|  |
| --- |
| **PA4: Image Captioning with RNN-based Neural Network** |

**Hao XiangHuimin Zeng**

Computer Science & Engineering Computer Science & Engineering

UC San Diego UC San Diego

La Jolla, CA 92093 La Jolla, CA 92093

*haxiang@ucsd.edu zenghuimin@yahoo.com*

**Xingyi YangZhenrui Yue**

Computer Science & Engineering Computer Science & Engineering

UC San Diego UC San Diego

La Jolla, CA 92093 La Jolla, CA 92093

*x3yang@ucsd.edu yuezrhb@gmail.com*

**Abstract**

This is the report of programming assignment 4 in CSE 253. Inspired by the Encoder-Decoder architecture for image captioning [1], we implemented an image captioning neural network with a CNN encoder based on ResNet-50 [2] and an RNN decoder based on LSTM and Vanilla RNN. We tested the two neural networks with different hyper-parameters on the COCO dataset [3]. Furthermore, we implemented two ways of caption generation, deterministic and stochastic, as well as the utilization of pretrained word embeddings with word2vec [4]. Finally, we visualized the training process with training loss & validation loss and evaluated the mode performances using BLEU-1 and BLEU-4 scores, discussed the advantages and reasons for different approaches.

**1 LSTM and Vanilla RNN Model Structure**

Where represents the batch mean, the standard deviation of the batch and a very small value in case of 0 batch standard deviation in denominator. Applying batch normalization to each batch, the normalized input could be transformed into data with zero mean and unit variance. With normalized data, the network weights no more depend on the range of the input data, and the model training process become less reliable on the initialization of the network weights, with the model itself more generalized and capable of predicting on equally preprocessed data.

**2 Evaluation of LSTM and Vanilla RNN**

In this section, we present a few related models and published work in the field of computer vision and semantic segmentation, based on which we have created modified models or reproduced the identical models for our task. These models and implementation methods will be introduced in the following subsections.

**3 Deterministic Caption Generation**

A few convolutional neural network models for semantic segmentation have been created and evaluated on the CityScapes dataset, these model structures and their implementation will be introduced in the following parts. Please note that in all of our models, we utilized the cross-entropy objective function, Adam optimization method, initialized the weights with Xavier initialization.

**4 Stochastic Caption Generation**

This part included the training, validation and test results of the models we mentioned in the last section. For each mode, we present a single plot of the training process with training and validation loss curves, validation pixel accuracy, average IoU and IoUs of five classes: building (11), traffic sign (20), person (24), car (26) and bicycle (33). Please see below for average IoU values and for each of the five classes. The lower part of table 5 shows the result comparison of different loss functions (dice loss and weighted loss) and the last for augmented images modified from the baseline fully convolutional networks.

**5 Utilization of Pretrained Embedding**

Concerning a semantic segmentation task, we only implemented Fully Convolutional Networks to extract the visual features from the images. The classification is directly conducted using an extra convolutional layer. That is, no fully connected layers were used in any of our models. Moreover, given the knowledge that the performance of neural networks largely depends on the capacity of themselves. Therefore, we built several neural networks with different architectures and different depths. For instance, the skip connection architectures were realized via Residual Learning (ResNet) structure and U-Net structure. In addition, for a better comparison, we choose ResNet50 and ResNet101 as backbone. We observed that the performance could be greatly improved with the growth of the depth of the model.

**6 Individual Contributions**

This part included the individual contributions of the two team members Hao Xiang, Huimin Zeng, Xingyi Yang and Zhenrui Yue in the programming assignment.

**6.1 Hao Xiang**

Hao Xiang implemented models including U-Net, ResNet + ASPP, U-Net + ASPP and build the pipeline for train.py as well as the evaluation metrics including IOU calculation, accuracy calculation, as well as contribute to the documentation of the code and README file for the repository. Also, Hao contributes to the discussion, results and methods section in the assignment report.

**6.2 Huimin Zeng**

Huimin Zeng implemented the first version of the baseline model and the function loading other pretrained models, data loader, visualizer, and plot function for saving the training curves and other utility functions. I also ran some experiments to fine tune the hyperparameters.

**6.3 Xingyi Yang**

Xingyi Yang completes all the experimental framework. He also trains and conducts the comparison analysis for different loss function and various network topology. He manages all model training and evaluate their performance. In addition, Xingyi finishes the data augmentation method like flip, rotation, random crop.

**6.4 Zhenrui Yue**

Zhenrui was responsible for implementing and training the baseline FCN model, he also conducted literature review on relevant CNN models such as VGG, ResNet, U-Net and DeepLab. Zhenrui also reviewed the models and completes most sections of the report including introduction, related work, models, and discussion.

**References**

1. Liang-Chieh Chen et al. “Rethinking atrous convolution for semantic image segmentation”. In:arXiv preprint arXiv:1706.05587(2017).
2. Marius Cordts et al. “The cityscapes dataset for semantic urban scene understanding”. In:Proceedings of the IEEE conference on computer vision and pattern recognition. 2016, pp. 3213–3223.
3. Karen Simonyan and Andrew Zisserman. “Very deep convolutional networks for large-scaleimage recognition”. In:arXiv preprint arXiv:1409.1556(2014).
4. Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “Imagenet classification with deepconvolutional neural networks”. In: (2012), pp. 1097–1105.
5. Kaiming He et al. “Deep residual learning for image recognition”. In: (2016), pp. 770–778.
6. Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-net: Convolutional networks forbiomedical image segmentation”. In: (2015), pp. 234–241.
7. Liang-Chieh Chen et al. “Semantic image segmentation with deep convolutional nets and fullyconnected crfs”. In:arXiv preprint arXiv:1412.7062(2014).
8. Liang-Chieh Chen et al. “Deeplab: Semantic image segmentation with deep convolutional nets,atrous convolution, and fully connected crfs”. In:IEEE transactions on pattern analysis andmachine intelligence40.4 (2017), pp. 834–84.