

École polytechnique de Louvain

Deep Learning in Mammography

Active Learning approach for smart annotation queries

Author: **Maxime ZANELLA**

Supervisor: **Benoît MACQ**

Readers: **Benoît MACQ, Elliott BRION, John LEE, Mariana ANDRADE VIEIRA, Vincent NICOLAS**

Academic year 2020–2021

Master [120] in Mathematical Engineering

Deep Learning in Mammography

Active Learning approach for smart annotation queries

Maxime ZANELLA

Abstract

Breast cancer is the most common cancer among women. Despite great progress in patient care and early screening, it remains a major health issue. Mammography screening is now widely used, but it faces significant problems such as a high rate of false positives and overdiagnosis. To alleviate the work load induced by this widespread strategy, computer-aided diagnosis has been introduced in the 1970s, but it still encounters reluctance from senologists. Deep Learning solutions for mammography assessment are reaching maturity and are now being used in healthcare facilities. However, these approaches are very data-intensive, making their deployment complex for medical tasks. We suggest using Active Learning to address these concerns, which allows the model to request specific annotations from an expert and achieve better performance.

First of all, this master thesis aims to assess three Deep Learning approaches to diagnose benign and malignant masses in mammograms. We use the publicly available dataset CBIS-DDSM to train our models and evaluate our work. We show that a whole-image classifier based on ResNet has difficulties identifying the right area. Then, a patch approach is investigated, proving that Deep Learning models can discriminate masses with a specificity of 0.9 at 0.1 sensitivity. Afterward, we implement a Faster R-CNN network with a sensitivity of 0.463 at 0.5 false positives average per image.

Furthermore, we provide a comprehensive overview of Active Learning techniques for Deep Learning. We evaluate several strategies from the literature with the same experimental parameters to make the results comparable. The entropy measure, core-set approach, ensemble method, Monte Carlo dropout, learning loss, and discriminative Active Learning are studied on the MNIST dataset. Finally, we apply some of these methods to mammography and investigate the relevance of using uncertainty, diversity, and a combination of both measures to annotate regions of interest intelligently.

Contents

1	Introduction	1
1.1	Breast anatomy	4
1.2	Abnormalities	5
1.3	Breast cancer	6
1.4	Mammography	6
1.5	BI-RADS terminology for structured report	7
1.6	AI for image interpretation	8
1.7	Contribution of this master thesis	11
1.8	Structure of this master thesis	11
1.9	Summary of Chapter 1	11
2	Materials and Deep Learning	13
2.1	Convolutional Neural Networks architectures: state of the art . . .	13
2.2	Deep Learning in mammography: state of the art	15
2.3	Experiments framework	16
2.3.1	Model architecture: Classification	16
2.3.2	Model architecture: Detection	18
2.3.3	Dataset presentation	20
2.3.4	Dataset preparation	20
2.3.5	Lack of data and overfitting	23
2.3.6	Performance measures	23
2.4	Classification networks	26
2.4.1	Whole-image classifier	26
2.4.2	Patch approach	28
2.4.3	Calibration	31
2.5	Detection networks	32
2.6	Summary of Chapter 2	35
3	Active Learning	39
3.1	Theoretical settings	40
3.1.1	Scenarios	40

3.1.2	Workflow	41
3.2	Acquisition Functions and Querying Techniques	41
3.2.1	Acquisition functions	42
3.2.2	Uncertainty Sampling	42
3.2.3	Diversity sampling	44
3.2.4	Query-By-Committee	44
3.3	Active learning in Convolutional Neural Networks: state of the art .	45
3.3.1	Uncertainty sampling	45
3.3.2	Diversity sampling	50
3.4	Summary of Chapter 3	54
4	Results	55
4.1	Experiment 1: Active Learning strategies comparison	55
4.1.1	Entropy uncertainty sampling and Monte Carlo dropout . .	56
4.1.2	Ensemble technique	59
4.1.3	Core-set and discriminative Active Learning	62
4.1.4	Learning Loss Active Learning	65
4.1.5	Summary of experiment 1	67
4.2	Experiment 2: Active Learning for Mammography	68
4.2.1	Uncertainty and diversity	69
4.2.2	Combining uncertainty and diversity	69
4.2.3	Summary of experiment 2	71
5	Future work	73
5.1	Deep Reinforcement Active Learning	74
5.2	Coalition theory	75
5.3	Summary of Chapter 5	76
6	Discussion and conclusion	77
6.1	Discussion on breast cancer screening	77
6.2	Limitations, improvements of the results and future work	78
	Appendices	95
A	Active Learning algorithms	95
B	Details on the experiments	99
C	Active Learning for a detection model	103

Chapter 1

Introduction

Breast cancer is the most common cancer among women, accounting for one in four cases. According to the World Health Organization, in 2020, there were 2.3 million women diagnosed with breast cancer and 685 000 deaths globally [103]. Despite considerable progress in patient care and early breast screening, it is still one of the leading causes of death worldwide: one woman in 39 will die from it, reports the American Cancer Society [1]. In Belgium, 11734 new cases and 2362 deaths were recorded in 2020, showing that it is still a significant health issue. Figure 1.1 shows that breast cancer represents 24.5% of all cancer incidence for female and 15.5% of all cancer deaths for female in 2020. Moreover, the estimated number of cases and the mortality rate will increase worldwide by an estimated 33.8% until the year 2040 [6] [7].

In this respect, early detection through large-scale breast cancer screening (primarily through mammography) is of major importance. In many countries, women between the ages of 50 and 70 are regularly (usually every two years) invited to participate in breast cancer screening. During the screening, the **senologist**, i.e., a doctor who studies diseases that affect the breast, evaluates a risk factor according to specific rules (BI-RADS). Each mammography is independently double-read in a blind manner or double-checked to avoid miss findings and to ensure better diagnosis with higher sensitivity and specificity.

Mammography has been proven to detect breast cancer at an early stage [80], allowing better patient management and treatment. Even if other screening technologies such as digital tomosynthesis (DTS) can be used, mammography stays the current standard method for screening and longitudinal patient monitoring with various strategies ranging from every year to every three years (European Union). Belgium had the highest detected breast cancer rate in the world in 2018, according to the World Cancer Research Fund [102]. This high incidence rate is partly due to our lifestyle, our rich diet, and our screening strategy. This situation significantly

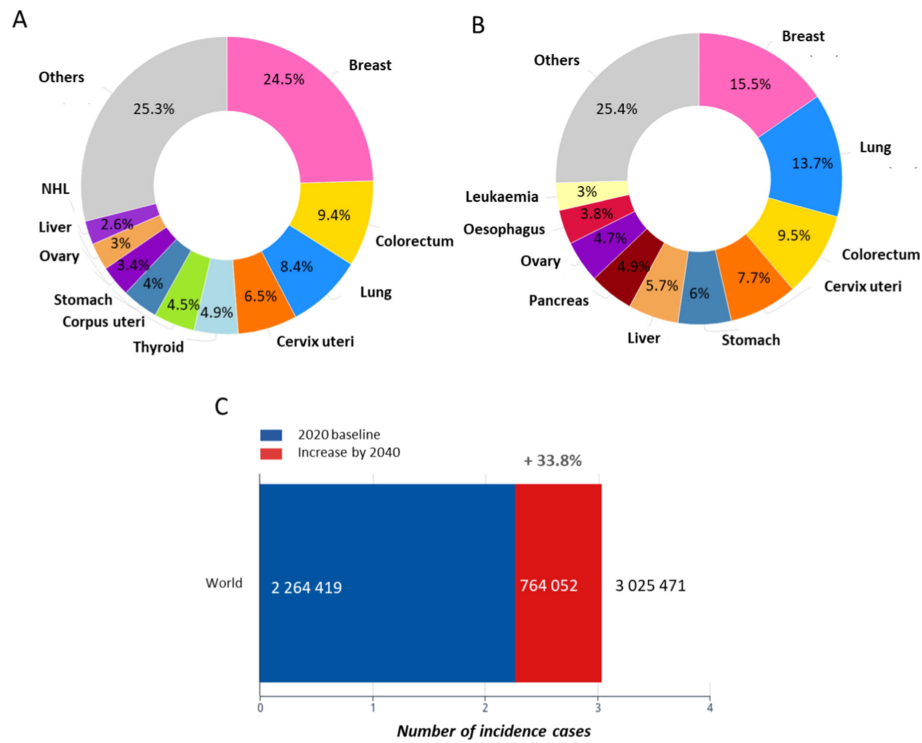


Figure 1.1: Breast cancer incidence (A) and mortality (B) for female in 2020. The estimated increase in breast cancer cases from 2020 to 2040 (C). Source: Paz Boix-Montesinos et al. [6], Freddie Bray et al. [7].

increases the radiologist’s workload and, therefore, the risk of missing questionable cases [38]. A second major problem is a possible overdiagnosis (diagnosed cancer that would not have come to attention in a woman’s lifetime, i.e., a false-positive case that is not dismissed) and the unnecessary medical examinations and treatments that may follow. Even if it is difficult to estimate precisely the extent of this phenomenon, overdiagnosis is recognized as a severe issue [64]. A final concern is the false-positive rate. Mammography has a statistically significant false-positive rate: one-third of healthy women receive an abnormal result that requires additional evaluation over ten years [34]. This issue generates a substantial extra cost for society (around 2.8 billion dollars each year for false positives and 1.2 billion dollars per year for overdiagnosis for the USA only [66] due to additional screenings, consultations, biopsies, etc.) but also avoidable stress and discomfort for the patients. Other concerns about traditional mammography and early screening are discussed in Chapter 6. Considering all the benefits and problems, mammography screening still confers a significant benefit but requires improvements in interpretation, reading time, and reader performance.

Since the 1970s, computer-aided diagnosis (CAD) has been introduced in the radiologist workflow. In 2010, three-quarters of the mammograms were read by a CAD (in the USA) [71]. However, the benefits were relatively poor, inducing on many occasions a reduced accuracy, higher recall and biopsy rates, unnecessary additional medical examinations, increased stress, and discomfort for the patient [65]. Moreover, interpretation time increased due to unnecessary verifications, which is the opposite effect of the original purpose. Finally, these systems have been shown to introduce automation bias due to overreliance on the proposed diagnosis [38].

These CADs mainly relied on traditional machine learning algorithms (computer systems that can learn and adapt themselves from patterns in data) and image processing methods. Detecting a tumor in a breast is a complex task. A tumor may take various forms: masses, microcalcifications, asymmetries, architectural distortions. A breast may also take diversified shapes and be more or less dense. Senologists use precise terminology during the screening that a CAD's prediction must match. The BI-RADS (Breast Imaging Reporting and Data System) classification [5] enables radiologists to communicate results in a standardized and consistent way with a fixed lexicon. Each mammogram is given a risk assessment from zero to six, and afterward, specific recommendations are made. Therefore, a good CAD must be based on this terminology, carefully designed, highly flexible, and ensure similar performances for all potential cases.

Recently, Deep Learning (a subset of machine learning) and especially Convolutional Neural Networks (CNN) (Deep Learning applied to images) has experienced a tremendous expansion, thanks to its state-of-the-art performance in a lot of tasks (including classification, detection, and segmentation). The main principle of these algorithms is that they do not base their prediction on carefully hand-engineered features like traditional machine learning algorithms. Instead, they learn them directly from data. On the one hand, this is their greatest strength, allowing Deep Learning models not to be constrained and freely learn new discriminative features that optimize the prediction power of the algorithm. On the other hand, this is one of the biggest obstacles to their acceptance in the work of the radiologists: their prediction is difficult to analyze and understand; they constitute a so-called « black box ». This lack of interpretability is one of the primary sources of the radiologists' reluctance and one of the biggest challenges for advanced AI assistants [76]. Deep Learning (DL) algorithms often need many data with annotations, also called labels, to learn their features (e.g., if there is a tumor or not, a bounding box surrounding the problematic area that it has to detect, etc.). The scarcity of annotated data and the lack of interpretability make their deployment in the

medical sector a complex task.

Deep Learning for mammography breast cancer screening is reaching maturity and starts to be used in screening centers. However, one of the central challenges in Deep Learning applications is still obtaining labeled data. Active Learning, a paradigm studied in this thesis, aims to address part of this problem: its goal is to determine the most informative images for the machine learning process. Facing vast information spaces like mammography, Active Learning significantly reduces the number of annotations required, and therefore the time the experts devote to training the AI. Active Learning can be implemented through several approaches and heuristics. Although this paradigm has been well studied for machine learning [85], the literature about Active Learning for Deep Learning remains relatively sparse, and its application is still challenging due to the complexity of the models. Therefore, with this paradigm, better evolutionary interaction between the radiologist and the AI is possible, making better use of the available annotation time of the experts.

This chapter aims to present firstly the breast anatomy and what can be observed on a mammography exam. Secondly, it proposes an overview of how a senologist works in his daily work and how artificial intelligence, as presented in the following chapters, can be integrated into his decision process.

1.1 Breast anatomy

The breast is composed of fatty, fibrous, and glandular tissue as shown in Figure 1.2. These tissues have different properties and can be distinguished during a mammography examination, which is presented in the following section.

- **glandular tissue** includes the lobes, organized in small lobules that are the glands producing milk. The milk is carried from the lobules to the nipple by tubes called breast ducts.
- **fibrous tissue** or **connective tissue** is the same tissue that ligaments are made of.
- **fatty tissue** fills in the space between the fibroglandular tissue.

Most breast cancers appear in the ducts and the lobes: a majority of them are located in the upper outer quadrant (near the armpit) because of the high density of glandular tissue. Breasts can have various densities (from A to D, see Section 1.5) based on the amount of fibrous and glandular tissue compared to the amount of fatty tissue. Moreover, dense breasts are more subject to cancer and make abnormalities harder to find on a mammogram as shown in Figure 1.3 with a fatty

breast (A) and a dense one (D). These considerations must be taken into account to design an intelligent assistant that evaluates mammograms and is fair to all women.

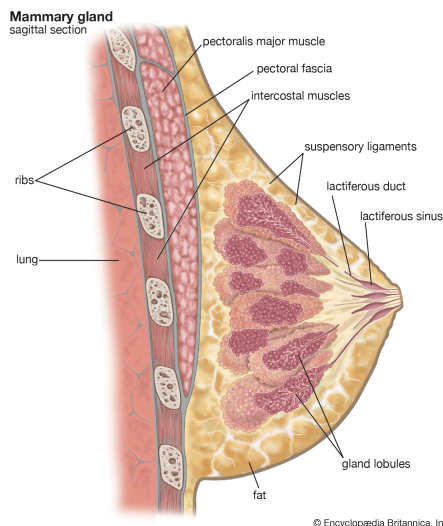


Figure 1.2: The breast morphology.
Source: Britannica [10].

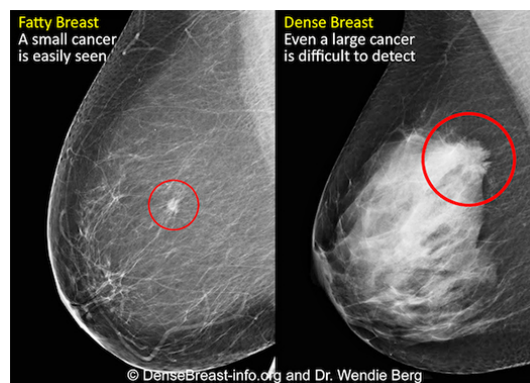


Figure 1.3: Variable difficulty in detecting cancer on a mammography exam. Source: Breastcancer.org [8].

1.2 Abnormalities

Sickles et al. [89] propose a complete guide for a standardized interpretation of breast images, including mammography. Radiologists are looking for abnormalities, any area that does not look like normal tissue is a cause of concern: white, high-density tissue, small white specks, etc. These findings are often classified into five categories: masses, asymmetries, calcifications, architectural distortions, associated features. Due to the lack of complete publicly available data, this thesis focuses on the assessment of masses (benign or malignant).

A mass is a space-occupying lesion characterized by its shape, contour and density [69], see Figure 1.4.

- **Shape:** round, oval, lobulated, or irregular.
- **Contour:** circumscribed, microlobulated, obscured, indistinct or spiculated.
- **Density:** high, medium, or low fat.

The more the contour of a mass is irregular, the more there is a risk of malignancy.

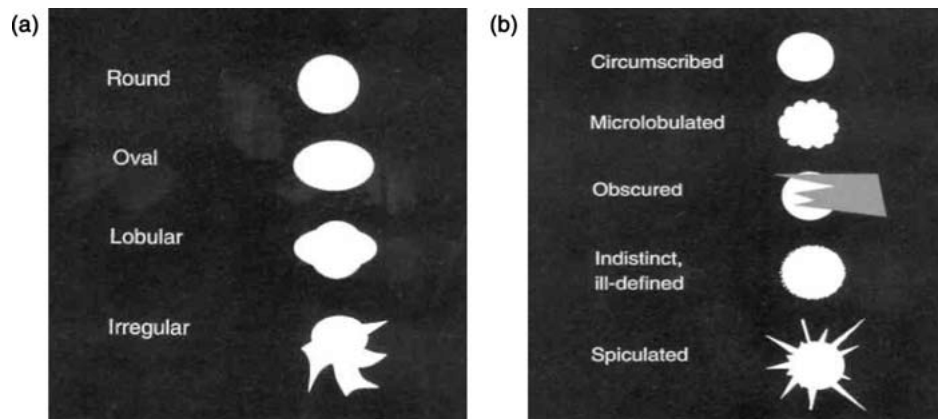


Figure 1.4: Masses can take various shapes (a) and contours (b). Source: Singh & Nagarajan [92].

1.3 Breast cancer

It is important to note that many tumors are benign and are not likely to evolve. Overdiagnosis is therefore a major issue. The senologist must classify these tumors as invasive or non-invasive (cancer does not spread into the surrounding tissue). A majority of cancers are adenocarcinomas: they start in gland cells.

