Mammogram classification using convolutional neural network

M.C. Mayra Cristina Berrones Reyes, PhD. María Angélica Salazar Aguilar

mayra.berronesrys@uanl.edu.mx, maria.salazaragl@uanl.edu.mx

1. Introduction

Breast cancer is one of the major health problems that lead to early mortality in women.

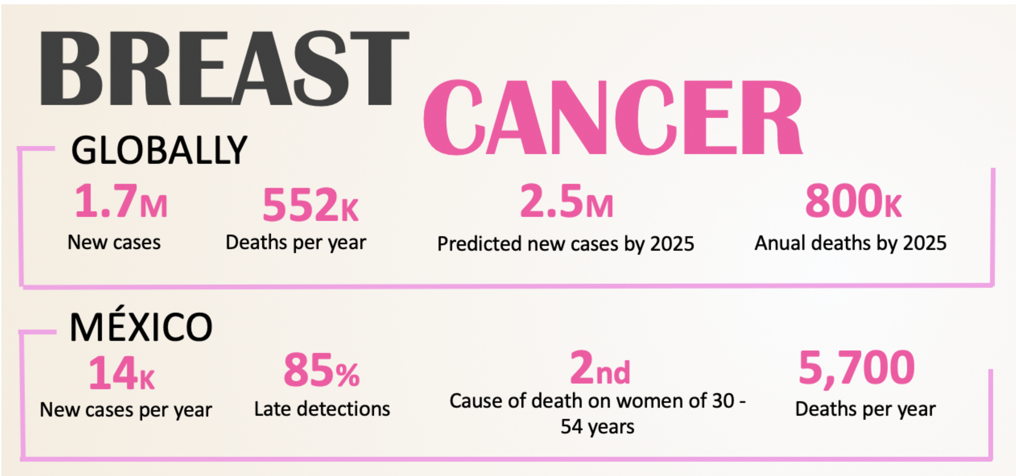

Figure 1: Signs for anomalies of breast cancer [1].

Medical image analysis requires a lot of measurements made by radiology experts to be able to identify an anomaly, which is time-consuming, and impractical when we compare the small number of specialists with the large amount of information they receive on daily basis. This causes a big bottleneck on the analysis diagnosis of all of the images.

To solve this problem, radiologists use tools known as CAD (Computer-Aided Detection) or CADx (Computer-Aided Diagnostic).

2. Objective

Implement different features on CNN architectures, and analyze the impact of these features and how they affect the performance of the models.

3. Resources

The datasets used for this experiment are:

Table 1: Image distribution of datasets.

Name of the database	Total number of images	Images with anomaly	Images without anomaly
Mini-Mias	322	121	201
DDSM	2778	0	2778
YERAL	200	70	130

python + K + TensorFlow

Figure 2: Tools used to build the CNN.

The implementation of all the CNN models was carried out on a Workstation with Intel Xéon (R) CPUE5-1620 v2 3.70 GHz x8 and a Quadro K600 PCIe SSE2 GPU.

4. Metodology

The performance of the CNNs models was evaluated by taking into account the classical metrics used to evaluate the performance of any machine learning tool. The first metric we have is the accuracy, which computes the percentage of correct predictions over all kinds of predictions made.

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN} \tag{1}$$

Then we have the F1 score which can be seen as a weighted average of the precision and recall, other two famously used metrics in machine learning.

$$\text{F1 Score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{(\text{Precision} + \text{Recall})} \tag{2}$$

For detailed information about the metrics and when is recommended to use them, we recommend to read [2].

Using the datasets mentioned in Table 1 we proceed with image processing. We applied the filters and translations to enlarge the total amount of images available. After the image transformations we obtained a total of 12,300 images with an anomaly, and 11,500 without.

5. Architecture

In summary, we set a priori the following features in the CNN models:

- 2 convolutional layers.
- 32 filters on first convolutional layer.
- Max pooling layer after each convolutional layer.
- 1 filter on the second fully connected layer.
- 10 epochs of 1,000 iterations on training set.
- 2,000 iterations on developing set.
- ReLu function for convolutional and first fully connected layer.
- Sigmoid function for second fully connected layer.
- Binary cross-entropy as a loss function.

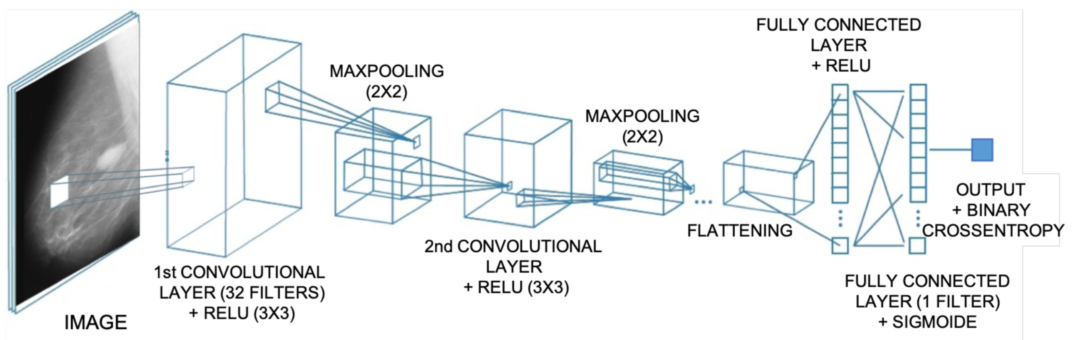

Figure 3: Structure of our CNN model.

