Kaggle Data Science Bowl 2017 Technical Report

qfpxfd Team

May 11, 2017

1 Team Members

Table 1: Team members

Name	E-Mail	University
Jia Ding	dingjia@pku.edu.cn	Peking University, Beijing, China
Aoxue Li	lax@pku.edu.cn	Peking University, Beijing, China
Zhiqiang Hu	huzq@pku.edu.cn	Peking University, Beijing, China
Dong Wang	wangdongcis@pku.edu.cn	Peking University, Beijing, China
Jiayuan Gu	gujiayuan@pku.edu.cn	Peking University, Beijing, China
Jun Gao	steve_gao@pku.edu.cn	Peking University, Beijing, China
Liwei Wang	wanglw@cis.pku.edu.cn	Peking University, Beijing, China
Jiabao Liu	liujiabao0811@163.com	Capital Medical University, Beijing, China

2 Introduction

Our model is mainly based on Convolutional Neural Network, which detects nodule candidates, classifies malignancy and predicts cancer probability of scans. Due to the computational cost, we adopt a "two stage" algotithm for nodule detection. 2D Faster R-CNN is applied to detect nodule candidates with high recall, which is followed by 3D CNN to reduce false positives; For nodule malignancy classfication, two modified 3D CNNs are used. For the final output, a logistic regression method is utilized to merge intermediate results.

To implement our algorithm, we use Caffe for Faster R-CNN, Keras backended on Tensorflow for other CNNs. We utilize GPU for training CNN, with 3 K80 and 4 TitanX.

3 Data prepare and preprocessing

The public dataset we have used for training are LIDC-IDRI, DSB stage1, SPIE. There are two versions of data converted from raw dicom files, named as v9 and v15.

- For LIDC-IDRI, we follow the method of LUNA2016 to process labels: only nodules labeled by at least 3 radiologists are positive and the rest are excluded. We extract descriptions of nodules of our interest from xml files that LIDC-IDRI provides.

  The major difference between two versions is that CT slices are sorted by SliceLocation(ImagePosition) by v9 while by InstanceNumber by v15. Other minor differences include labels of some nodules with holes and with multiple connected components.

- For DSB stage1, apart from labels at scan-level, we manually and roughly label scans in training set, where cancers are diagnosed, and those in test set. We check the whole dataset and flip CT volumes if they are not scanned from head to foot.

  For SPIE, no additional processing is applied.

  DSB stage1 and SPIE are consistent in both v9 and v15.

- CT Volumes are converted into hdf5 files and nodule-level labels are stored in npy files. Each following step might process these data and generate their own training data accordingly. All the models are trained on two versions respectively.

flip regression

For convenience, all CT volumes should be scanned from head to feet. To achieve it automatically, we implement a simple CNN, which is trained on DSB stage1. The input is the slice chosen from 11th to 19th slices, and the output is whether the scan is from head to feet. For prediction, we average outputs for 5 slices.

Lung Segmentation

In Lung Segmentation step, we use a threshold to get low-HU areas. The largest area is regarded as the pulmonary parenchyma area after some morphological operations such as dilation and erosion.

4 Nodule Detection

In our algorithm, we first introduce a deconvolutional structure to Faster Region-based Convolutional Neural Network (Faster R-CNN) for candidate detection on axial slices. Then, a 3D DCNN is presented for false positive reduction. The framework of our nodule detection algorithm is illustrated in Fig. 1.

Owing to the much smaller size of pulmonary nodules compared with common objects in natural images, original Faster R-CNN, which utilizes five-group