- **ductal carcinoma** starts in the milk ducts and is either non-invasive (called ductal carcinoma in situ (DCIS)) or invasive (called invasive ductal carcinoma). In the case of invasive ductal carcinoma, the tumor grows into nearby breast tissues causing a lump or thickening of the breast. It can also spread to the lymph nodes in case of invasive tumors and other parts of the body: this phenomenon is called metastasis. Invasive ductal carcinoma accounts for roughly 70 to 80 percent of all cases [12].
- **lobular carcinoma** starts in the lobules and is also either non-invasive (called lobular carcinoma in situ (LCIS)) or invasive (invasive lobular carcinoma). In the case of invasive lobular carcinoma, the tumor grows into the nearby breast tissues and can also spread (metastasis) to lymph nodes and other parts of the body. The cancerous cells do not make a lump; instead, they make a thickened area of breast tissues making it harder to diagnose with mammography.

1.4 Mammography

Mammography aims to see inside the breast with a low-dose x-ray system (a small dose of ionizing radiation); a mammography exam is called a mammogram. The

common use of mammography is for screening (early detection) and diagnostic (evaluate a patient with abnormal findings). During the screening, both head-to-foot (bilateral craniocaudal or CC) and angle side-view (mediolateral oblique or MLO) images of the breast are taken [107]. Figure 1.5 illustrates a mammography exam. The breast is compressed between two parallel plates to equalize the thickness of the breast tissue and reduce the radiation dose needed to obtain a quality image.

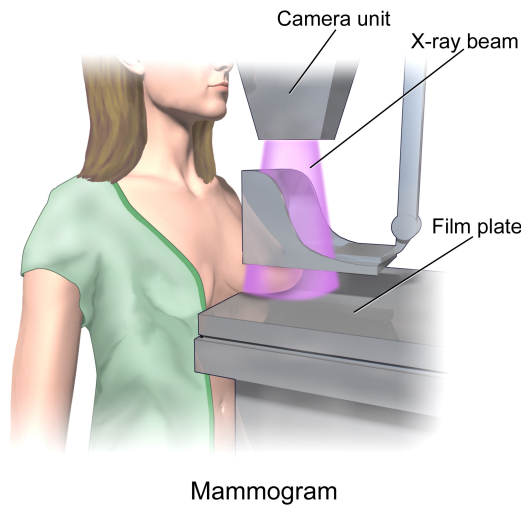


Figure 1.5: Mammography illustration, bilateral craniocaudal (CC) view. Source: Brent Burbridge [11].

Another screening technology called digital tomosynthesis (DTS), a three-dimensional breast scan, was approved for breast cancer screening in 2016, but its benefit is debated [82]. The reading time needed is also increased due to the number of slices. However, this type of exam could contain much more information than classical mammography, which could be interesting for an AI.

1.5 BI-RADS terminology for structured report

The American College of Radiology introduced a standardized terminology to assess a mammogram called BI-RADS (Breast Imaging Reporting and Data System). It enables radiologists to organize their reports in a standardized way and with a fixed lexicon, with final assessment and specific management recommendations (nothing to do, further screenings, biopsy, more specific recommendations) [5]. It classifies abnormalities from 0 to 6 as shown in Table 1.1, but also breast densities from A to D (called ACR categories) as shown in Table 1.2.

BI-RADS classification	Signification	Risk of cancer	Recommendations
0	Not conclusive	n/a	Recall (e.g., additional screening)
1	Normal	Essentially 0 %	Routine screening
2	Benign abnormality	Essentially 0%	Routine screening
3	Probably benign abnormality	Between 0% and 2 %	Follow-up (e.g., in 6 months)
4	Abnormality needing a biopsy	Between 2% and 95 %	Biopsy (tissue diagnosis)
4A	Low suspicion	Between 2% and 10 %	
4B	Moderate suspicion	Between 10% and 50 %	
4C	High suspicion	Between 50% and 95 %	
5	Very high suspicion of cancer	More than 95%	Biopsy (tissue diagnosis)
6	Biopsy-proven cancer	100%	Treatment (e.g., surgical excision)

Table 1.1: BI-RADS classification for abnormalities according to the ACR BI-RADS® Atlas [89].

ACR category	Description	Percentage of fibrous and glandular tissues
A	Fatty breasts	Less than 25%
B	Scattered fibroglandular densities	Between 25% and 50%
C	Heterogeneously dense breasts	Between 51% and 75%
D	Extremely dense breasts	More than 75%

Table 1.2: Levels of breast density (ACR categories) according to the ACR BI-RADS® Atlas [89].

All of these considerations, i.e, the technical properties of a mammogram, what can be seen on it, how to diagnose an exam (BI-RADS), must be taken into account when designing an intelligent assistant. The following section presents some ways to integrate an artificial intelligence into the workflow of the senologist.

1.6 AI for image interpretation

As mentioned above, a practical application of AI could reduce the time required for screening and reduce false positives. Mammograms are most of the time read twice: one role of the AI could be a replacement for this second reading.

Mayer McKinney et al. [60] study the impact of the use of an AI regarding time-saving and performance. They observe that they get similar performance compared to the traditional double-reading process, saving 88% of the second reader time. They also note that for some cases, the AI detects cancers that are not detected by the reader (and inversely), suggesting potentially complementary roles in reaching accurate conclusions.

Here is a list of positive and negative aspects of several ways to integrate an intelligent assistant into the radiologist's workflow. The integration of such systems is complex and some practitioners will prefer one solution over another. The last approach **Active preliminary results** introduces a machine learning paradigm: Active Learning, which is presented in Chapter 3, and which will be the core of this thesis.

- **On-demand:** the radiologist queries an analysis from the AI when he wishes.

Pros	Cons
– the clinician keeps the control (asking AI interpretation only when relevant). – reduces false positives by only asking selected cases.	– requires manual steps. – potential miss findings (not all cases are interpreted by the AI).

- **Dog barker:** the AI sends an alert in case of disagreement.

Pros	Cons
– automatic notifications (no manual steps). – potential to capture more findings (AI continuously in the loop).	– can be disruptive for the radiologist. – needs a really low false-positive rate to be accepted (could increase examination time, overdiagnosis, unnecessary complementary exams, etc.).

- **Preliminary AI results:** the AI proposes automatically a diagnosis/report that the radiologist can modify.

Pros	Cons
– potential to capture more findings (AI continuously in the loop). – potential to capture critical findings faster. – potential to reduce significantly the examination time. – radiologist ensures accuracy of final diagnosis and/or report.	– needs a low false-positive rate to be accepted (could increase examination time, overdiagnosis, complementary exams, etc.). – possible bias and passive reading of the results. – needs fast turnaround time.

- **Active preliminary results :** the AI proposes automatically a diagnosis/report that the radiologist can modify. Additionally, the AI can ask questions from the radiologist.

Pros	Cons
– potential to capture more findings (AI continuously in the loop). – potential to capture critical findings faster. – radiologist ensures accuracy of final diagnosis and/or report. – Active participation of the radiologist (better acceptance of the system, continuous stimulation, and ultimately, better performance of the assistant).	– needs a low false-positive rate to be accepted (could increase examination time, overdiagnosis, complementary exams, etc.). – requires a careful balance between time savings and additional time required of the radiologist. – needs fast turnaround time.

1.7 Contribution of this master thesis

The purpose of this master thesis is twofold: (i) study Deep learning approaches for mammography assessment and (ii) implement and benchmark specific Active Learning strategies for Deep Learning models applied to classic images and mammograms.

(i) Classification and detection networks are designed for mammography assessment: Three different ways of tackling the problem of mammography assessment are studied. We show that the classical whole-image approach makes it harder for the model to look at interesting areas and that a more refined approach combining spatial information with more precise areas performs better on the CBIS-DDSM dataset.

(ii) Active Learning strategies designed for Deep Learning models are explored: we provide a comprehensive overview of Active Learning strategies and benchmark several of them on the well-known dataset MNIST with the same experiment setting, making results comparable. In a second time, some strategies are applied to mammograms assessment models, showing the interest of future work in this area.

1.8 Structure of this master thesis

The master thesis is organized as follows: in Chapter 2, a classification Deep Learning model is applied to discriminate benign and malignant masses in mammograms. At the end of this chapter, a complete model allowing abnormalities detection is introduced. In Chapter 3, the machine learning paradigm Active Learning is presented, and some of the most promising methods are described. In Chapter 4, these methods are assessed on a well-known dataset and compared on the same basis before being tested on mammograms with uncertainty sampling, diversity sampling and a combination of both measures. Chapter 5 presents possible future work, and Chapter 6 concludes this thesis by discussing the limitations of the results obtained and the improvements to be investigated.

1.9 Summary of Chapter 1

Breast cancer, despite considerable progress in screening and treatment, remains one of the most common cancers causing death. Integration of AI in the senologist's workflow has already led to significant advances in patient monitoring. However, there is still reluctance in trusting an AI. Substantial improvements are still possible and research to help diagnose breast cancer has much to offer in the years to come.

Chapter 2

Materials and Deep Learning

Even if Deep Learning’s history can be traced back further decades ago, its evolution has taken on a great deal of importance only very recently. J. Schmidhuber [81] gives an overview of Deep Learning until 2014. Since the unveiling of AlexNet [49] and its state-of-the-art performance in 2012, the research and hype around Deep Learning and Convolutional Neural Networks have only increased. Each year, new models that are even more efficient, more complex, or with more original approaches, appear. Deep Learning models are good in several tasks interesting for image-based medical diagnostics (e.g., classification, detection, and segmentation). This thesis focuses on the classification and detection task.

In this chapter, the main advances in Deep Learning models for classification and detection in recent years are exhibited before introducing some state-of-the-art models applied to mammography. Finally, the first experiments based on the ResNet classification network are presented, and a state-of-the-art detection network called Faster R-CNN is implemented.

2.1 Convolutional Neural Networks architectures: state of the art

In classification, the task is to give a label to an image. The known dataset Imagenet¹ contains more than 1 million images divided into 1000 classes or labels (for comparison, one of the best-known largest publicly available mammography datasets contains 2620 scanned films). An important breakthrough after Alexnet is the presentation of VGGNet [91] in 2014, a deeper model (up to 19 weight layers). The same year GoogLeNet [98] was presented with different paths, enabling the network to choose the most appropriate architecture during training. The next

¹<https://image-net.org/>, Accessed 2020-23-11

year, ResNet [36] was introduced with the novel idea of shortcuts that allows modern networks to beat partially the gradient vanishing problem (the fact that the update signal during training can become very small for the first layers, making deep networks impossible to train). Networks can therefore become way deeper (up to more than 200 layers for ResNet). DenseNet [39] in 2016, with a similar idea, added connections not only to the next layer but to each layer until the end of the network (per block to keep the number of connections reasonable), enabling the reuse of feature maps from different levels. Numerous models seeking to improve performances have risen like EfficientNet [100] in 2019. It strategically scales deep neural networks in terms of depth, width, and resolution of the input image, resulting in a more accurate network with often much fewer parameters and a much faster training procedure. Speaking of architectures, it is worth mentioning transformers, initially introduced for natural language processing [101]. They are nowadays applied to image processing like [14]. They introduced Sequence-to-Sequence architectures, transforming a sequence (like words) in another sequence (like words in a different language) with the help of Encoder, Decoder, and more especially attention-mechanism.

Sometimes classification is not explicit enough, as a radiologist needs to know the problem area that conducted the system to its diagnosis. In detection, the task is to detect objects in an image, often by drawing a rectangle that embraces each of them. Several state-of-art methods have been proposed recently [118] with different ways to tackle the problem. Region-Based Convolutional Neural Networks (R-CNN) is a popular family of detection models, suggesting regions of interest as a first step and then classifying or discarding those regions for the final detection prediction. In order of appearance and improvement, mention may be made of R-CNN [31] (each region is classified separately), Fast R-CNN [30] (each region is classified from the same feature extractor), Faster R-CNN [75] (the region proposal step is now a Deep Learning model), and more recently Mask R-CNN [37] that adds a segmentation prediction. Another approach, famous for its unified architecture and its speed allowing real-time object detection, is the You Only Look Once (YOLO) [74] architecture. The idea is to divide the image into a grid and predict bounding boxes (and labels) for each grid square, thus removing the region proposal part of previous methods and enabling much faster inference. A major drawback of YOLO compared to previous models, is its difficulty to detect small and closed objects due to its spatial constraint. This method is further refined with improved YOLOv2 [72], and YOLOv3 [73], which extracts three different feature maps for detection at three different scales allowing the detection of objects of more variable size than the previous versions. Single Shot MultiBox Detector (SSD) [53] takes a quite different but still similar direction. The network computes different feature maps,

each associated with a set of default bounding boxes of various dimensions and aspect ratios, intending to detect objects of diverse sizes. Finally, CenterNet [20] improves the CornerNet approach [51] by adding, among other things, center-points detection.

2.2 Deep Learning in mammography: state of the art

There is a great interest in applying Deep Learning techniques for breast cancer detection. Only a few heterogeneous datasets are publicly available (e.g., CBIS-DDSM [52], INbreast [62], BCDR²), but they still allow a fair comparison of several proposed methods. Recently, the Digital Mammography Dream Challenge³ motivated candidates to develop and assess algorithms for risk stratification of screening mammograms, resulting in novel approaches. The one proposed by Ribli et al. [77] is a good example: a combination of the state-of-art detection algorithm Faster-RCNN with classification according to the malignancy at the end of the process, obtaining a true positive rate (or sensitivity, see Section 2.3.6) of 0.90 at a 0.3 false positive rate (or specificity, also see Section 2.3.6). Previously, Akselrod-Ballin et al. [3] developed an ensemble of different Faster-RCNN models, each specialized in detecting certain types of breast abnormalities with a CNN classifier, achieving a sensitivity of 0.93 at 0.56 specificity.

Other approaches are also being investigated, like the one proposed by Lotter et al. [56], with a CNN patch-based classifier, tackling one of the main problems in medical imaging: the high-resolution of the images or also the one studied by Dughel et al. [19] with a cascade methodology for detecting, segmenting, and classifying breast masses as malignant or benign with a sensitivity of 0.98 and specificity of 0.7. Kooi et al. [47] compare traditional CAD systems with a patch-level trained CNN and study the impact of adding hand-crafted features in the CNN model decision, showing that they outperform traditional state-of-the-art CAD and reach similar performances as radiologists, with an area under the receiver operating curve (AUC, see Section 2.3.6) of 0.88. These results demonstrate that great progress can still be made in the field.

Agarwal et al. [2] propose to use this patch approach in a detection framework, using and combining the classified patches for interpolation and final detection of masses. They study the performances for different sizes of lesions, resulting in

²<https://bcdr.eu/>, Accessed 2021-15-4

³<https://sagebionetworks.org/research-projects/digital-mammography-dream-challenge/>, Accessed 2020-05-10

a specificity of 0.97 at 0.5 sensitivity for medium and large lesions. In comparison, smaller lesions have a specificity of 0.89 at 0.5 sensitivity, calling for more improvements. Shen et al. [88] propose to combine these patches in an end-to-end model, with a pretrained backbone for ROI (region of interest) classification and a classifier head assessing the presence of a tumor in the whole mammogram.

Yala et al. [112] use the well-known Resnet18 network to assess cancer risk in the next five years by looking at the whole mammogram at once. They use a hybrid DL model to add other information than the one present in the mammogram, achieving an area under the curve (AUC) of 0.79 for premenopausal patients and AUC of 0.70 for postmenopausal patients. It shows great improvements when adding the family history of breast or ovarian cancer in the decision process. Indeed, the risk of breast cancer nearly doubles if the patient has a first-degree relative (mother, sister, daughter) who has been diagnosed with breast cancer [9].

A significant problem in the application of Deep Learning for medical imaging is the lack of data and more specifically the lack of annotations. Chougrad et al. [16] study the possibility of using transfer learning from natural images. They improve the performances of state-of-the-art pretrained models by fine-tuning them on public datasets, showing that transfer learning, which is still studied and the subject of significant debate, is still noteworthy at present.

2.3 Experiments framework

This section presents the models, the datasets, and the models' evaluation methods used in the following experiments.

2.3.1 Model architecture: Classification

We use the ResNet architecture [36], a model that was designed to tackle the gradient vanishing problem by introducing identity shortcut connections that skip one or more layers. Figure 2.1 shows the idea of a shortcut, allowing the network to easily learn the identity function (layers that replicate values). By this fact, a deeper network should not perform worse than its shallower version. Moreover, these connections allow the gradient to better propagate to the first layers, making the whole model easier to train. Formally, the output of one building block (stacked layers with a shortcut connection) is:

$$y = \mathcal{F}(x, W_i) + x$$

with y and x respectively the input and output of the building block, W_i the parameters of the layers of the building block, and $\mathcal{F}$ the corresponding function computed by these layers.

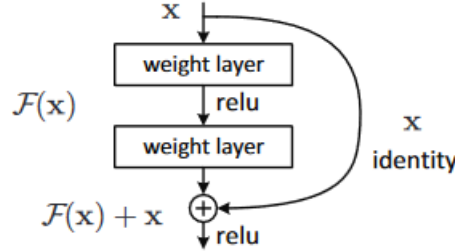


Figure 2.1: The shortcut connection introduced in ResNet. Source: He et al. [36].

Figure 2.2 shows the various increasingly deep architectures of the ResNet network. Due to the limited computational capacity and the important number of experiments to realize, we will focus on ResNet18 (18 layers).

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$
conv4_x	14×14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$
conv5_x	7×7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

Figure 2.2: The different Resnet architectures. Source: Kaiming He et al. [36].

In addition, because of the limited size of the dataset, we will use the pretrained network available in the Pytorch library (when specified), remove the fully connected layers and add a new one with a softmax function to classify two or three classes depending on the task. The architecture is shown in Figure 2.3, each arrow in an arc represents a shortcut connection. Given that pretrained networks available from the Pytorch library are designed to receive as input RGB images (three channels), we stack the same grayscale image three times before passing it through the network.