Once we set the features described in the previous subsection, we designed and analyzed an experiment by using different levels of image size (1), number of filters in the second convolutional layer (2), number of neurons in the first fully connected layer (3), and different optimizers for training (4) (see Table 2).

Table 2: Changing parameters on models.

1	2	3	4
1 (80 ×200)	1 (32)	1 (128)	1 (Adam)
2 (160×400)	2 (64)	2 (240)	2 (Adamax)
	3 (128)	3 (380)	3 (Nadam)
			4 (RMSProp)
			5 (SGD)
			6 (Adadelta)

As the combination of variant features yields to more than 100 models, we labeled each model as a combination of the feature levels, for example, the name of a model with an image size of 160×400, a second convolutional layer with 128 filters, a first fully connected layer with 380 neurons, and the Adam optimizer, will be **2 3 3 1**.

6. Results and discussion

To analyze the optimizer's performance, we consider the accuracy and loss obtained with all CNN models when using the developing set with both image sizes. The result of this comparison can be seen in Figure 4

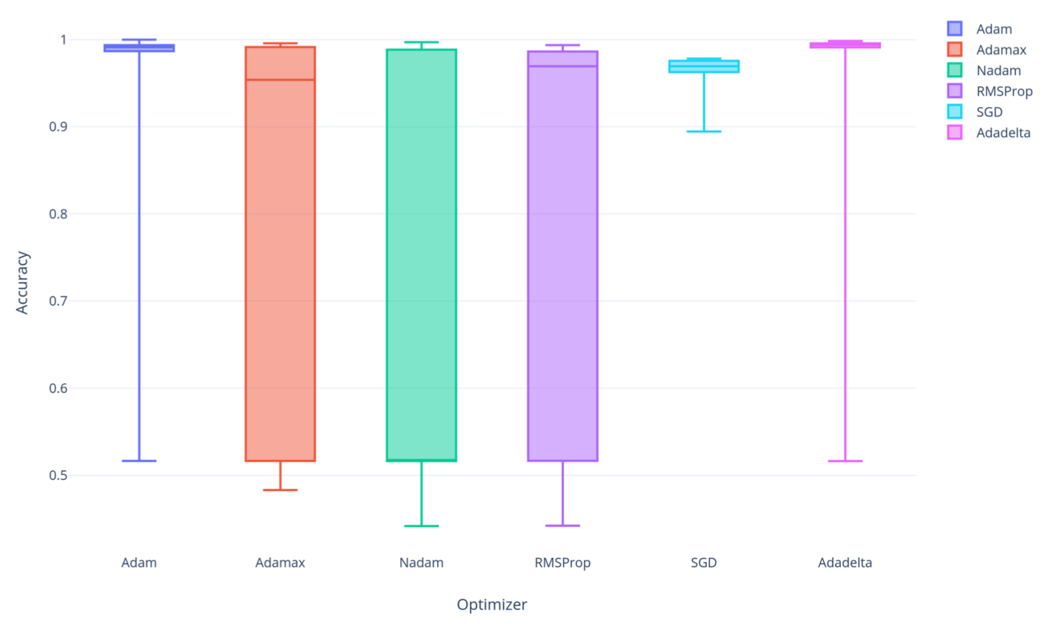

Figure 4: Accuracy comparison in optimizers.

From Figure 4, we can see that the accuracy, are more compact looking in the Adam, Adadelta, and SGD optimizers. With these optimizers the results were more consistent for all models, showing the SGD the most consistent results.

After observing the optimizers reported the best behavior, we decided to keep the best two CNN models for each of those optimizers and each image size. Therefore, we obtained 12 different models divided into two classes, A and B, one per image size.

Table 3: Best performing models for A and B classes.

Class	Model Id	Accuracy: Training set	Accuracy: Developing set	Accuracy: Test set	F1
A	1 2 2 6	99.65%	99.57%	70.63%	82.53%
B	2 1 3 1	99.41%	99.79%	72.50%	83.70%

Model class A contains the best models reaching 70.63% of accuracy and 82.53% of F1 value. In model class B we reach 72.50% in accuracy and 83.70% of F1.

Intending to improve this performance, we carried out another experiment in which we added another convolutional layer with 32 filters to each model in classes A and B. The resulting classes are called class C and D, respectively, as shown in Table 4. In these classes, we also used the same datasets than in class A and B.

Table 4: Best performing models for C and D classes.