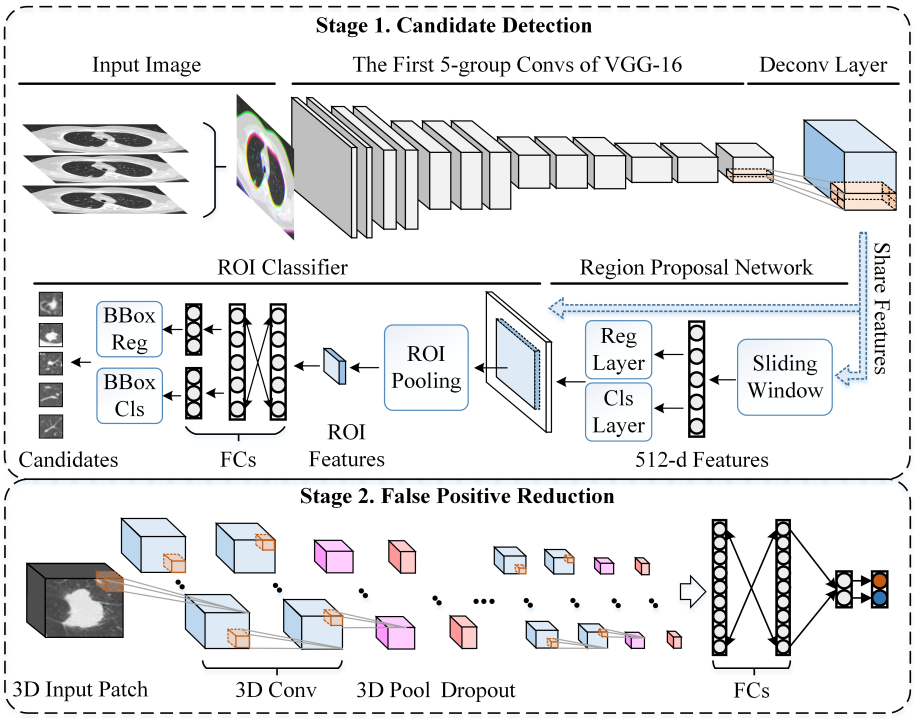

Figure 1: The framework of our nodule detection algorithm

convolutional layers of VGG-16Net [3] for feature extraction, cannot explicitly depict the features of nodules and results in a limited performance in detecting ROIs of nodules. To address this problem, we add a deconvolutional layer, whose kernel size, stride size and padding size are 4, 4, and 2 respectively, after the last layer of the original feature extractor. Note that, the added deconvolutional layer recovers more fine-grained features compared with original feature maps, the proposed model thus yields much better detection results than the original Faster R-CNN. To generate ROIs, we slide a small network over the feature map of the deconvolutional layer. This small network takes a 3×3 spatial window of deconvolutional feature map as input and map each sliding window to a 512-dimensional feature. The feature is finally fed into two sibling fully-connected layers for regressing the boundingbox of regions (i.e. Reg Layer in Fig. 1) and predicting objectness score (i.e. Cls Layer in Fig. 1), respectively.

At each sliding-window location, we simultaneously predict multiple ROIs. The multiple ROIs are parameterized relative to the corresponding reference boxes, which we call anchors. To fit the size of nodules, we design six anchors with different size for each sliding window: 4×4, 6×6, 10×10, 16×16, 22×22, and 32×32 (See Fig. 2). The detailed description of RPN is given in [2].

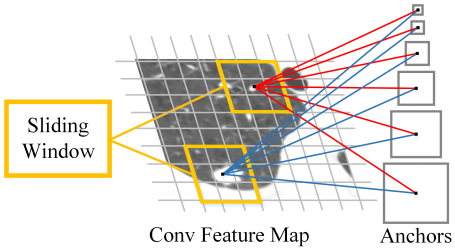

Figure 2: Illustration of anchors in the improved Faster R-CNN

4.1 False Positive Reduction Using 3D DCNN

In the consideration of time and space cost, we propose a two-dimensional (2D) DCNN (i.e. Improved Faster R-CNN) to detect nodule candidates. With the extracted nodule candidates, a 3D DCNN, which captures the full range of contexts of candidates and generates more discriminative features compared with 2D DCNNs, is utilized for false positive reduction. This network contains several 3D convolutional layers which are followed by Rectified Linear Unit (ReLU) activation layers, 3D max-pooling layers and fully connected layers are also used. A final 2-way softmax activation layer to classify the candidates from nodules to none-nodules. Moreover, dropout layers are added after max-pooling layers and fully-connected layers to avoid overfitting. We initialize the parameters of the proposed 3D DCNN by the same strategy using in the literature [1].

As for inputs of the proposed 3D DCNN, we firstly normalize each CT scan. After that, for each candidate, we use the center of candidate as the centroid and then crop a $40 \times 40 \times 24$ patch. The strategies including crop, flip and duplicate.

Note that, 3D context of candidates plays an important role in recognizing nodules due to the inherently 3D structure of CT images. Our 3D convolutional filters, which integrate 3D local units of previous feature maps, can 'see' 3D context of candidates, whereas traditional 2D convolutional filters only capture 2D local features of candidates. Hence, the proposed 3D DCNN outperforms traditional 2D DCNNs.

5 Nodule Malignancy Classification

For training Convolutional Neural Network to classify malignancy of nodules, we adopt the architecture of 3D VGG modified with shortcut. The input is same to false positive reduction, except candidates have been reduced greatly. We use the technique like data augmentation including rotation and flip to prevent overfitting, and exponentially decrease the learning rate to converge steadily. An optional center loss is applied in order to produce more distinguishing models. We train our model on two versions of candidates produced by our detection algorithms. We ensemble different epochs of models trained on two distinct versions of candidates with and without center loss. The number of models to ensemble is up to nearly 40 models. Cross validation results are generated for the next section to train logistic regression. Prediction results are generated

for further processing in the next section to ensemble models. The modified 3D CNN is trained by Adam with Keras backended on Tensorflow. 4-fold cross validation is applied.

6 Model Ensemble

Logistic regression is applied to get the final maligancy probability of a CT scan from malignancy of nodules. We add l2 regularization penalty scale to logistics regression. At the final stage, We constraint the output probability into to avoid severe mistakes. We ensemble the results generated by different epochs of models trained on two versions of data. For each model at the section malignancy classification, we train logistics regression on training folds and validate it on validation folds. And then, we only need to ensemble outputs of these logistics regressions. We exclude the minimum and maximum scores, then average the remaining scores from different models to get the final probability. To place more priority to well-performed models, we calculate the score of these models twice.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *IEEE international conference on computer vision*, pages 1026–1034, 2015.

[2] S. Ren, K. He, R.B. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[3] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, pages 1–9, 2015.