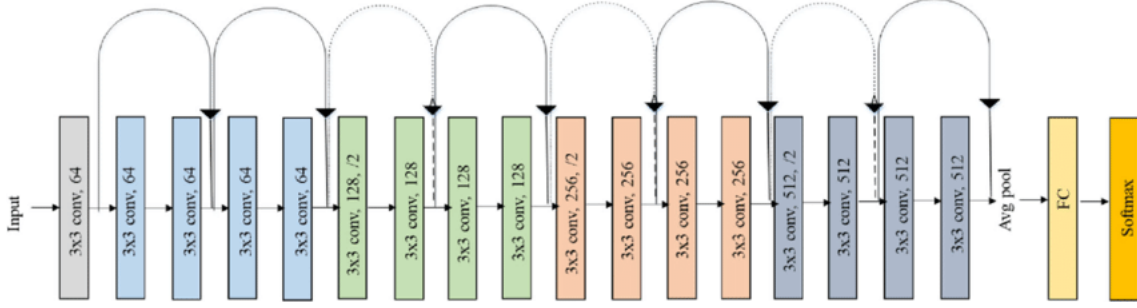


Figure 2.3: The ResNet18 architecture. Source: Ramzan et al. [70].

2.3.2 Model architecture: Detection

For the last experiment of Chapter 2, we use one of the best-known detection networks called Faster R-CNN [75]. Choosing the right architecture is difficult and often requires a compromise between accuracy and speed [41]. Since we do not restrict ourselves to real-time detection, we can choose a slightly more complex architecture than YOLO. Faster R-CNN applications and implementations are well documented and its two-stage inference process is suitable to the detection of small objects [41] as masses in mammography. The architecture is summarized in Figure 2.4. Faster R-CNN can be decomposed into three sub-networks: a feature extractor (or backbone), a region proposal network (RPN), and a head.

Feature extractor

Faster R-CNN has a backbone (in our experiments, we will use ResNet 18 from which we removed the average pooling layer and the fully connected layer) computing features from the whole image. Thus, the original image is downsampled by a factor of 32 at the end of the feature extractor. Then, a vector is extracted from these features for each prediction of the region proposal network. This method was introduced in Fast R-CNN to avoid multiple computations for each region as was the case in their predecessor R-CNN. This method has also the advantage of using more spatial information.

Region proposal network

The region proposal part in Faster R-CNN, unlike its predecessors which were based on a fixed algorithm like Selective search, is based on a Deep Learning network. A crucial implementation point is the use of anchor boxes. Faster R-CNN uses anchor boxes of three aspect ratios and three scales, such that for each pixel of the feature map there are nine anchor boxes. Instead of a regression problem, the problem of predicting bounding boxes is reduced to an offset prediction for each

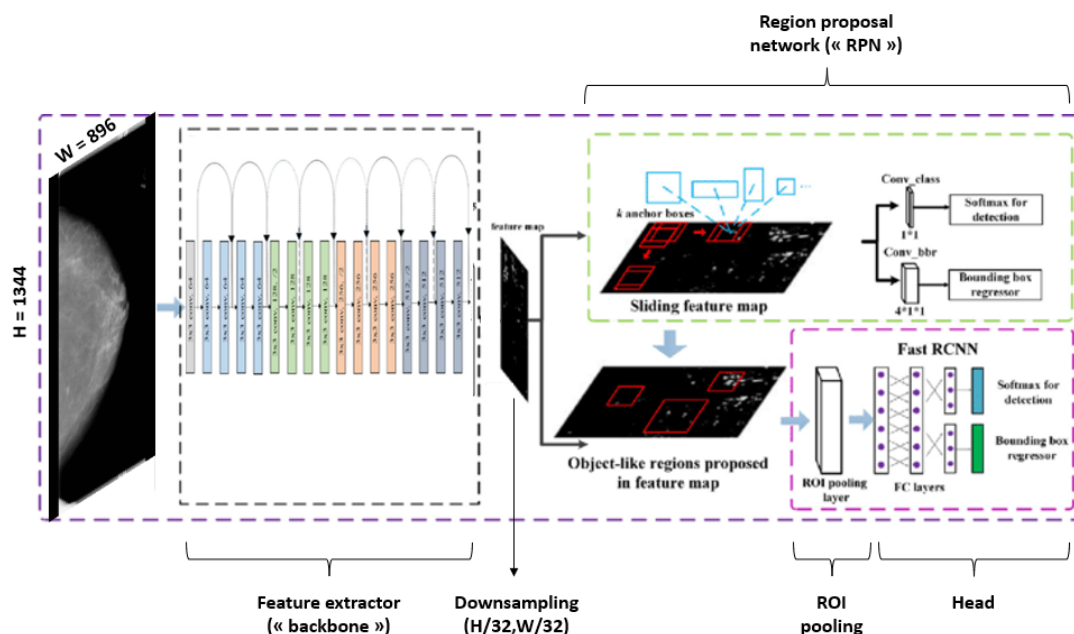


Figure 2.4: Faster R-CNN architecture for mammography. Figure adapted from Deng et al. [18], Ramzan et al. [70].

possible anchor box.

The region proposal network has two terminations, one for the bounding boxes (with a smooth L1 loss) and another one to give a confidence level for the presence of an object (binary classification with a cross-entropy loss). Finally, as the number of proposed bounding boxes can be very high, the authors use another well-known trick in detection called non-max suppression. The idea is to compare each box with the others and remove it if 1) it has an overlap greater than a certain IoU threshold (see Section 2.3.6, often set to 0.7 during training) with one of these other bounding boxes and 2) its confidence level is lower than this second bounding box.

Head

For each proposed region, a fixed size feature vector is extracted thanks to a ROI pooling layer which is basically a max-pooling layer applied to each feature map around a given area (the area that corresponds to the RPN prediction). The idea is to resize each region before passing it through the head since they can be very different in size. The head is composed of fully connected layers and ends with two terminations. The first one classifies the region into background or one of the possible classes (with a cross-entropy loss) and the second one predicts a bounding box to refine the initial bounding box guess of the RPN (with a smooth L1 loss).

2.3.3 Dataset presentation

Several mammography datasets are publicly available. One of the most used in the literature is the CBIS-DDSM (Curated Breast Imaging Subset of DDSM) dataset [52], which contains 2,620 scanned film mammograms. The distribution of benign and malignant cases is shown in Figure 2.5 where we can note that some scans show both abnormalities and that the four sets are overlapping.

	Benign Cases	Malignant Cases
Calcification Training Set	329 cases (552 abnormalities)	273 cases (304 abnormalities)
Calcification Test Set	85 cases (112 abnormalities)	66 cases (77 abnormalities)
Mass Training Set	355 cases (387 abnormalities)	336 cases (361 abnormalities)
Mass Test Set	117 cases (135 abnormalities)	83 cases (87 abnormalities)

Figure 2.5: Repartition of benign and malignant cases in the CBIS-DDSM dataset. Source: Lee et al. [52].

Moreover, a region of interest (ROI) segmentation is available for each abnormality and the two views (CC and MLO) are available for each patient’s breast as shown in Figure 2.6.

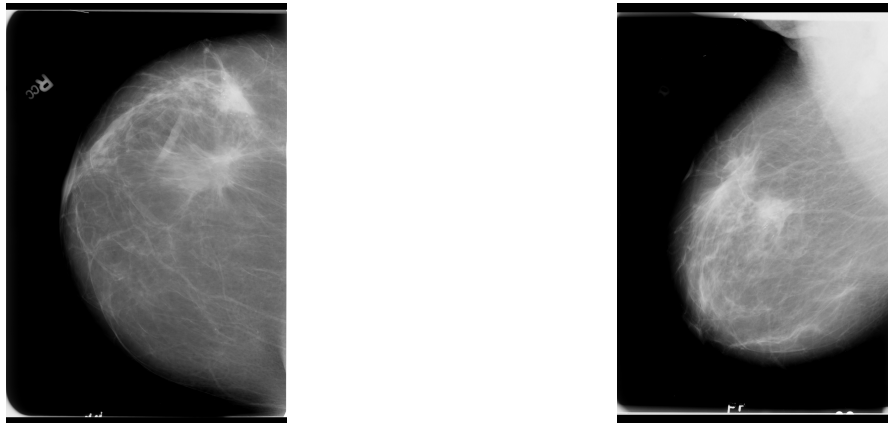


Figure 2.6: Two mammograms of the same breast: CC (left) and MLO (right) from the CBIS-DDSM dataset.

2.3.4 Dataset preparation

Figure 2.7 shows the different steps for the creation of two datasets: a patch dataset and a whole-image dataset. The mammograms are padded and resized to the size 1344x896, keeping the aspect ratio of the image which is important since the

shape of a tumour is highly discriminative for its malignancy (see Chapter 1). The original separation between training and test samples is kept and a validation set is randomly selected as a subset of the original training set to keep the experiments' results comparable with the other papers. For the whole-image dataset, the two different views (MLO and CC, see Section 1.4) from the same patient are kept in the same set to avoid bias in results.

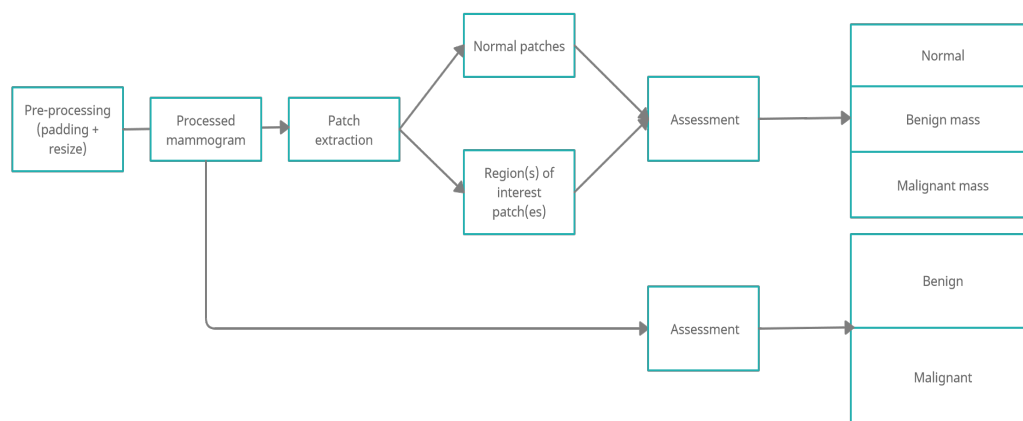


Figure 2.7: Creation of the dataset pipeline for patches and the whole image.

For each mammogram, one patch per region of interest (mostly one per image) and two normal patches are extracted. The two normal patches are randomly chosen. The first one is constrained to have 99% of its pixels representing the breast, and the other is randomly constrained to 0%, 25%, or 50% of its pixels representing the breast. This choice was made to have a majority of patches representing breast tissue but also other ones with the contour of the breast (thus with dark pixels) and finally a small subset of them representing the mammography background. Figure 2.8 shows two patches of the dataset, one including a benign mass and the other a malignant one.

After processing, the whole-image dataset splitting is shown in Table 2.1 (a mammogram is classified as malignant from the moment it presents at least one malignant mass). For each mammogram, bounding boxes annotations of its abnormalities (benign or malignant) are computed from the ROI segmentation of the original dataset. These annotations will serve to train our Faster R-CNN model. As presented in Section 2.3.2, anchor boxes are a key part of current detection models. Therefore, we computed the average mean and standard deviation of the bounding boxes' height and width, the results are shown in Table 2.2. It will be useful to better estimate the anchor boxes ratios and scales parameters of the RPN network.

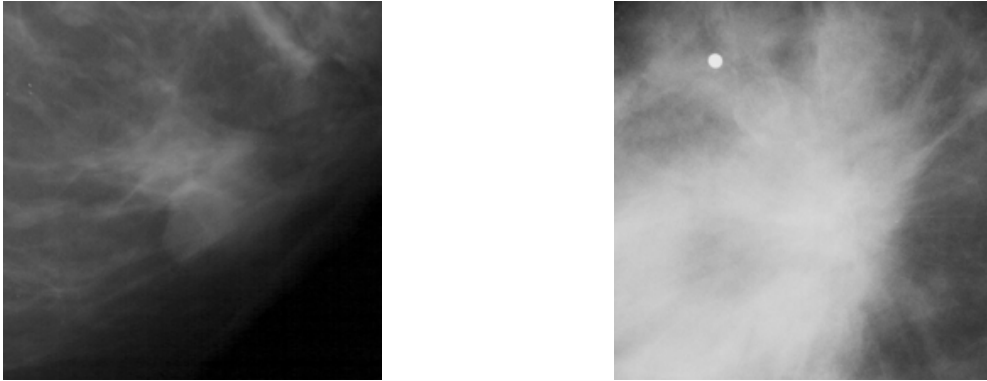


Figure 2.8: A benign mass (left) and a malignant mass (right).

	number of benign mammograms	number of cancerous mammograms
training set	453	479
validation set	128	106
test set	203	145

Table 2.1: Whole-image dataset division.

	mean	standard deviation
height (number of pixels)	84.43	37.34
width (number of pixels)	84.98	36.79

Table 2.2: Bounding boxes format.

The patch dataset division is shown in Table 2.3. The training set/validation set of the patch dataset is further randomly divided into a training set (80%) and a validation set (20%).

	number of normal patches	number of benign patches	number of malignant patches
training set/validation set	2462	632	621
test set	722	218	147

Table 2.3: patch dataset division.

2.3.5 Lack of data and overfitting

One of the main challenges in machine learning in general, and in Convolutional Neural Networks in particular, is to avoid overfitting. A model overfits when it is too closely aligned to the training samples; it learns data patterns but also other information not inherent to the original problem, such as noise, reducing drastically its predictive power. Overfitting can happen notably when the network's capacity, i.e., its number of trainable parameters, is too high compared to the number of training samples. Because the task of classifying mammograms is complex, the network must learn a wide range of features, which requires a relatively high capacity. To tackle the overfitting problem, several solutions have appeared in the literature.

- Dropout [96] is a regularization technique that randomly sets to zero neurons with a given probability during training. Intuitively, this forces the network not to rely on a subset of neurons to make its decision, and thus to learn more robust and general features.
- Batch normalization [42] is another regularization technique already implemented in ResNet networks. It normalizes (re-centers and re-scales) the activation values of a layer allowing a faster and more stable training procedure.
- Weight decay is often used to add a complexity penalty in the loss function, i.e., to penalize the model when weights take too high values in our case. It can have a regularization effect by preventing the network to adjust its weights too well on the training set and can improve its generalization ability.
- Transfer learning [105] is the fact of storing knowledge gained from one problem and applying it to a related but different problem. Pretrained weights on ImageNet for ResNet18 are available from the Pytorch library.
- Data augmentation tackles the problem of lack of data at the root and increases the training set's size by applying transformations directly on the data points to create new ones. Data augmentation can act as a regularizer, preventing the network from learning the training samples too well.

2.3.6 Performance measures

To measure the performance of the models, the confusion matrix is often used. It is a summary of the true positive (TP), false positive (FP), false negative (FN), and true negative (TN) classified samples. Their meaning appears clearly in Figure 2.9.

	Actual positive (1)	Actual negative (0)
Predicted positive (1)	TP	FP
Predicted negative (0)	FN	TN

Figure 2.9: The confusion matrix. Source: Statistical Odds & Ends [97].

It allows notably to measure the accuracy of the model, i.e., the proportion of samples correctly classified by the model.

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

However, in the case of class imbalance, which is often the case in disease detection, the accuracy can be overly optimistic. To address the class imbalance problem, balanced accuracy is preferred (if the detection of each class is equally important).

$$Balanced\ Accuracy = \frac{Sensitivity + Specificity}{2}$$

Sensitivity is the proportion of positive samples correctly identified as positive compared to all positive samples.

$$Sensitivity = \frac{TP}{TP + FN}$$

Specificity is the proportion of negative samples correctly identified as negative compared to all negative samples.

$$Specificity = \frac{TN}{FP + TN}$$

Models that are trained to detect diseases often face a trade-off between 1) detecting many positive cases at the risk of classifying many negative cases as positive and 2) classifying not too many negative cases as positive at the risk of missing a significant part of the positive cases.

Area Under Curve (AUC) can be used to choose the right threshold at which the samples are discriminated, i.e., the minimal threshold at which the model predicts

positive cases. AUC is based on the True Positive Rate (Sensitivity in the literature) and True Negative Rate (Specificity). AUC is the area under the receiver operating characteristic (ROC) curve, the curve plot of 1-specificity versus sensitivity at varying threshold values. The greater the value, the better the model's performance (with $AUC = 0.5$ corresponding to a random classifier).

For a multi-class problem, the one-versus-all macro-average AUC computes the AUC of each class versus the others and then takes the average. The one-versus-all micro-average AUC is nearly the same but each class contributes by weight according to its size. Hence, macro-average will be preferred since cancer and benign cases are as important to detect as normal cases, even though they are present in a smaller proportion in the dataset.

$$AUC_{macro} = \frac{\sum_{i=1}^C AUC_i}{C}$$

with C the total number of classes to classify

$$AUC_{micro} = \frac{\sum_{i=1}^C c_i * AUC_i}{\sum_{i=1}^C c_i}$$

with c_i the number of samples of class i .

For detection models, other metrics are used to assess the quality of the predicted bounding boxes. Mean average precision (mAP) is often chosen in research papers; it is calculated by taking the mean of the average precision (the area under the precision-recall curve) at varying IoU (intersection over union) thresholds. The intersection over union is defined as:

$$IoU = \frac{\text{Area of overlap}}{\text{Area of union}}$$

The recall is the sensitivity (or true positive rate), and the precision is the number of correct bounding boxes (for a given IoU threshold) over all the predicted bounding boxes for a given class:

$$Precision = \frac{TP}{TP + FP}$$

The mAP measure can be computed only for a given IoU threshold, which will be the case for our Faster R-CNN network below. The specific IoU threshold value is specified as $mAP@IoU_{threshold}$.

Detection models often have the task of classifying the detected objects. Like the ROC curve for the classifier, we would like to measure which threshold offers the best compromise between detecting a large number of objects at the cost of a low

number of false positives per image. The free-response ROC curve (FROC) answers this question for each decision threshold. It is characterized by the proportion of identified objects (sensitivity) and by the average number of false positives per image.

2.4 Classification networks

Now that we have defined the experimental settings, we evaluate a first approach that classifies the entire mammogram at one time as benign or malignant.