Class	Model Id	Accuracy: Training set	Accuracy: Developing set	Accuracy: Test set	F1
C	1 2 3 1	99.26%	99.97%	76.25%	86.23%
D	2 2 2 1	99.04%	99.89%	78.75%	88.11%

In comparison with the performance of the best models in classes A and B, the increment of accuracy reaches around 6% and in the F1 value 5%.

7. Conclusions and work in progress

As we can appreciate in Figure 6 as well as Table 3 and 4 there is a significant improvement when we add another convolutional layer.

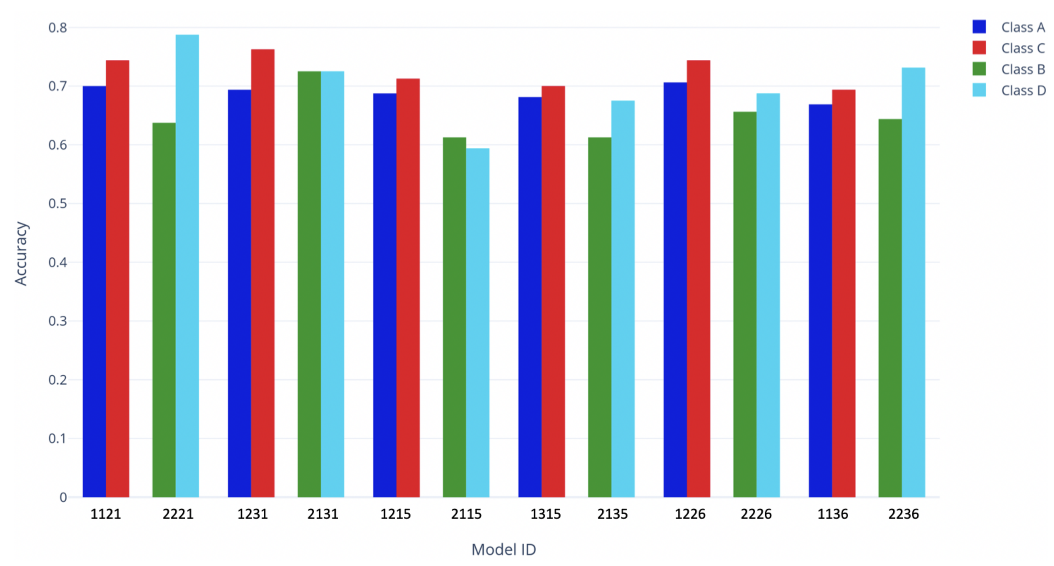

Figure 5: Accuracy comparison in class models.

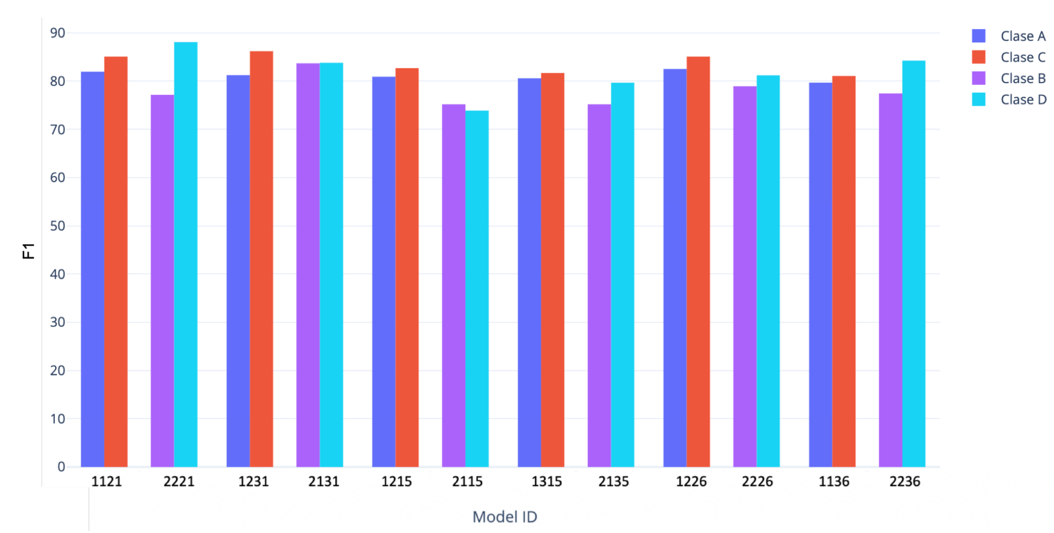

Figure 6: F1 comparison in class models.

For this we want to continue our investigation with the best resulting models, continuing this experiment by using a different set of parameters on the distribution of the data, playing some more with the number of dropouts, or even adding more convolutional layers.

References

[1] Cancer.org. (2018). ¿Qué es el cáncer de seno?. *[Online] Available in: https://www.cancer.org/es/cancer/cancer-de-seno/acerca/que-es-el-cancer-de-seno.html American Cancer Society.*

[2] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM, 60(6), 84–90. doi:10.1145/3065386*

The authors would especially like to express their gratitude to the National Institute of Cancerology.

PISIS Posgrado en Ingeniería de Sistemas