Machine Learning Engineer Nanodegree

Capstone Proposal

Ewa Nowacka

June 3rd, 2018

Proposal

Domain Background

According to the information found in [2], lung cancer is the second most common cancer and the leading cause of cancer death for men and women. Lung cancer makes up 14% of all new cancer diagnoses and accounts for 1 in 4 cancer deaths.

The American Cancer Society (see [1] for details) estimates that in 2018 there will be about 234,030 new cases of lung cancer (121,680 in men and 112,350 in women).The deaths from lung cancer in 2018 are expected to be about 154,050: 83,550 in men and 70,550 in women

Lung cancer detected early can be successfully treated. However, usually the patients do not experience symptoms of the cancer unless they are already at an advanced, non-curable stage. These symptoms (cough, chest pain, wheezing) are often mistaken for respiratory infections or effects of long term smoking,delaying the correct diagnosis.

Screening people with high risk of lung cancer, e.g. long-time smokers, people exposed to asbestos, mine, mill, textile.and plant workers, has been the most successful method in lung cancer detection. As per [3], chest x-rays aren't as effective as CT scans in detecting early signs of lung cancer. Using CT scans for lung cancer screening, has a much higher rate of successfully diagnosed lung cancer.

This project intends to design a model that will perform a classification of the potential cancer nodules found in the CT scans of two groups: negative and positive for cancer. Convolutional neural network framework has been chosen for this purpose. This selection is dictated by the research conducted in this

field: deep neural network for skin cancer detection ([4]), deep neural network for breast cancer detection ([5]) and lung cancer detection with CT scans ([6]).

Problem Statement

The project's main objective is to define a Convolutional Neural Network (CNN) that is capable of assigning the correct label (positive for cancer, negative for cancer) in lung CT scans. Therefore, it is a simple classification problem with binary response (only two possible labels) which uses lung CT scans as the input data.

Convolutional networks are widely used for image classification with many successes especially in the medical field (see [4], [5] and [6]). Outside of the medical applications, there are several CNNs that became industry standard such as InceptionV3 by Google ([7]). An alternative to CNN can be Supported Vector Machine (SVM) which also has been used to solve imagine classification problems ([8], [9]). However, this project will mostly focus on the CNN approach.

The classification problem is easily measurable; the accuracy of the model can be verified by simple count of the correctly classified images, which is known as *accuracy.* Additionally, for the medial applications, sensitivity and specificity measures are widely used.

The problem considered in this project is also quantifiable - it essentially consists of finding the CNN design and CNN weights that will give the best (or greater than established threshold) accuracy. Moreover, the problem is replicable - given the input data, CNN design and CNN weights, the results can be reproduced.

Datasets and Inputs

The dataset used for this project consists of 2948 nodule snippets extracted from *LUNA16* dataset found on GitHub repository ([9]). Original *LUNA16* can be found in [10] but it is available only upon participation in the challenge.

LUNA16 is a collection of lung CT scans of 888 patients. They are originally in MetaImage format (mhd/raw) and they have size of 512x512 pixels. The snippets contain only potential cancer nodules and have size of 40x40 pixels, which makes them much more convenient for the modelling purposes. Apart from the CT scans, *LUNA16* dataset contains also nodule coordinates used to extract these snippets, however this process is out of the scope of this project.

Each snippet file is a PNG file and has the following naming convention: *seriesid* (provided in the csv file with the nodules' coordinates), *nodule id* (*nod0, nod1, nod2* all the way up to XX), *slice id* (*slc1, slc2*, etc.) and l*abel* (*pos* for positive and *neg* for negative). The example of the file name is provided below: *1.3.6.1.4.1.14519.5.2.1.6279.6001.100621383016233746780170740405nod0slc1pos.png.*

Solution Statement

The lung CT scans classification problem will be solved in two main steps: data preparation and CNN design and training. Data preparation step will explore different techniques such as data augmentation. The heart of the project, CNN design and training, will focus on defining different CNN architectures and choosing the one with the best accuracy. The choice of the different CNN architectures will be guided by the research papers on CNNs (such as [14]) and blog posts published by Deep Learning professionals ([12], [13], [14]).

Benchmark Model

The model described in [10] will be used as a benchmark model since the same data and the same methodology was used. The obtained accuracy was 96.38%. The main objective of this project will be to beat this result.

Another benchmark model will be a binary classifier based on the Support Vector Machine approach. A simple SVM classifier will be created as a part of this project. To decide which model performs better, accuracy, sensitivity and specificity measures will be used.

Evaluation Metrics

Three evaluation metrics will be used to measure the model performance: accuracy, sensitivity and specificity.

Accuracy is defined as a percentage of the correctly classified images out of the total number of the images. Sensitivity and specificity are measures used mostly in medical applications. In this project, sensitivity will inform what percentage of the positive lung cancer was correctly classified as positive (true positive rate). On the other hand, specificity will provide a true negative rate - what percentage of the negative lung cancer was correctly classified as negative.

Project Design

The project will consists of four main steps:

1. Data Preparation
2. Exploration of the different CNN architectures.
    a. Define CNN architecture.
    b. Train the model.
    c. Validate the model.
    d. Compare the accuracy of the obtained model versus CNN benchmark model.
3. Build SVM benchmark model and compare the accuracy.
4. Calculate the sensitivity and specificity of the CNN and SVM models.

For step 1, the following data preparation techniques may be considered: data augmentation (vertical and horizontal flips, zooming, rescaling data), data normalization and histogram equalization. To build CNN model, different CNN architectures will be explored: e.g. different net depths, different numbers of fully connected layers, different overfitting techniques and batch normalization. Ultimately, a combination of the data preparation approach and the CNN architecture with the best accuracy score will be selected.

References

[1] *Cancer Facts and Figures 2018,* American Cancer Society.

[2] https://www.cancer.net/cancer-types/lung-cancer-non-small-cell/statistics

[3] *Can Lung Cancer Be Found Early?,* American Cancer Society.

[4] *Dermatologist-level classification of skin cancer with deep neural networks*, Andre Este, Brett Kuprel, Roberto A. Novoa, Justin Ko, Susan M. Swetter, Helen M. Blau and Sebastian Thrun, 2017.

[5] *Convolutional Neural Networks for Breast Cancer Screening:Transfer Learning with Exponential Decay,* Hiba Chougrad, Hamid Zouaki and Omar Alheyane, 2017.

[6] *Deep Convolutional Neural Networks for Lung Cancer Diagnostics,* Mehdi Fatan Serj, Bahram Lavi, Gabriela Hoff and Domenec Puig Valls, 2018.

[7] *Going Deeper with Convolutions*, Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke and Andrew Rabinovich, 2014.

[8] *Toward intelligent training of supervised image classifications: directing training data acquisition for SVM classification,* Giles M. Foody, Ajay Mathur, 2004.

[9] *Support vector machine for breast MR image classification,* Chien-Shun Loa, Chuin-Mu Wang, 2012.

[10] https://apollack11.github.io/machine-learning.html

[11] https://luna16.grand-challenge.org/data/
[12]https://alexisbcook.github.io/2017/global-average-pooling-layers-for-object-localization/
[13]https://blog.keras.io/building-powerful-image-classification-models-using-very-little-data.html
[14]https://blog.insightdatascience.com/automating-breast-cancer-detection-with-deep-learning-d8b49da17950
[15] *Dropout: A Simple Way to Prevent Neural Networks from Overfitting,* Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, Ruslan Salakhutdinon, 2014.