2.4.1 Whole-image classifier

As a baseline, it is interesting to measure the performance of a classifier that receives the whole mammography from the whole-image dataset introduced in Section 2.3.4 as input. The pretrained ResNet18 is used and the average pool layer (see Figure 2.3) is modified to have an output size of 512x6x4 (number of channels x height x width) to better handle large images. Moreover, random data augmentation on the fly were performed on the training set: rotation (between -25 degrees and 25 degrees), scaling (with a factor between 0.8 and 1.2), horizontal and vertical translation (with an offset between -0.1 and 0.1), brightness adjustment (between -10% and 10%), horizontal flipping (with a probability of 0.5). After a random search performed on the weight decay, the learning rate, and the dropout rate, this configuration appears hard to train. We propose a 4-stage training:

- learning rate at 10^{-3} to only train the fully connected layer for 5 epochs.
- learning rate at 10^{-4} to train the two last building blocks (with channel size 512) and the fully connected layer for 15 epochs.
- learning rate at 10^{-4} to train the full network for 10 epochs.
- learning rate at 10^{-5} to train the full network for 20 epochs.

The final accuracy on the test set is 0.652. Longer training in this configuration does not lead to major improvements. It appears that this network is not suitable for processing such images; see the learning curve in Figure 2.10. It shows high variability in results even for a slow learning rate. We can deduce from these curves that the classification network has a lot of difficulties computing strong and general features, as the accuracy can change abruptly with only one step of training.

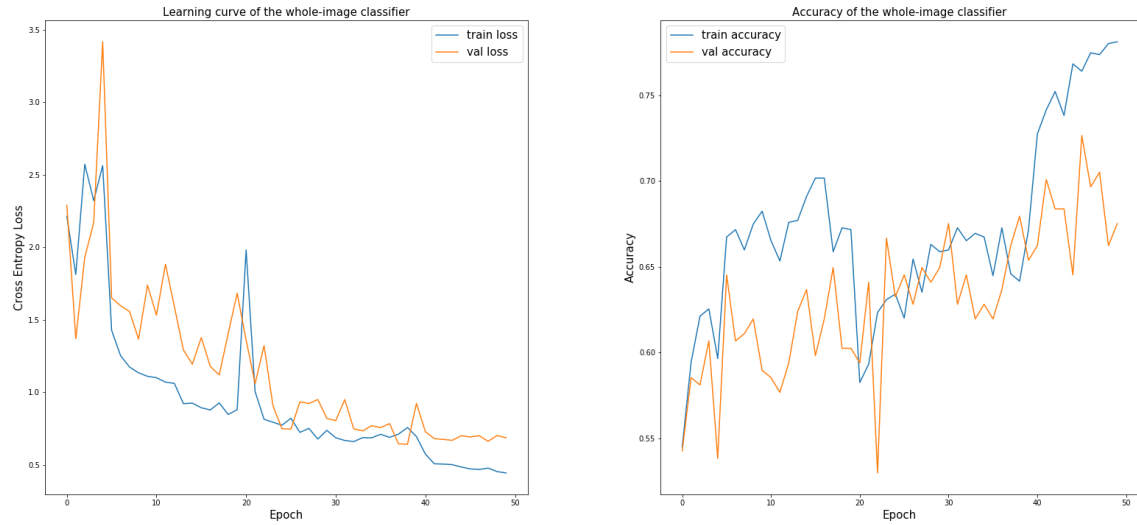


Figure 2.10: Learning curves for the whole-image classifier.

Several reasons could explain these poor results:

- The pretrained ResNet18 is trained on 224×224 images, i.e, much smaller images than the ones used above. A network adapted to larger images should be used.
- Mammograms are complex images, with only a small portion of pixels that are decisive, see Figure 2.11. Abnormalities can be very small compared to the whole image, making it harder for the network to compute discriminative features based on such a small area.



Figure 2.11: Example of a mammogram to classify with its associated mask (malignant mass).

We use Grad-CAM [83] to give us an insight about the most informative area for the model. Figure 2.12 shows the result for two mammograms of the test set. We observe that the network is not able to point out the regions of interest, confirming the difficulty of our model to detect the small interesting areas from which to extract the discriminative features in large images. These observations lead us to a patch approach where only a small area of the mammogram has to be classified and a detection approach where coordinates of the regions of interest are fed into the model.

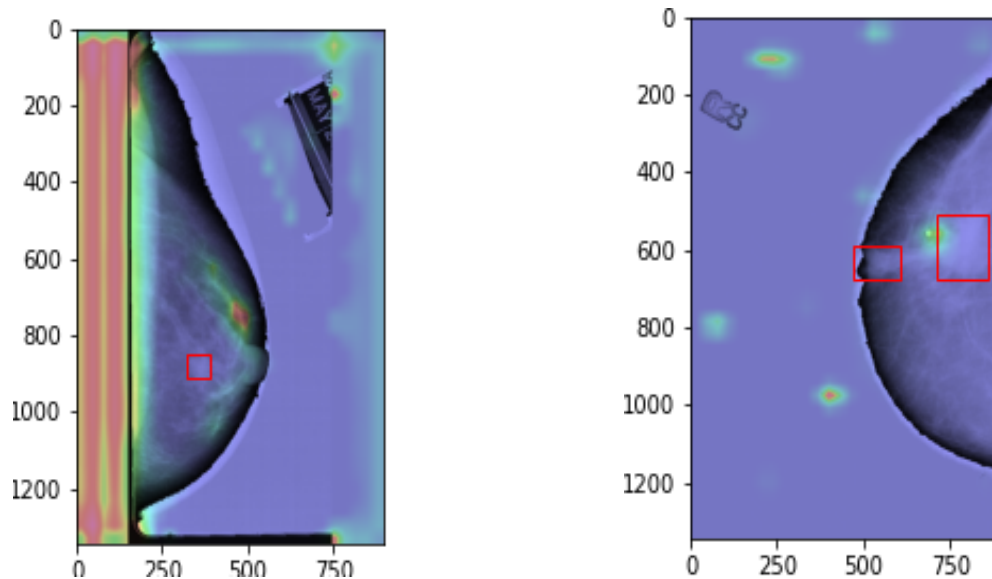


Figure 2.12: Grad-CAM results for a mammogram with a benign mass (left) and a mammogram with two malignant masses (right), blue color corresponds to low gradient values and red to high gradient values, the red rectangles embrace the masses.

2.4.2 Patch approach

In this approach, the network to train is classifying patches of images, as shown in Figure 2.13. We take the pretrained ResNet18, remove the fully connected layer and add a new one to classify images into 3 classes. The best model is found with a random search (on the batch size, the weight decay, the steps at which the learning rate is reduced, with and without dropout before the fully connected layer). To avoid overfitting, random data augmentation was performed on the fly during training: rotation (between -25 degrees and 25 degrees), scaling (with a factor between 0.8 and 1.2), brightness adjustment (between -10% and 10%), horizontal and vertical flipping (with a probability of 0.5 for each). The

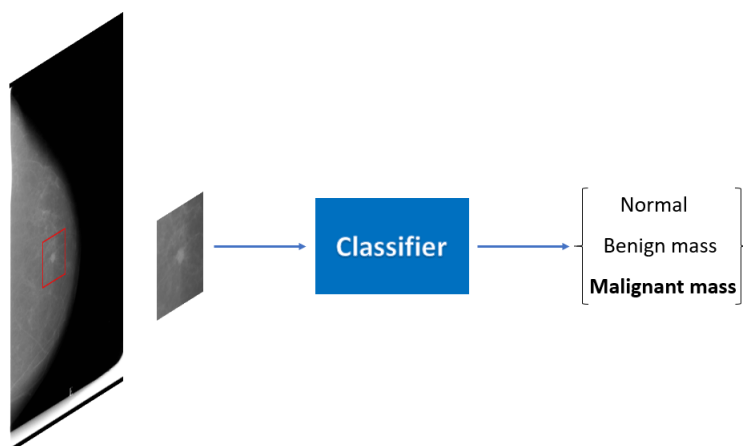


Figure 2.13: A patch approach to classify regions of interest (malignant case).

best-performing hyperparameters are presented in Table 2.4. In addition, the learning curves are shown in Figure 2.14. We observe that the learning rate decay at epoch 20 engenders a reduction in the validation loss and increases the accuracy. The final model achieves an accuracy of 0.874 and a balanced accuracy of 0.783.

Hyperparameter	Value
epochs	65
optimizer	Adam
learning rate (initialization)	10^{-3}
learning rate decay	0.1 at epoch 20 and 60
weight decay	$5 * 10^{-4}$
dropout	0
batch size	32

Table 2.4: Hyperparameters for the patch classifier.

This result is explained by Figure 2.15; normal patches (BI-RADS 1) are indeed easier to discriminate than benign and malignant ones (class 1 and 2 respectively). This observation gives us insight into the ability of the neural network to classify masses.

This leads us to our third approach, which combines the whole-image classifier and a patch-based approach to detect masses in a mammogram. Prior to this, we do an experiment on calibration and ensemble methods, which will be used in Chapter 4.

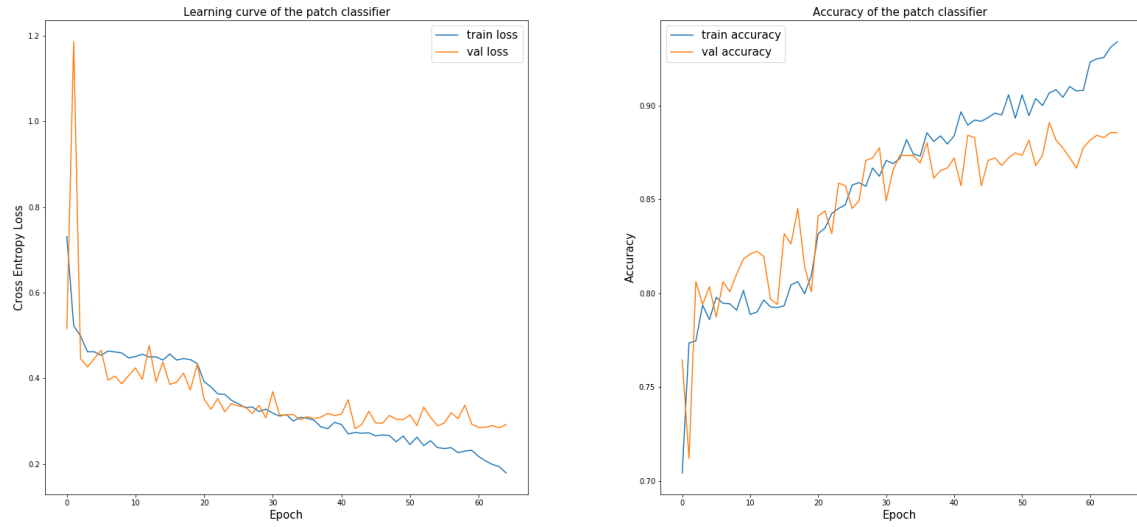


Figure 2.14: Learning curves for the patch classifier.

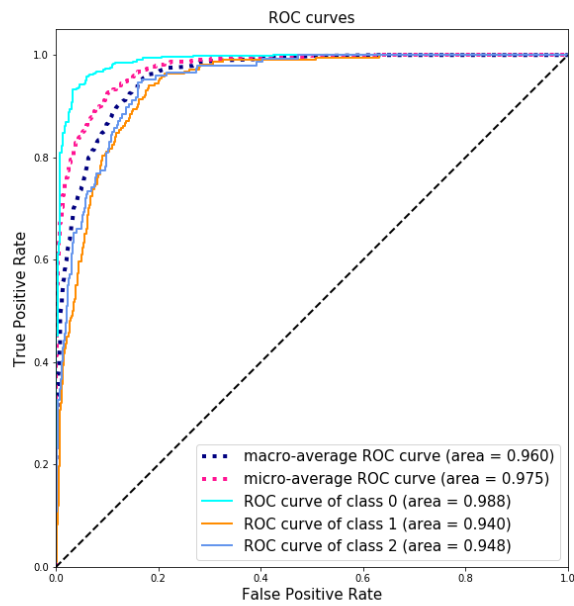


Figure 2.15: ROC curve of the patch classifier based on ResNet18.

2.4.3 Calibration

A major drawback in using Deep Learning algorithms is that they are not good at telling their confidence. They tend to be overconfident [33], which can drastically reduce the trust of the senologist in the prediction of the model in case of errors. A well-calibrated model will associate a high probability to a prediction for which it is certain and a low probability to a prediction for which it is uncertain. But more importantly, we expect a low probability output when the model is wrong. Therefore a good calibration of the model used can be decisive, especially in an Active Learning framework (see Chapter 3 and 4). Some of the methods presented in these chapters attempt to measure the uncertainty about their predictions in order to determine which samples will be the most interesting ones to annotate. One way to obtain more calibrated models is to use ensemble techniques [50].

The experiment below uses three models from the random search conducted in Section 2.4.2 (they all had almost similar performance on the validation set). It is important to keep in mind that all these networks were trained from the same pretrained Resnet18 (therefore they have the same weight initialization) and their training procedures were quite similar, inducing less diversity in the ensemble.

Figure 2.16 shows that the ensemble method is less confident when making an error than a single model. However, the ensemble seems also less confident when it makes a correct prediction. The gap between the confidence level for a good prediction and a bad prediction is nearly the same in both approaches.

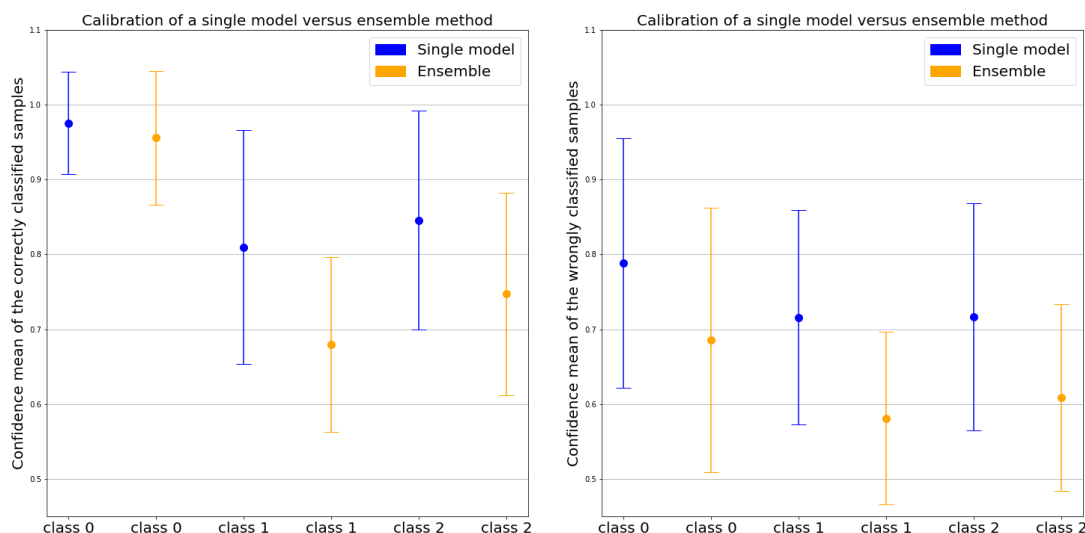


Figure 2.16: Calibration of a single model versus an ensemble method.

What is important to note here, is that the models have a lower confidence level when they make an error. This ability of the models to estimate when it makes an

error will be useful later to measure its uncertainty and will be at the center of one of the Active Learning methods studied in the following chapters.

2.5 Detection networks

As we observed above, the CNN based on pretrained ResNet18 can achieve better performance when it looks closer to the abnormality, demonstrating its ability to discriminate masses despite the observations for the whole-image classifier. Instead of surrounding manually the interesting areas, we would like our model to be able to detect them by its own. We use the results from previous experiments to build a Faster R-CNN network⁴. The IoU threshold at which a predicted bounding box is considered as correct (the overlap with the ground truth bounding box) is set to 0.5, since [77] observe that a too high IoU engenders too few positive samples in minibatches and unstable training.

Unlike the training procedure of the original paper (in four steps), we choose to train our Faster R-CNN in two steps for the sake of simplicity:

- Train the feature extractor network (see Section 2.4.1).
- Train jointly the feature extractor, the region proposal network, and the head.

Early stopping was used to keep the best-performing network according to the validation set. The learning rate was decreased by a factor 0.1 each 60 epochs and the best performing model was reloaded before each of these decays. The batch size was fixed to 1 to keep the implementation simple and reduce GPU usage. The hyperparameters are presented in Table 2.5.

The main modifications with respect to the original paper are the anchors scales (reduced by a factor 2 since the objects to detect are much smaller than for many other tasks). Moreover, the ROI pool size was reduced for the same reasons and because the output of our backbone is downsampled by a factor 32 instead of a factor 16.

The learning curve is shown in Figure 2.17. We observe that a lower validation loss is not always correlated to a higher mean average precision. A more precise training procedure could be investigated to achieve better performance. On the test set, the model reaches a map@0.5 of 0.295. The relatively low performance can be explained again by the small size of some masses and therefore the difficulty to locate them.

⁴Our implementation was greatly inspired by the work of Yun Chen: <https://github.com/chenyuntc/simple-faster-rcnn-pytorch>

Hyperparameter	Value(s)
anchor scales	56^2 , 128^2 , 256^2
anchor ratios	0.5, 1, 2
ROI pool size	2
non-max suppression threshold (training)	0.7
non-max suppression threshold (test)	0.1
IoU threshold	0.5
number of neurons (fully connected layer 1)	1024
number of neurons (fully connected layer 2)	512
dropout rate (head)	0.5
optimizer	SGD
momentum	0.9
learning rate	10^{-3}
learning rate decay	0.1 at epoch 59 and 119
weight decay	10^{-4}

Table 2.5: Hyperparameters of the Faster R-CNN network.

