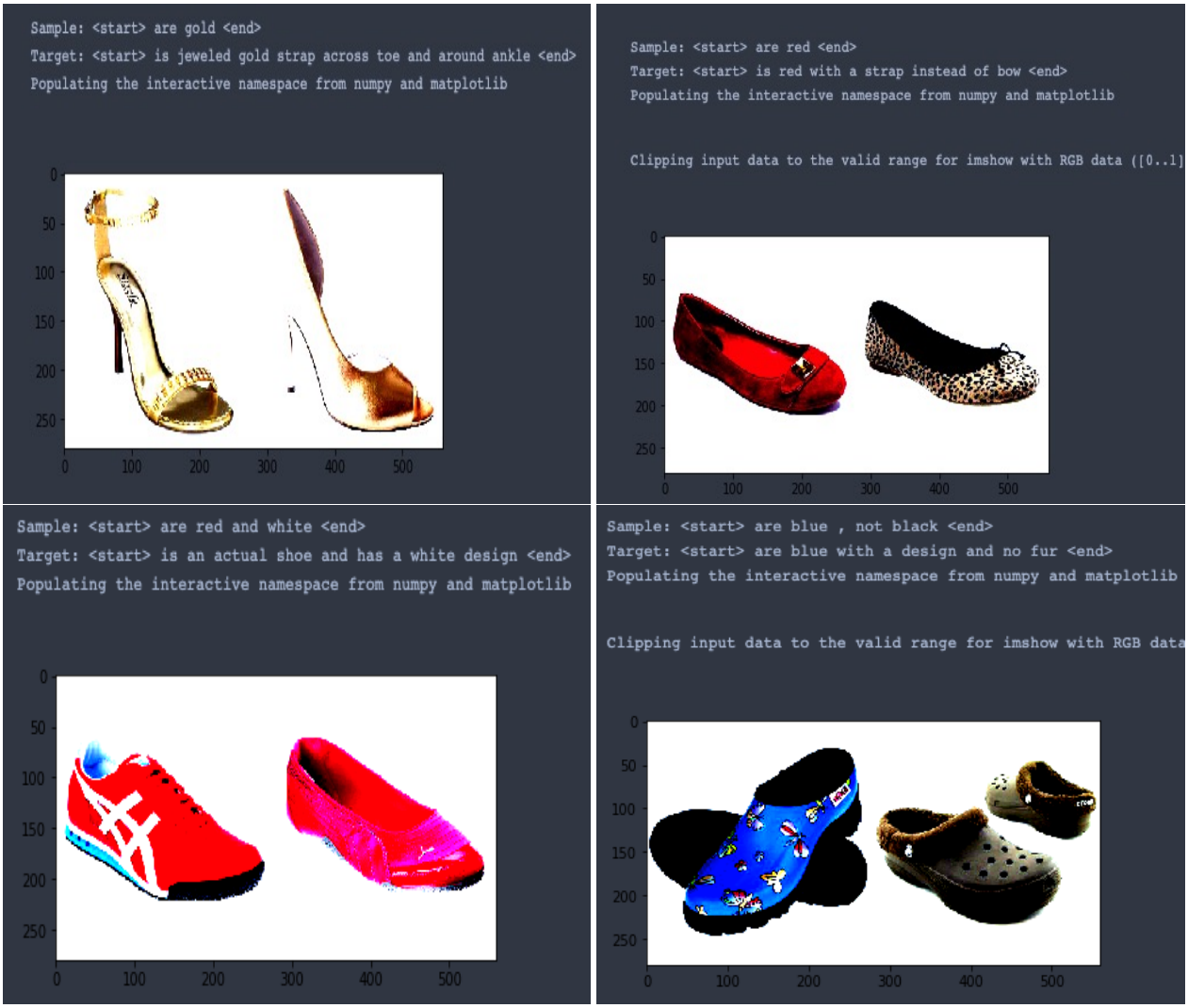


Figure 6: Some samples with captions ignoring local information.

Firsly, a good procedure for data augmentaion will help a lot. We notice that the difference in captions are static and invariant when the target images and reference images are rotated or enlarged. And we think until now, we do not have enough images to cover all the key words for spotting the differences. Hence, data augmentaion can give more samples as well as more captions, which will help network extract features efficiently.

Secondly, we find out that our network cannot handle the local differences accurately, which may be due to the large scale of images and effects of some global differences. This lead us to some local methods, such as attention-based network. We insist if attention mechanism are used on the CNN-encoder, more local features will be extracted, combined with data augmentation, the network will spot more differences accurately in multiple aspects.

Finally, our method is an end-to-end method, which may not suitabe to do ranking procedured for outputted sentences. And we don't do any comparisons with other state-of-art methods aimming for similar tasks, which will not give strong evidence for the advantages of our method in this project.

4 References

[1] Kulkarni G, Premraj V, Ordonez V, et al. Babytalk: understanding and generating simple image descriptions.[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2013, 35(12):2891-2903.

[2] Karpathy A , Fei-Fei L . [IEEE 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) - Boston, MA, USA (2015.6.7-2015.6.12)] 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) - Deep visual-semantic alignments for generating image descriptions[J]. 2015:3128-3137.

[3]Xu K, Ba J, Kiros R, et al. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention[J]. Computer Science, 2015:2048-2057.

[4] Vinyals O, Toshev A, Bengio S, et al. Show and tell: A neural image caption generator[C]// IEEE Conference on Computer Vision & Pattern Recognition. 2015.