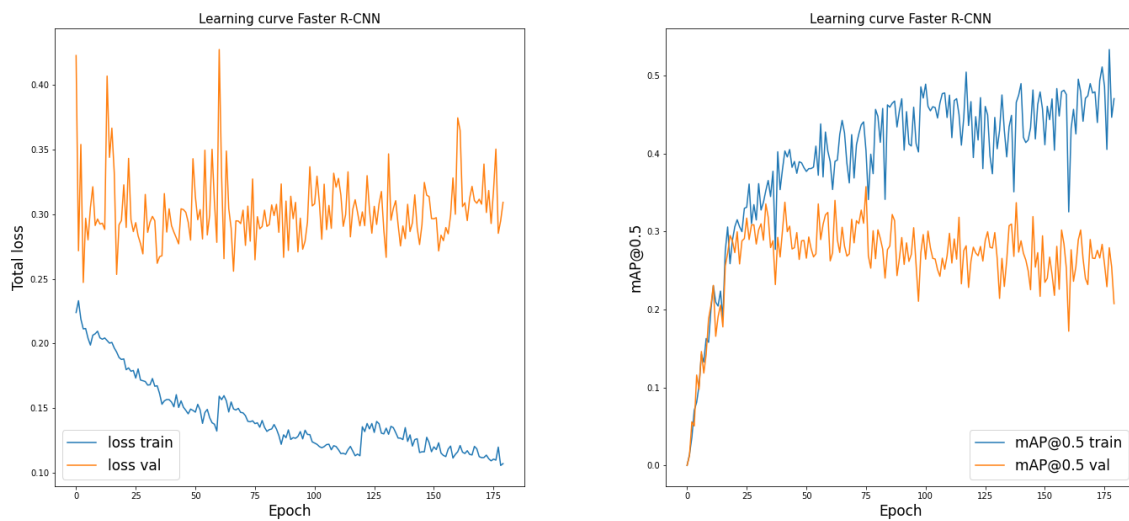


Figure 2.17: Learning curves for Faster R-CNN.

Finally, we assess our model with the free-response ROC curve for three different non-max suppression thresholds, see Figure 2.18. A lower non-max suppression threshold engenders better results for high score thresholds. We compare our results with a paper of the literature using Faster R-CNN at same IoU threshold (0.5) in

Table 2.6 for the non-max suppression threshold set to 0.1 (the same value is used at test time in the paper).

Sensitivity	at 0.5 FP average	at 1 FP average	at 2 FP average
Our model	0.463	0.514	0.560
Results from Famouri et al. [23] (without hard sampling)	0.39	0.59	0.7

Table 2.6: Comparison of performance of our Faster R-CNN with the literature.

Our model performs better at low false positives average per image but has much lower performance for higher false positives average. This difference can be explained by our 2-stage training procedure against a 4-stage procedure in [23] as presented in the original paper of Faster R-CNN. Moreover, the authors use pre-trained (on ImageNet) ResNet50 as backbone, allowing to extract richer features than our ResNet18. Additionally, they add a different matching criterion during training: each region proposed by the RPN is considered as positive if the IoU is greater than 0.5 and negative if lower than 0.3, the others are considered as neutral.

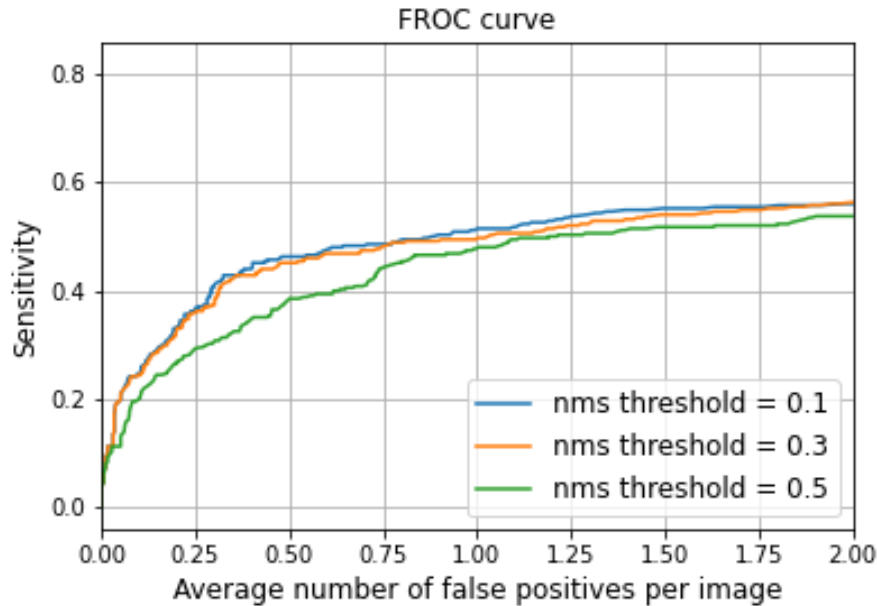


Figure 2.18: Free-Response ROC curve of our Faster R-CNN model for different non-max suppression thresholds.

Some results are shown in Figures 2.19 (correct result), 2.20 (correct detection but with an FP) and 2.21 (incorrect result). The red rectangles are the ground truth bounding boxes while the green ones are the bounding boxes predicted by the model.

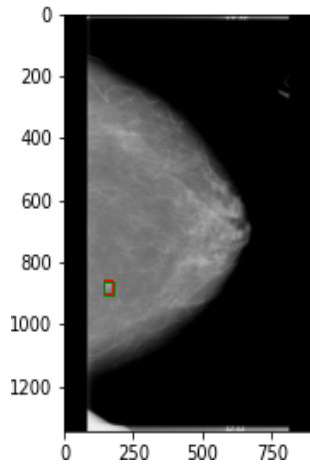


Figure 2.19: Good detection.

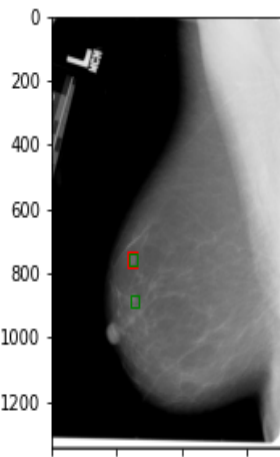


Figure 2.20: Good detection with an FP.

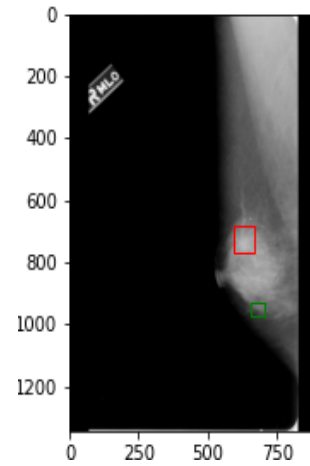


Figure 2.21: Abnormality not found.

2.6 Summary of Chapter 2

We presented several Deep Learning approaches to assess the presence of tumors in a mammogram. We observed that evaluating the entire mammogram at once causes difficulties during learning and does not seem to lead to substantial performance. Secondly, a patch-based approach proved that Deep Learning models are able to classify masses, which led us to a third approach, with a Faster R-CNN network, combining feature extraction of the whole image and proposal of regions of interest for a finer diagnosis. Compared to the results of a literature paper, we obtained a better sensitivity at a low average of false positives per image. However, a better training procedure and better selected hyperparameters still need to be investigated. Moreover, our experiments are limited in terms of validation, and our results could gain significance with cross-validation. This issue is discussed in Chapter 6.

Notations

In this section, the concept of Active Learning is formally defined and set up with appropriate notations.

- $\mathcal{X}$ is the data space, i.e., the set of objects called data points, $\mathcal{Y} = \{1, \dots, C\}$ the label space, and C the number of possible labels (or more specifically the number of classes in case of a classification task).
- x_i denotes the i th data point, the label of x_i is written y_i , ($y_i = j$ for $j \in \{1, \dots, C\}$ in the case of classification and a continuous value in case of regression tasks).
- $D = \{x_i\}_{i=1}^n$ denotes a dataset of n samples, $Y = \{y_i\}_{i=1}^n$ the corresponding labels, $\{\hat{y}_i\}_{i=1}^n$ the label predicted by the current model for the corresponding data points.
- D_{train} is the currently annotated samples of D and D_{pool} the unlabeled ones: $D = D_{train} \cup D_{pool}$. The current iteration may be specified by t , written D_{train}^t and D_{pool}^t at iteration t , meaning that D is fixed and that the model is trained iteratively with more and more labeled data.
- $\mathcal{W}$ denotes the parameters space of the model, i.e., all the values the parameters can take, and W the corresponding Random Variable.
- b is the budget of queries to ask an oracle at each Active Learning iteration.
- $p_{\Theta}(y = c | x_i, D_{train})$ is the probability associated by the model to the class c (or to a more general prediction in case of other problem than classification) for a given data point x_i after being trained on the set D_{train} with the model Θ , often written $p(y = c | x)$ for the sake of conciseness.
- $l(\cdot, \cdot, w) : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ is the loss function parameterized over the parameters w of the Deep Learning model.

Chapter 3

Active Learning

Machine learning models, especially Deep Learning ones, are demanding in large annotated data sets. Even if information is flooding, with more and more data collected, recent studies suggest that bigger-capacity networks with more training samples can still achieve better performance [58].

A big challenge in machine learning applications, at least for supervised learning and to a lesser extent semi-supervised learning, is therefore obtaining labeled data, which is often labor-intensive and time-consuming. This lack arises particularly in the case of annotation in medical imaging. Hence there is still a constant need to gather more data. Active Learning (AL) aims to partly solve this problem: to propose a framework that allows the model to choose which data it wants to be labeled.

Active Learning has been studied for several decades in the machine learning field [85]. However, the literature about Deep Active Learning, i.e., the application of Active Learning for Deep Learning, remains relatively sparse, mainly due to the complexity of such models. Understanding the behavior and the impact of labeled data on the training of the model and thus finding which is the optimal subset of labeled samples is challenging and still an open question. In most cases, it is answered by certainty or diversity measures, which allow the model to iteratively select the most informative or most representative samples according to some criteria. With a fixed budget (in time, in manpower), the network selects a few specific data points to be labeled, with the goal of achieving better performance than a random selection. Active Learning is therefore an iterative process in which a model is trained and can ask an expert, called the oracle, to annotate data from a pool of unlabeled data.

A visualization of what Active Learning wants to achieve for a classification task is shown in Figure 3.1. The first image is the whole dataset with binary labeling. The second image gives a possible decision boundary result if 30 samples to annotate are randomly selected, and the last image, the decision boundary if the 30 best

points to annotate are chosen. Intelligent sampling can have a tremendous impact in the case of sparse and expensive annotated data.

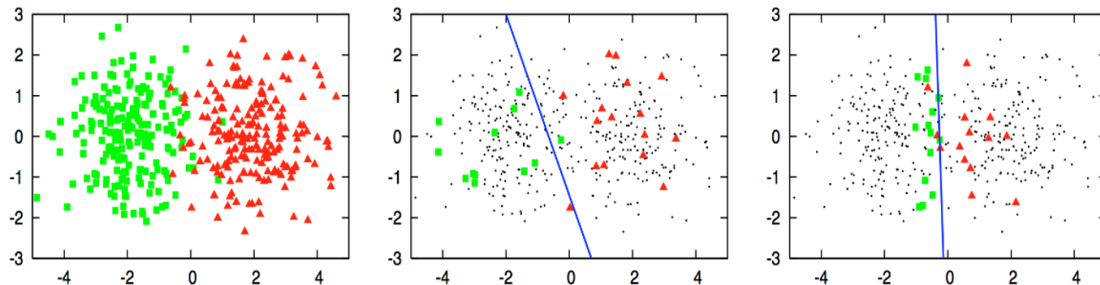


Figure 3.1: A concrete example of Active Learning. Source: Burr Settles [85].

In this chapter, we introduce the concept of Active Learning in a general way and how it can be applied to any machine learning model before presenting in depth the current techniques adapted to Deep Learning models.

3.1 Theoretical settings

3.1.1 Scenarios

Active Learning can be used in different scenarios depending on the task to perform.

- **Membership Query Synthesis:** In this scenario, the Active Learning algorithm generates its own unlabeled instances and queries an oracle for labeling. This scenario is particularly convenient when the number of available data points is small but can be difficult to apprehend when the generated data are not understood by the oracle.
- **Stream-based selective sampling:** The unlabeled data are continuously sent as a stream to a model that can either assign a label or ask an oracle to annotate it. In some cases, the model has one unique opportunity to see each instance.
- **Pool-based sampling:** The instances are selected from a pool of unlabeled data based on a given sampling strategy (assigning an informativeness score to each sample). The model tries to estimate how well it understands the sample or how much this sample will be useful. This scenario is the most common and the best adapted to the situation presented in this thesis since it allows the model to select a batch of samples simultaneously; it is also the most studied one in the literature.

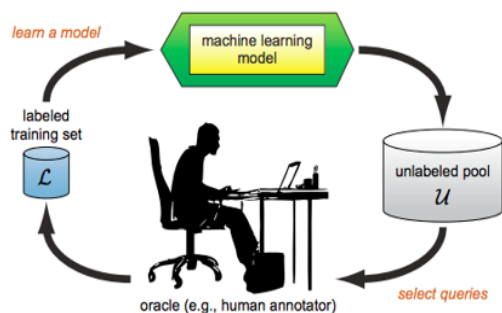
3.1.2 Workflow

There are many possible iterative processes, but the typical way to integrate Active Learning is to begin with a small amount of labeled data, the so-called seed. Indeed, it can be assumed that most tasks have already annotated data; otherwise, randomly sampled and annotated data may be used at the first iteration.

An iteration can be summarized in three steps:

- Train the model with the labeled data (and the unlabeled data in another learning approach than supervised learning like weakly-supervised learning or unsupervised learning).
- The model selects queries in the unlabeled pool.
- An oracle (e.g., a human annotator such as a radiologist) labels the data.

The model is retrained with the set of labeled data. This loop (see Figure 3.2) is repeated until a certain criterion is reached (performance, budget, etc.).



Algorithm 1 Active Learning framework

Require: *seed*, *unlabeled pool* (D_{pool}),
budget, *total budget*
while *total budget* has not been
reached **do**
 train model
 select *budget* samples
 oracle annotates the samples
end while

Figure 3.2: The pool-based sampling scenario. Source: Burr Settles [85].

3.2 Acquisition Functions and Querying Techniques

One of the most common querying strategies is uncertainty sampling, as presented in [85] and [86]. Other strategies based on the distribution of the data or their inherent features [86] try to select the most representative data; they are called diversity sampling. Another one developed in the literature is the Query-By-Committee (QBC) [86] [59], in which the agreement between several models is measured to choose data carrying the most information. Finally, some strategies attempt to measure the impact of labeling a specific instance on the model (Expected gradient

length/Expected model change [87]), on the error (Expected error reduction [78] [79]), or on the model's output (Variance reduction [117]).

In the next section, uncertainty sampling, diversity sampling, and the Query-By-Committee sampling are developed to give the reader intuition before diving into more appropriate solutions for the Deep Learning framework.

3.2.1 Acquisition functions

The Active Learning task can be formalized as an optimization problem in which an acquisition function f is minimized or maximized.

$$x^* = \operatorname{argmax}_{x \in D_{pool}} f(x) \quad \text{or} \quad x^* = \operatorname{argmin}_{x \in D_{pool}} f(x)$$

with the acquisition function taking as input the data of the unlabeled pool and the current model.

$$f(\cdot, w) : \mathcal{X} \rightarrow \mathbb{R}$$

The acquisition function is therefore a way to measure how interesting an unlabeled data point is and the impact it may have on the model if it was added to the training set (D_{train}).

3.2.2 Uncertainty Sampling

When a model is uncertain about its prediction for a given instance, this can be interpreted as new information that the model has not yet seen. Uncertainty can therefore be seen as a measure of the amount of information newly labeled data can bring.

Several uncertainty sampling strategies, with the appropriate acquisition function, can be used.

- **The least confident:** The idea is to select the sample for which the model is the least confident of its prediction.

$$x^* = \operatorname{argmax}_{x \in D_{pool}} 1 - p(\hat{y}|x)$$

with $\hat{y} = \operatorname{argmax}_{y \in Y} p(y|x)$, the predicted label for the data point x .

- **Margin sampling:** A simple improvement of the first acquisition function is to take into account the probabilities associated with the second most likely label. The idea behind this is that data with a small margin are more

ambiguous, so knowing the label of this sample will help the model to better discriminate these two classes.

$$x^* = \operatorname{argmin}_{x \in D_{pool}} p(\hat{y}_1|x) - p(\hat{y}_2|x)$$

$$\text{with } \hat{y}_1 = \operatorname{argmax}_{y \in Y} p(y|x) \text{ and } \hat{y}_2 = \operatorname{argmax}_{y \in Y \setminus \hat{y}_1} p(y|x)$$

- **Entropy:** Another improvement is to get information from all probabilities assignment of the model (in case of classification tasks, the model assigns for a given data point a value for each class that can be interpreted in some way as a probability that this data point belongs to this class). The entropy is a disorder measurement of the probability distribution. The more the probabilities are uniformly distributed, the higher the model is uncertain about its prediction and the higher is the entropy.

$$x^* = \operatorname{argmin}_{x \in D_{pool}} - \sum_{\hat{y} \in Y} p(\hat{y}|x) \log p(\hat{y}|x)$$

Inside machine learning models, two types of uncertainties can be distinguished. The epistemic uncertainty is the model uncertainty that can be reduced with more training points and the aleatoric uncertainty which is the uncertainty of the problem itself that can be seen as the noise inherent in the observations [44]. Some methods try to distinguish between the two types of uncertainty, epistemic being the one that can be actually reduced, but this question is not addressed in this master thesis

A crucial property of the Bayesian approach described later in Section 3.3 is the ability to approximate its uncertainty. By inference, it obtains a probability distribution over its parameters (posterior distribution), which can be computed with the Bayes' Theorem

$$p(w|D) = \frac{p(D|w)p(w)}{p(D)}$$

where $p(w)$ is the prior distribution (distribution before any data is observed) of the parameters of the model and $p(D|w)$ the sampling distribution (distribution of the observed data conditional on the parameters of the model), also called likelihood.

An important issue of these methods is that an uncertain sample can be marginal and will therefore not carry information about a richer region (it is not similar to other samples) and will not lead to a big change in the performance. Moreover, in CNN's training, optimization is often carried by passing batches through the

network. In a batch selection setting, these acquisition functions could choose interesting samples that are quite similar and will not represent the whole data space. These main issues bring to another strategy: diversity sampling.

3.2.3 Diversity sampling

These methods are based on the fact that it is often more interesting to know the label of a sample representative of a larger subset of the data. A known measure is a weighted sum of the similarities between a data point and all the other ones.

$$x^* = \operatorname{argmax}_{x \in D_{pool}} \phi(x) \left(\frac{1}{N} \sum_{i=1}^N \operatorname{sim}(x, x_i) \right)^\beta$$

with $\phi(x)$ an informativeness measure (e.g., an uncertainty measure), and β a parameter that weights the importance of the similarity measure and the information measure.

3.2.4 Query-By-Committee

This strategy involves a set of models called the committee, all of them trained on the labeled set (in the case of supervised learning). Each member of the committee has a vote and the sample on which the committee disagrees the most is considered as the most informative and is therefore chosen. A simple way to measure disagreement that reminds uncertainty sampling is the **vote entropy** :

$$x^* = \operatorname{argmax}_{x \in D_{pool}} - \sum_{\hat{y} \in \hat{Y}} \frac{V(\hat{y})}{Q} \log \frac{V(\hat{y})}{Q}$$

with Q the size of the committee and $V()$ the number of votes for the prediction $\hat{y}$. The **variation ratio** is a measure of the dispersion of a nominal variable. In this case, it corresponds to the proportion of labels that are not falling into the modal prediction (the most frequent prediction). Therefore, a high variation ratio indicates a greater dispersion in opinion.

$$x^* = \operatorname{argmax}_{x \in D_{pool}} 1 - \frac{f_m}{Q}$$

with f_m the number of predictions for the modal prediction.

Another way to quantify the disagreement between models is the **Kullback-Leibler divergence**. It is a measure of the difference between 2 distributions.

$$x^* = \operatorname{argmax}_{x \in D_{pool}} \frac{1}{Q} \sum_{q=1}^Q \sum_i p_{\theta_q}(y_i|x) \log \frac{p_{\Theta_q}(y_i|x)}{p_Q(y_i|x)}$$

with $p_Q(y_i|x) = \frac{1}{Q} \sum_{q=1}^Q p_{\theta_q}(y_i|x)$, Θ_q being the q th model of the ensemble.

3.3 Active learning in Convolutional Neural Networks: state of the art

In the past few years, deep neural networks have outperformed other machine learning models in many tasks. These achievements can be explained by the improvement in computing capacity and the ever-increasing access to more and more data. It is in this context that the application of Active Learning makes sense. However, the literature about Active Learning in Convolutional Neural Networks is relatively sparse and new. As described above, existing methods can be divided into categories (separated below into uncertainty and diversity measures) even if some of them can be interpreted as belonging to both of them.

3.3.1 Uncertainty sampling

These methods try to assess the uncertainty of the model's prediction, which can be an intuitive way to measure the information contained in the data point for the model. First of all, the different acquisition functions described in Section 3.2.2 can be used. However, many experiments show that simply taking the output of a deterministic CNN to compute these measures does not guarantee good results, as these models tend to be overconfident [33]. Moreover, the ability of deep neural networks to easily learn a random labeled training set with high confidence [116], or the fact that small perturbations can easily fool them in some examples [99] despite strong confidence in their predictions, shows the need for another approach to measure their certainty.

Bayesian Neural Networks

There is a lot of excitement around the application of Bayesian learning to the field of Deep Learning because most modern Deep Learning models cannot represent their uncertainty nor take advantage of the well-studied tools of probability theory. This lack has started to change following the recent developments of tools and techniques combining Bayesian approaches with Deep Learning ones.

Since Deep Learning models are not good in telling their confidence, the idea is to use specialized models like Bayesian Convolutional Neural Networks, such as the ones described in [26] [27].

Gal et al. [28] present an active sampling method based on this Bayesian approach. A Bayesian CNN is a CNN with a prior probability distribution over a set of model parameters $\omega = \{W_1, \dots, W_L\}$. They define a likelihood model

$$p(y = c|x, \omega) = \textit{softmax}(f^\omega(x))$$

with $f^\omega(x)$ the output of the model. Here softmax is used in the case of regression (other likelihood for another architecture or another task to perform could be used). To perform approximate inference, many tools developed in the field of Bayesian models could be used, but the authors focus on the stochastic regularisation technique known as dropout [96] and which is usable in practice for approximate inference in Deep Learning models [27].

This method is referred to as Monte Carlo dropout

$$\begin{aligned} p(y = c|x, D_{train}) &= \int p(y = c|x, \omega)p(\omega|D_{train})d\omega \\ &\approx \int p(y = c|x, \omega)q_\theta(\omega)d\omega \\ &\approx \frac{1}{T} \sum_{t=1}^T p(y = c|x, \hat{\omega}_t) \end{aligned}$$

with T the number of stochastic forward passes with Monte Carlo dropout from the approximate posterior and $\hat{\omega}_t \sim q_\theta(\omega)$ the dropout distribution.

Intuitively, it is not possible to generate all the possible networks for a given training set. The Bayesian approach approximates the posterior distribution on the model's parameters by an easier distribution (here a dropout distribution) to make it computationally feasible in practice. It generates several different networks by applying dropout with the purpose to compute a better-calibrated prediction and therefore better capture the model's uncertainty.

This method was also applied in object detection tasks like the one presented recently in [48] or [35].

A major problem with this method is the difficulty in adjusting the parameters properly [67] but also the calculation overhead during test time since it demands the drawing of a large number of samples to have a reasonable estimate of the certainty.

Another severe problem in uncertainty measurement is when a batch of data points needs to be selected. As these methods do not consider the similarities between the selected data, similar data can be all uncertain for a given model configuration. However, it could need only one of them to better understand this group of points.

Kirsch et al. [45] propose a new way to measure informativeness in Bayesian models by measuring not only uncertainty but also the mutual information between the potentially selected points. The authors base their method on an acquisition function called BALD that estimates the mutual information between the model predictions and the model parameters, which is shown to work well when acquiring each point individually [28].

The main contribution is the introduction of a new acquisition function that mea-

sures the mutual information between a joint of multiple data points and the model parameters called BatchBALD, and a greedy linear-time $1-1/e$ approximate algorithm to avoid the combinatorial computations.

Other methods taking the diversity of the data points into account will be described below.

Ensembles

Ensembles methods like [4] [50] can be used to measure uncertainty of the decision. This method is directly related to the query-by-committee measure in which the opinions of several models are compared. In this context, Bayesian networks inference might be interpreted as taking into account several models' predictions. With the same idea as above, Beluch et al. [4] use uncertainty measurement in ensemble methods by approximating

$$p(y = c|x, D_{train}) \approx \frac{1}{T} \sum_{t=1}^T p(y = c|x, \omega_t)$$

with T the number of models (and w_t their parameters) in the ensemble trained with the same dataset. They can be initialized with different weight initializations or even be based on different architectures.

The authors show that with a small number of networks (even three), their sampling method can achieve similar or even better performance than Monte Carlo dropout with a much larger number of forward passes. They explain this result by the fact that ensemble methods can generate predictions from much more diverse networks.

Finally, ensembles and Bayesian networks can be combined in deep probabilistic ensembles [15] to benefit from the advantages of both approaches.

Bootstrapping

Yang et al. [113] propose to measure uncertainty by using bootstrapping [22] in the context of segmentation based on fully convolutional network architectures. They train a set of models like in the ensemble method but restrict each of them to a subset of the training set. The final disagreement is computed as the mean uncertainty between the models. The authors show a high correlation between the testing errors and the uncertainty computed this way.

Data augmentation

Zhou et al. [121] measure the informativeness of an image by generating new patches like it is often done with data augmentation (see Chapter 2) during training. The

idea is to look if the predictions for all the patches are consistent or not. To handle too hard patches (or patches that are completely unusable), they only take into account a subset of all the patches generated to measure the confidence (majority selection).

Moreover, many papers retrain their models from scratch after each acquisition of labeled data. The authors implement a framework in which a pretrained model is actively and incrementally trained for biomedical images. Their experiments show that their continuously fine-tuned model converges faster than a repeatedly fine-tuned model (from scratch).

The same authors propose an improved approach [120] adapted for the multi-class case. They combine both uncertainty measure based on entropy and at the same time diversity measure of a data point by measuring the prediction consistency among the candidate patches.

A last interesting point is the injection of randomness in the selection step allowing the model to explore areas of the feature space where the model performs poorly.

Learning Loss

Another way to tackle the problem of measuring uncertainty is the Learning Loss method. The idea has been developed by Yoo and Kweon [114] and has also been used recently for COVID-19 diagnosis [110]. They add a loss prediction module (Θ_{loss}) to the model (Θ_{target}) as shown in Figure 3.3.

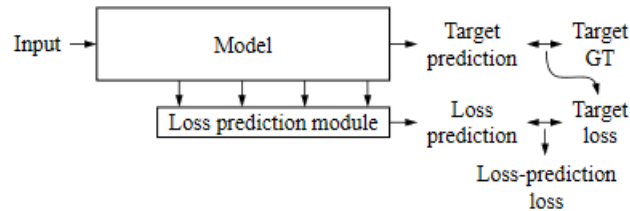


Figure 3.3: The loss prediction module. Source: Yoo and Kweon [114].

It has access to computed features from several layers of the target model and is trained to predict its prediction loss.

$$\hat{l} = \Theta_{loss}(h)$$

with h some features taken from the model.

An important trick described by the authors is the need to discard the overall scale of l since the error tends to decrease at each iteration of training. The expression of the loss-prediction loss (the loss of the added module) for each data pair denoted

by p , $x^p = (x_i, x_j)$ is given by

$$L_{loss}(\hat{l}^p, l^p) = \max(0, -\mathbb{1}(l_i, l_j)(\hat{l}_i - \hat{l}_j + \xi))$$

with $\mathbb{1}(l_i, l_j)$ equals 1 if $l_i > l_j$ and -1 otherwise and ξ a positive margin. As the initial purpose was to propose a task-agnostic method, the final loss function allows to jointly train the target model and the loss prediction model without any additional assumption

$$\frac{1}{|B|} \sum_{(x,y) \in B} L_{target}(\hat{y}, y) + \lambda \frac{2}{|B|} \sum_{(x^p, y^p) \in B} L_{loss}(\hat{l}^p, l^p)$$

with B the current batch, $\hat{y} = \theta_{target}(x)$, $\hat{l}^p = \theta_{loss}(h^p)$ and $l^p = L_{target}(\hat{y}^p, y^p)$. Like other methods quoted above, [110], based on the same architecture, develops a diversity-based sampling strategy by data augmentation and agreement measurement.

Margin based approach

Ducocffe and Precioso [21] propose a way to measure the distance with the decision boundary of the CNN and select the closest samples, i.e., the samples that could engender the biggest impact on the decision boundary and therefore carry the most information to train the model. The method is based on the DeepFool algorithm [61] to compute what is called an adversarial attack, a transformation of input so that it crosses the decision boundary. The advantage of this method is that it can get two labeled samples from one labeled input and it allows to make the model more robust to these attacks.

Cost-Effective Active Learning

Wang et al. [104] suggest to not consider only the most informative/representative samples but automatically select and annotate unlabeled samples, which is called pseudo-annotation. The unlabeled data are fed into the model and are divided into two categories.

The minority samples are the data with low confidence of the model concerning its prediction (the most informative data). This uncertainty can be measured with several tools, including the common ones like entropy or margin sampling. The selected data are annotated by the oracle and transferred to the labeled set. They are used to fine-tune the model and for further training.

The majority samples are the data with high prediction confidence. The method assigns a pseudo-label (without any human interaction). This allows using these pseudo-labels to fine-tune the model. After each step of training, the pseudo-labels

are erased and the loop continues until a budget or another constraint is reached. Each $x_i \in D_{pool}$ is pseudo-annotated if

$$j^* = \operatorname{argmax}_{0 \leq j \leq C} p(y_i = j | x_i)$$

$$y_i = \begin{cases} j^* & \text{if } en_i < \delta_t \\ \text{not used} & \text{otherwise.} \end{cases}$$

with en_i a given uncertainty measurement for sample i (such as entropy), δ_t an adaptive threshold (at iteration t).

The main contribution of this paper is the use of pseudo-labels for fine-tuning the model at each selection step, allowing the network to learn more discriminative features and therefore be trained and converge faster. Therefore, the uncertainty measurements at each step are more precise thanks to better features. On the other hand, there is a risk of using wrongly annotated samples; the authors address this problem by adapting the confidence threshold, which is decreased at each iteration to avoid incorrect auto-annotation.

This method enters in the semi-supervised category (a category of machine learning algorithm that uses labeled data as well as unlabeled data). A recent example is the meta pseudo labels network [68], currently one of the most efficient classification networks. This breakthrough shows the possible gain of combining this method with any other methods presented in this section, as a better fine-tuned model can compute more interesting intermediate features and take better decisions.

3.3.2 Diversity sampling

These methods focus on the representativeness of the data and generally try to take as different data points as possible to preserve diversity. In contrast to the previous methods, the risk is to select points that the model understands and for which it is confident, thus carrying less information for the training.

Core-set approach

Sener & Savarese [84] introduce the core-set approach, based on the fact that Deep Learning models need to be optimized by batch. It needs a method that works well for a relatively important budget at each iteration. The Active Learning process at iteration t can be defined as

$$\min_{s \subseteq D_{pool}: |s| \leq b} \mathbb{E}_{x, y \sim p_F} \left[l(x, y; A_{D_{train}^{t-1} \cup s}) \right]$$

with $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ and A_l a learning algorithm which outputs a set of parameters w given the labeled set l .

The authors show that, under certain assumptions met by CNN models, this optimization problem can be reduced to the minimization of the so-called core-set loss, which is the difference between the average empirical loss over the set of points with labels and the average empirical loss over the entire dataset (including unlabeled points). It is defined as (at iteration t)

$$\min_{s \subseteq D_{pool}: |s| \leq b} \left| \frac{1}{|D|} \sum_{i=1}^{|D|} l(x_i, y_i; A_{D_{train}^{t-1} \cup s}) - \frac{1}{|D_{train}^{t-1}| + |s|} \sum_{j \in D_{train}^{t-1} \cup s} l(x_j, y_j; A_{D_{train}^{t-1} \cup s}) \right|$$

For which they prove the following theorem that gives an upper bound to this loss

Theorem 1 *For a set of selected points s , if s is the δ_s cover of the dataset like shown in Figure 3.4 we have,*

$$\left| \frac{1}{|D|} \sum_{i=1}^{|D|} l(x_i, y_i; A_{D_{train} \cup s}) - \frac{1}{|D_{train}| + |s|} \sum_{j \in D_{train} \cup s} l(x_j, y_j; A_{D_{train} \cup s}) \right| \leq \mathcal{O}(\delta_s) + \mathcal{O}(\sqrt{\frac{1}{|D|}})$$

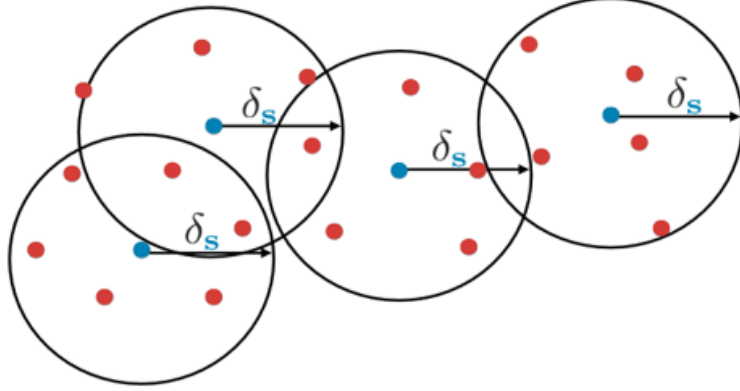


Figure 3.4: Visualization of Theorem 1. Source: Sener and Savarese [84].

This theorem can be applied to CNN and states that the core-set loss is bounded by the covering radius and a decreasing function of the number of samples in the dataset. The problem becomes therefore

$$\min_{s \subseteq D_{pool}: |s| \leq b} \delta_{D_{train} \cup s}$$

Furthermore, it is equivalent to a minimax facility location or the k-Center problem; in other words, the active sampling algorithm has to select a batch of data points

covering the unlabeled set as closely as possible. This is done by formulating an integer program robust to outliers. Finally, the authors propose to extract the features computed by the model, based on empirical evidence that similar cases engender similar features [17].

Discriminative/Adversarial Active Learning

Gissin & Shalev-Shwartz [32] propose a straightforward and more scalable method. During the selection of data points, ideally, the distribution of the training set matches the actual data distribution and therefore takes into account more dense areas, which is not the case for the core-set approach. Because of high dimensionality, the problem can quickly become too hard to solve or even unsolvable. The authors come with another question; they ask rather if the examples from the unlabeled set are indistinguishable from those in the labeled set. In this framework, the Active Learning task is reduced to a binary classification between the labeled and unlabeled classes. A classifier that receives a representation of the data as input is trained to discriminate the data points.

Sinha et al. [93] come with a similar idea in which variational autoencoders (a deep learning model able to learn rich latent spaces with good properties) computes a latent space, and a discriminator (adversarial network) has to classify the data in the labeled or unlabeled set with a certain probability. The bigger the probability of the data point to belong to the unlabeled set, the more this case is not well understood and carries information for the model.

Diverse mini-batch

With the same idea of focusing on diversity, Zhdanov [119] proposes a scalable geometric approach by using the well-known clustering algorithm K-means and taking into account both informativeness and diversity. The author demonstrates how to reformulate the Facility Location problem into a clustering problem and how to incorporate informativeness measures in the objective functions by adding weights computed by the margin sampling acquisition function.

(O-)MedAL

Smailagic et al. [94] bring a pretty similar idea to the field of medical image analysis with MedAL (Accurate and Robust Deep Active Learning). The basic idea is to select the most distant data points. The method extracts rich data descriptors of unlabeled data from the CNN model and compares the average distance with the

training set (the already labeled data).

$$x^* = \operatorname{argmax}_{x \in C_M} \frac{1}{N} \sum_{i=1}^N d(f(x_i), f(x))$$

with $N = |D_{train}|$, C_M the top M data points with the highest entropy, f the descriptors, and d a distance metric (euclidian distance in this case).

An extension is presented in O-MedAL [95] with some improvements. Firstly, to find x^* , the average distance can be reduced to the distance with the centroid of the labeled set, reducing the complexity from quadratic to linear (euclidian distance). Secondly, the authors introduce the online learning technique to manage the arrival of new annotated data and not retrain the model from scratch, enabling significant computational savings. O-MedAL retrains on the newly labeled data and a random subset of the previously labeled data.

Multi-criteria

Yuan et al. [115] combine four sampling criteria divided into two groups.

On the one hand, density and similarity measures reduce information redundancy (using the labeled samples), on the other hand, uncertainty and label-based measures speed up the convergence and address the class imbalance problem. The last measure tries to select two types of samples: the samples belonging to a class that exhibits performance improvement in the last iterations and the samples from classes associated with low performance.

$$W_t^m = \begin{cases} \max\left(0, \frac{AR_t^m - AR_{t-1}^m}{Z_1}\right) & \text{if } AR_t^m < k \\ \frac{1/AR_t^m}{Z_2} & \text{if } AR_t^m \geq k \end{cases}$$

with Z_1 , Z_2 normalization factors, k a threshold that balances the 2 types of samples, AR_t^m the accuracy of the CNN model for the m th class at the training step t .

Konyushkova et al. [46] combine prediction uncertainty and geometric uncertainty in a 3D segmentation framework. The method builds a graph linking the voxels (pixel in 3 dimensions) and measures the prediction certainty by applying the Random Walk algorithm. The main idea is to measure the prediction consistency for one voxel with respect to its neighbors.

3.4 Summary of Chapter 3

Active Learning is at the crossroad of several machine learning techniques: supervised learning, non-deterministic networks (e.g., Bayesian approach), ensemble techniques, semi-supervised learning, etc. Each of these is the core of the methods presented above.

This wide variety reflects a great interest in both uncertainty and diversity measures. While they seem quite different, these two types of measures have their advantages and drawbacks. Uncertainty selects samples that could have a great impact during the training since the model has difficulties to classify them at the risk of selecting similar samples. Diversity, on the contrary, chooses representative samples to avoid selecting similar samples at the risk of choosing samples already understood by the model. It is clear that these two ways of selecting data points for annotation can be complementary, as discussed in the next chapter.

Chapter 4

Results

4.1 Experiment 1: Active Learning strategies comparison

In this experiment, some of the Active Learning methods presented in Section 3.3 are compared, using the MNIST dataset¹ which is already divided into training and test set. The MNIST dataset is composed of grayscale images representing numbers from 0 to 9 (see Figure B.1 in the Annex). The training set defines the unlabeled set (the labels are hidden from the model), and the test set is used to assess the performance of the models at each Active Learning iteration, i.e., each time a new subset of the unlabeled set is annotated.

In the first instance, 100 samples from the unlabeled set are randomly chosen and annotated. The model is trained for the first time. Then, at each new AL iteration, 100 additional samples are selected according to the studied criterion until 1000 samples are annotated. For each new labeled set, five ResNet18 models (see Chapter 2) are trained from scratch (without pretrained weights) with a random search to find the best-performing hyperparameters, and an early stopping based on the accuracy on the validation set. Table 4.1 presents the hyperparameters and their search range. The model achieving the best accuracy on the validation set (composed of 1000 samples randomly selected and removed from the initial unlabeled set) is selected and assessed on the test set. The experiment is repeated five times for each sampling strategy to reduce random variability in results. For each method, the mean accuracy curve with standard deviation across the five trials (the wider and transparent curve) is shown.

¹<http://yann.lecun.com/exdb/mnist/>, Accessed 2021-17-5

Hyperparameter	Search range or value
number of epochs	50
learning rate	between 10^{-1} and 10^{-2}
dropout rate (fully connected layer)	between 0.3 and 0.7
weight decay	$5 * 10^{-4}$ (fixed)
learning rate decay	0.1 at epoch 20 and 35
optimizer	Adam
batch size	64
budget	100

Table 4.1: Hyperparameters of the ResNet18 model for experiment 1.

Each method is compared with the random sampling as a baseline, see Algorithm 2 (Appendix A). At each AL iteration, a subset of the unlabeled set of size budget is randomly chosen and annotated. Figure 4.1 shows that random sampling preserves the diversity of the labels. This will serve as a comparison to other methods below. For all other methods, the results of two trials out of five are presented. The third trial’s results are available in Appendix B.

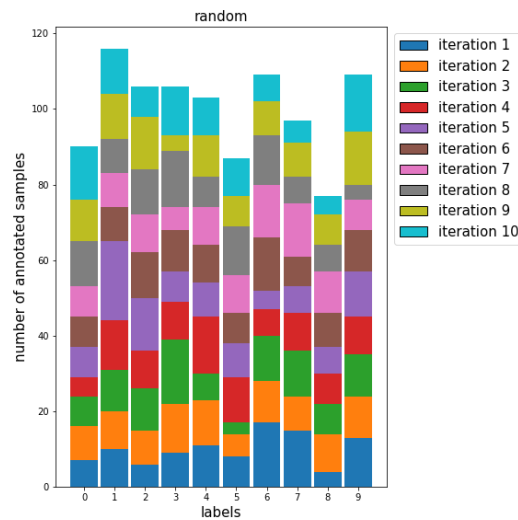


Figure 4.1: Cumulative chosen labels for the random strategy, trial 1 out of 5.

4.1.1 Entropy uncertainty sampling and Monte Carlo dropout

For these two methods, entropy (on the prediction of the model as presented in Section 3.2.2) is used as an uncertainty measure. This measure will be referred to

as MaxEntropy. The higher the entropy, the more the model associates equivalent probabilities to each class, measuring the difficulty of the model to predict the right label.

The deterministic approach is compared to Monte Carlo dropout (see also Section 3.2.2), given that the latter tries to better approximate the prediction probabilities through multiple forward passes with different canceled neurons thanks to the dropout layer. The procedures are presented in Algorithm 3 and 4 respectively, in Appendix A.

Experiment settings

The same random search is performed for the deterministic MaxEntropy uncertainty measure as presented at the beginning of the experiment. Two experiments are carried out for Monte Carlo dropout, with a dropout rate fixed at 0.3 and at 0.5 the second time. Only three models are trained at each iteration instead of five during the random search since the number of optimal hyperparameters to find at each iteration is reduced (the dropout rate is fixed). The number of forward passes (`nb_MC_forward`) is set to 100 as proposed by the paper's authors [28].

Results

Figure 4.2 shows that despite previous results highlighting the problem of overconfidence of neural networks, MaxEntropy uncertainty sampling performs as well as Monte Carlo dropout. This result is consistent with [32] for the MNIST dataset. Moreover, the two strategies constantly outperform the random approach, indicating that uncertain samples are more valuable than randomly chosen ones. The dropout rate for this case seems to have little impact on the quality of the selected samples. In the original paper, Monte Carlo dropout performs better than deterministic MaxEntropy. The difference of performance could be explained by the use of a more complex network (ResNet18 versus a simple two convolutional layers network followed by two fully connected layers in the paper), enabling the network to extract richer features and perform better. Moreover, the original paper uses two dropout layers, resulting in the prediction of more diverse networks at each forward pass.

Finally, as expected, these two strategies face mode collapse as shown on Figure 4.3 and 4.4. The mode collapse problem in Active Learning happens when the strategy selects too many similar samples or too many samples belonging to the same class. However, this issue tends to decrease with the number of iterations as the network seems to become less confident about samples with labels that have been less selected previously.

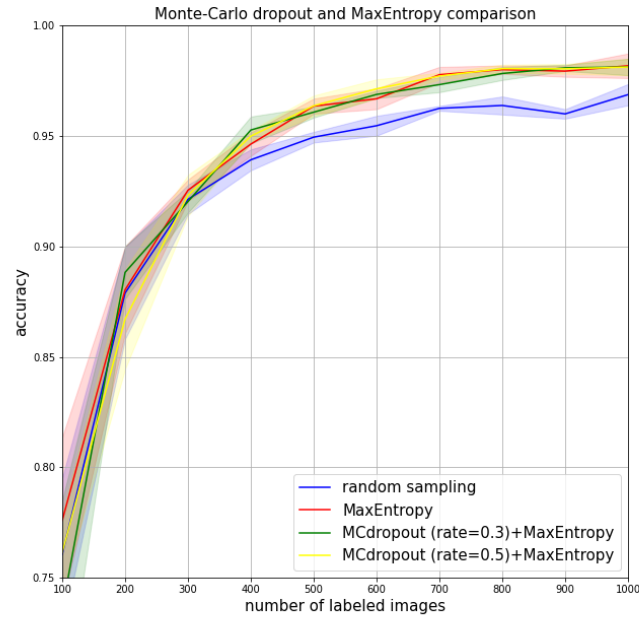


Figure 4.2: Monte Carlo dropout with MaxEntropy versus deterministic MaxEntropy.

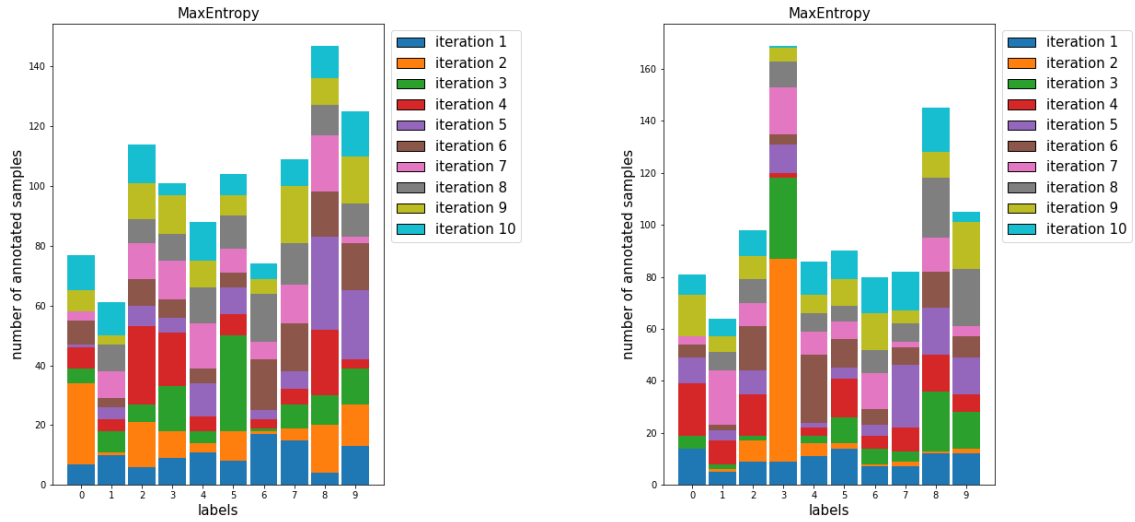


Figure 4.3: Cumulative chosen labels for entropy uncertainty sampling, trial 1 and 2 out of 5.

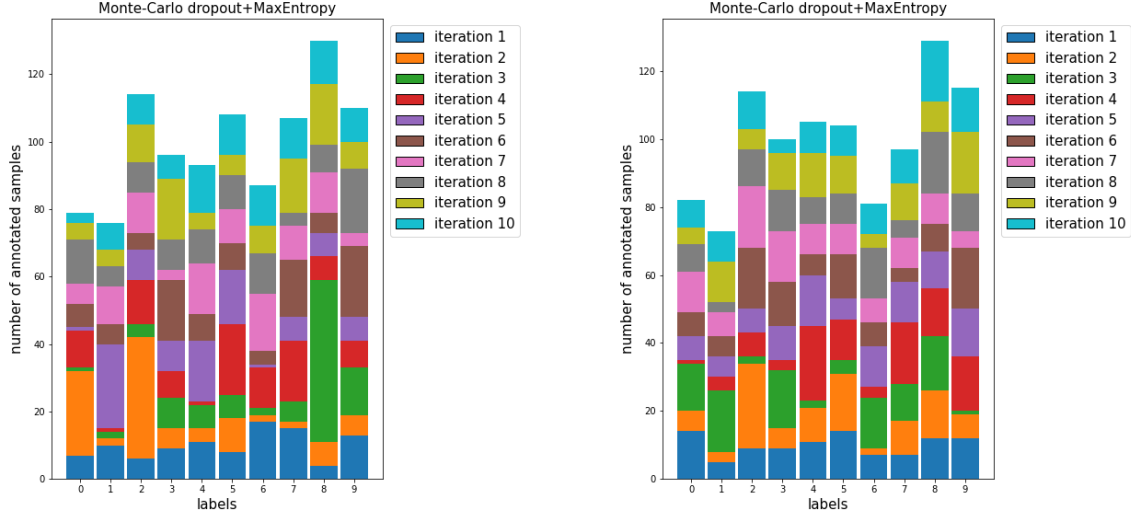


Figure 4.4: Cumulative chosen labels for Monte-Carlo dropout, trial 1 and 2 out of 5.

4.1.2 Ensemble technique

Instead of considering multiple predictions from the same model, the ensemble method considers the opinion of several models trained separately. For a matter of computing time, three models are trained at each AL iteration. However, [4] shows already an improvement for this number of models compared to 100 forward passes of the Monte-Carlo dropout as done above. The procedure of this method is presented in Algorithm 5 in Appendix A. The predictions are averaged over the three models, and then MaxEntropy or variation ratio (see Section 3.2.4) is used to measure the uncertainty of the ensemble predictions.

Experiment settings

Each of the three models is trained according to the method presented at the beginning of the experiment.

Results

Figure 4.5 shows that the ensemble method outperforms by a large margin the random strategy. Moreover, this performance can not only be explained by the better decision at test time due to the several models since the same ensemble procedure (same number of models) with a random strategy is less efficient. Ensembles allow better calibration of the output [50] and therefore a better uncertainty measurement than a single model.

Finally, the ensemble method achieves better accuracy than Monte Carlo dropout. This result is consistent with [4]. The authors suggest ensemble allows a better diversity than the different networks generated by Monte Carlo dropout. This lack of diversity can be explained by the same weights initialization for each model and the reduced capacity of each generated network (with a reduced number of neurons). They also trained their ensemble with the samples selected by the Monte Carlo dropout method and conclude that the higher performance can not only be explained by the fact that evaluating with an ensemble is more accurate than a single model.

We analyze the ensemble with variation ratio as it is the most efficient strategy of this section (results for ensemble with MaxEntropy are available in Appendix B). Figure 4.6 shows that the selected labels are diversified at the end of the sampling but some labels are much more often selected in the first Active Learning iterations. We can examine more precisely the evolution of the accuracy of each class and observe if the method is able to select the samples of the classes for which it is the least efficient. In Figure 4.7, we can observe that label 2 is the most difficult to predict for the model, and also that it is the most chosen label at iteration 2. The same observation can be made for label 8 at iteration 3. Based on this Figure, the method can select labels for which the ensemble has difficulties to predict well.

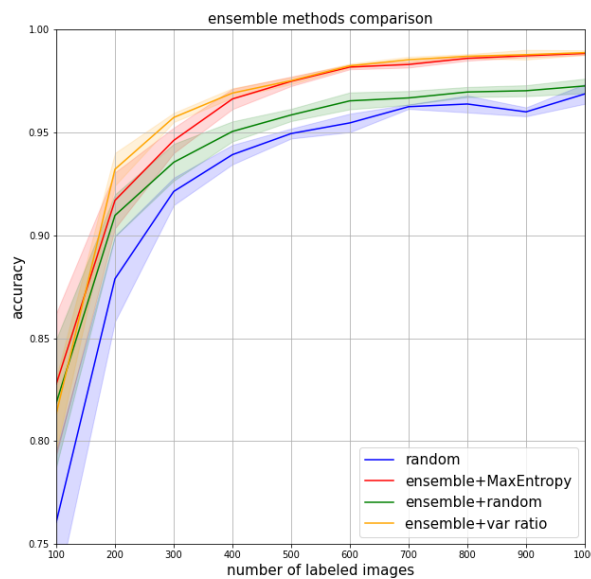


Figure 4.5: Ensemble Active Learning strategies comparison.

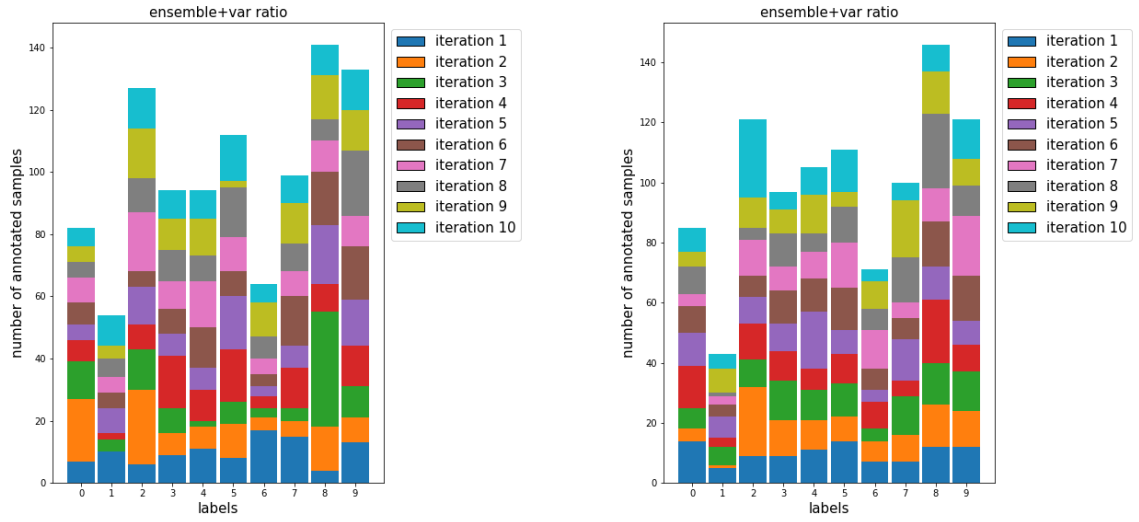


Figure 4.6: Cumulative chosen labels for the ensemble method with variation ratio, trial 1 and 2 out of 5.

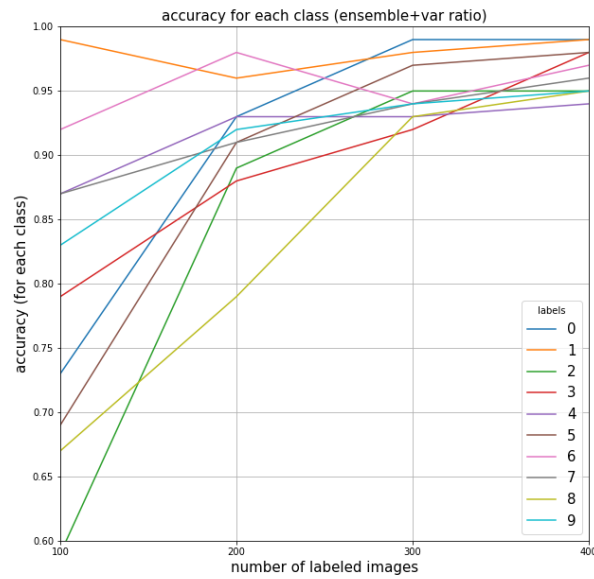


Figure 4.7: Accuracy for each class of the ensemble with variation ratio measure (trial 1).

4.1.3 Core-set and discriminative Active Learning

These two methods can be seen as quite similar in their approach to tackle the problem of choosing which samples to annotate. On the one hand, core-set selects samples in order to cover unlabeled samples by a minimum radius cover in a given feature space. On the other hand, discriminative Active Learning trains models that select unlabeled samples to match the labeled samples' distribution.

In Appendix A, the procedures are presented in Algorithm 6 for the greedy version of the core-set approach and 7 for the discriminative AL. Both approaches extract the features computed by the network in the last layer and promote the diversity of the selected samples.

Experiment settings

The same random search as presented at the beginning of the experiment is used. For the core-set approach, the euclidian distance is employed to measure distances between the samples as proposed in the original paper [84].

Discriminative AL needs to train models composed of fully connected layers to select the samples (binary classification) at each iteration; the hyperparameters used by the authors [32] were employed, see Table 4.2.

Hyperparameter	Value
input size	512
number of layers	4
number of neurons	256
activation function	ReLU
optimizer	Adam
learning rate	$2 * 10^{-4}$
number of epochs	500

Table 4.2: Hyperparameters of the discriminative models.

Unlike the method presented by the authors, only five discriminative models were trained at each AL iteration (instead of twenty). This choice was made to keep the experiment reasonable in terms of duration and stay fair compared to other strategies. Despite this choice, each iteration remains computationally intensive compared to all other methods evaluated in this experiment. Computation time of each method is summarized in Section 4.1.5.

Results

The performances are presented in Figure 4.8. Surprisingly, core-set outperforms discriminative AL by a large margin. This result is inconsistent with the discrimi-

native AL paper, where these two methods have similar performances. This could be explained by the difficulty in training the discriminative models and the reduced number of models trained in our experiment compared to the original method proposed by the authors (five instead of twenty discriminative models).

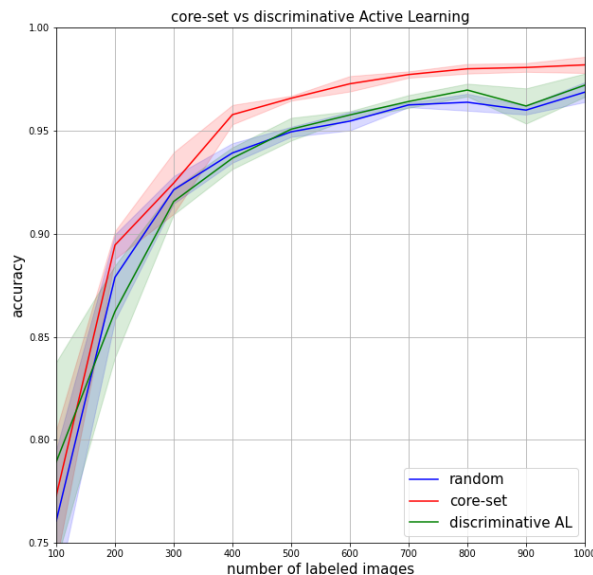


Figure 4.8: Core-set approach versus discriminative AL.

Another explanation can be deduced from Figures 4.9 and 4.10. The core-set approach chooses much more diverse labels at each iteration. It allows a better learning of the model and thus to calculate more discriminating features at each step. The discriminative AL strategy, on the contrary, faces a significant mode collapse problem at each iteration. Only a small subset of labels are selected most of the time, preventing the model from converging well.

In the core-set approach, samples with label 1 are significantly less chosen. It could be due to the fact that images containing the number 1 are more similar since it is a vertical bar, which is much less ambiguous than other numbers with a more complex representation (see Figure B.1 in Appendix B). Therefore, samples with label 1 could be quite close to each other, and selecting only a few of them could be enough to represent all the other similar samples. Table 4.3 shows that despite selecting lesser label 1, the model achieves great performance for this class.

It also gives an additional argument on the fact that label 1 is easier to classify and that more generally, similar samples of the same class have a similar feature representation.

	class 0	class 1	class 2	class 3	class 4	class 5	class 6	class 7	class 8	class 9
Accuracy (trial 1)	0.993	0.994	0.983	0.984	0.976	0.983	0.981	0.978	0.980	0.973
Accuracy (trial 2)	0.999	0.993	0.986	0.991	0.977	0.979	0.985	0.984	0.976	0.972

Table 4.3: Accuracy of each class at the end of the core-set sampling (trial 1 and 2).

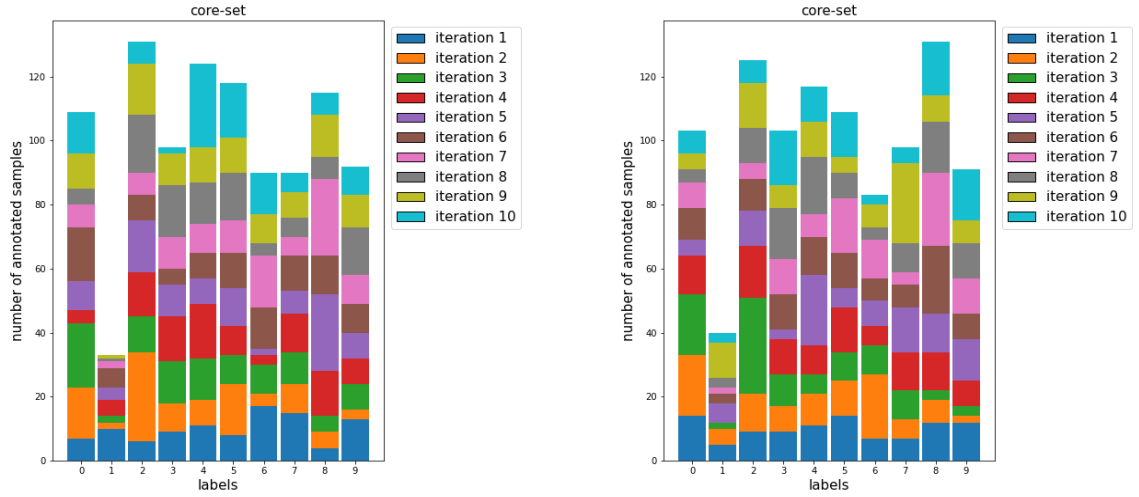


Figure 4.9: Cumulative chosen labels for core-set, experiment 1 and 2 out of 5.

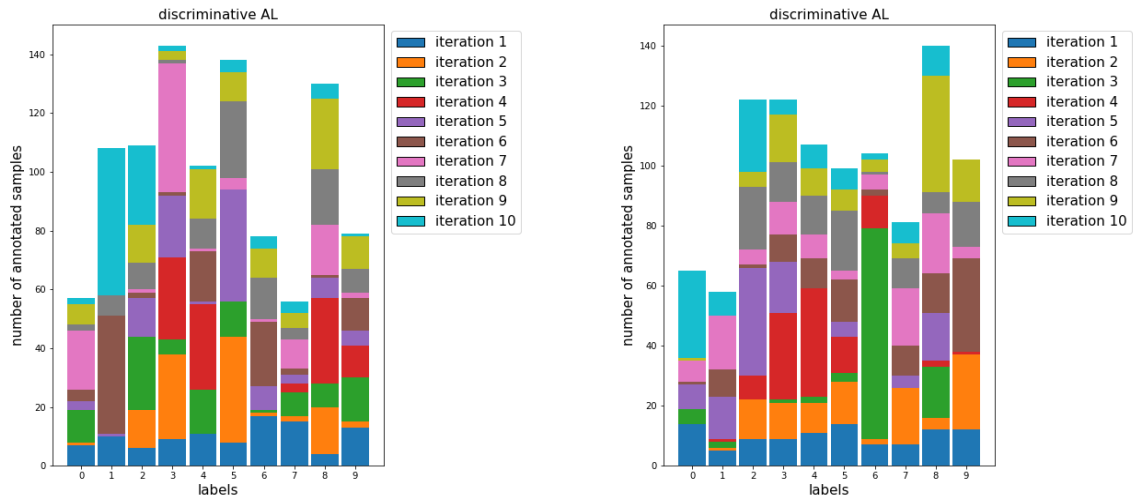


Figure 4.10: Cumulative chosen labels for discriminative AL, trial 1 and 2 out of 5.

4.1.4 Learning Loss Active Learning

Compared to most methods presented above, the drawback of this method is the difficulty of fine-tuning the module attached to the main model. The original paper adds an epoch loss hyperparameter, i.e., the number of epochs after which the module’s error is not propagated anymore to the main model, adding a new hyperparameter to tune. The procedure is presented in Algorithm 8 in Appendix A.

Experiment settings

Since the training procedure is different from the original one presented at the beginning of the section, this method needs an other hyperparameter search and a longer training due to the need to train the loss prediction module and the main model jointly before the number of epochs reaches epoch loss. The new hyperparameters are presented in Table 4.4, the other ones are kept identical.

Hyperparameter	Value
epochs	60
epoch loss	25
learning rate decay	0.1 at epoch 33 and 45

Table 4.4: Hyperparameters for the learning loss method.

Results

Figure 4.11 shows that the learning loss method performs worse than the random strategy in the first iterations. This could be explained by the difficulty to find the optimal hyperparameters during the random search. The high standard deviation of the accuracy across the five trials supports this observation. However, the method outperforms the random strategy during the last four iterations. A more stable training procedure thanks to more annotated samples and more discriminative features extracted by the module are certainly the cause of this improvement. The main model achieves better performance and can therefore compute richer features from more diverse samples. Figure 4.12 shows an important mode collapse. Only a few different labels are selected at each iteration, which partly explains the poor performance during the first iterations. Our results are difficult to compare with those presented in the original paper [114] since they assess their method on the CIFAR-10 dataset (a dataset of natural images with 10 classes) with a budget of 1000. Their method outperforms the deterministic MaxEntropy and the core-set approach by a small margin. This may lead us to believe that some methods work

better on certain datasets or with other constraints (e.g., a larger budget) than others. This fact may be an important limitation to the application of Active Learning in practice if a method with stable performance on multiple tasks is desired. Further studies with more diverse datasets, networks, and budgets should be conducted to confirm this hypothesis.

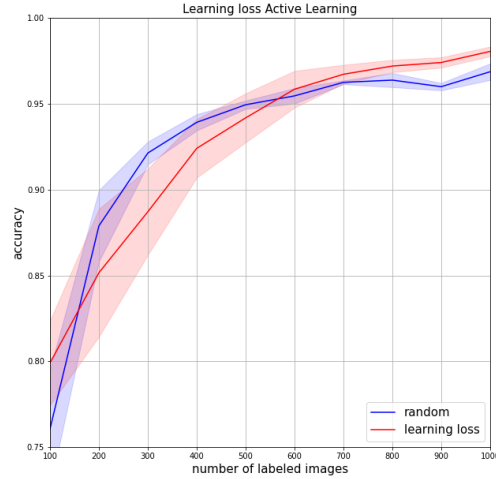


Figure 4.11: Learning loss AL versus the random strategy.

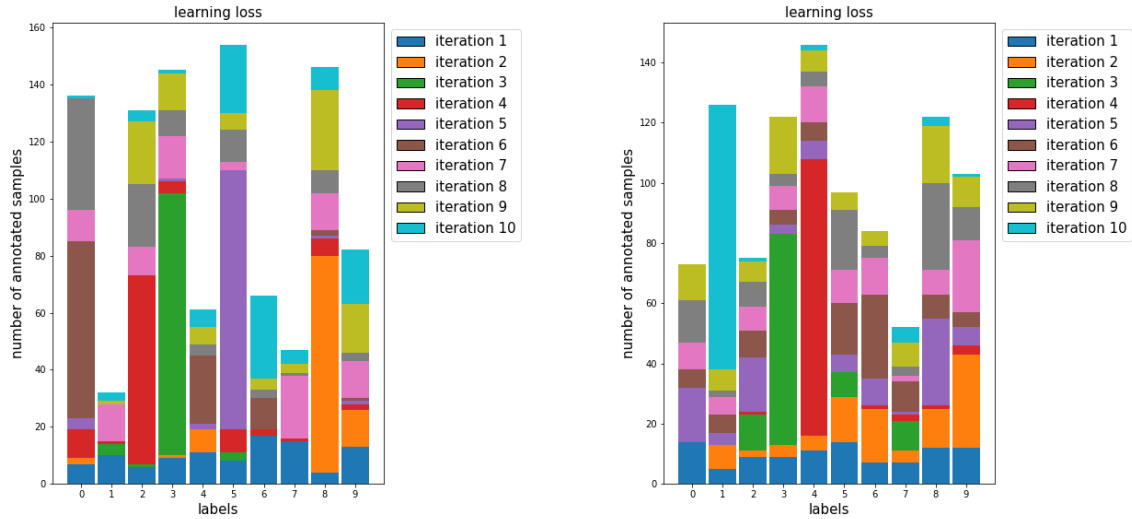


Figure 4.12: Cumulative chosen labels for learning loss AL, experiment 1 and 2 out of 5.

4.1.5 Summary of experiment 1

The majority of the methods presented in this experiment outperform the random strategy by a large margin. In addition, even if these methods have completely different approaches, they are able to select interesting samples allowing better training and better performance. This leads us to believe that many different criteria are useful for selecting these samples, and a new method that optimally weights these criteria might be interesting to investigate. Figure 4.13 and Table 4.5 summarizes our results, as well as the computation time per trial of each method. The experiment was conducted on a Nvidia GeForce GTX 1080 graphics card.

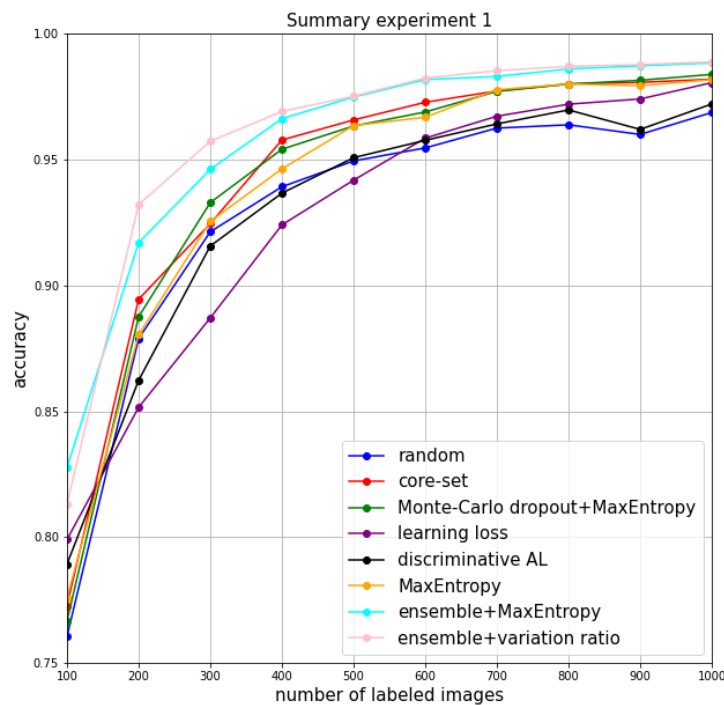


Figure 4.13: Summary of experiment 1.

Method	Mean accuracy $\pm$ standard deviation	Computation time (for one trial)
random	0.969 ± 0.0048	≈ 35 minutes
MaxEntropy	0.982 ± 0.0056	≈ 40 minutes
ensemble+variation ratio	0.989 ± 0.0011	≈ 2 hours
ensemble+MaxEntropy	0.988 ± 0.0007	≈ 2 hours
core-set	0.982 ± 0.0039	≈ 40 minutes
Monte Carlo dropout+MaxEntropy	0.984 ± 0.0015	≈ 45 minutes
learning loss	0.981 ± 0.0028	≈ 55 minutes
discriminative AL	0.972 ± 0.0057	≈ 2 hours 30 minutes

Table 4.5: Summary of experiment 1 after 10 AL iterations.

4.2 Experiment 2: Active Learning for Mammography

This experiment aims to evaluate some Active Learning methods applied to mammography classification. As the whole-image classifier (see Section 2.4.1) seems hard to train, we do not expect solid results and therefore we prefer the patch classifier (see Section 2.4.2). In addition, it provides a direct link to our Faster R-CNN network, which must also classify areas of a mammogram as normal, benign or malignant.

At each Active Learning iteration, five models are trained with a random search and early stopping. The experiment is repeated five times for each method. The final experiment settings are summarized in Table 4.6.

Hyperparameter	Value or search range
seed size	100
budget	100
total budget	1000
model	patch classifier
optimizer	Adam
epochs	65
learning rate (initialization)	between 10^{-3} and 10^{-4}
learning rate decay	0.1 (at an epoch randomly chosen between 5 and 25)
dropout rate	0
weight decay	between $5 * 10^{-3}$ and $5 * 10^{-5}$

Table 4.6: Hyperparameters search range of experiment 2.

We set a relatively small total budget so that methods can select more various training sets at the end of each trial.

4.2.1 Uncertainty and diversity

Firstly, we compare the random, the MaxEntropy, the core-set and the ensemble with variation ratio approaches. Results are shown in Figure 4.14.

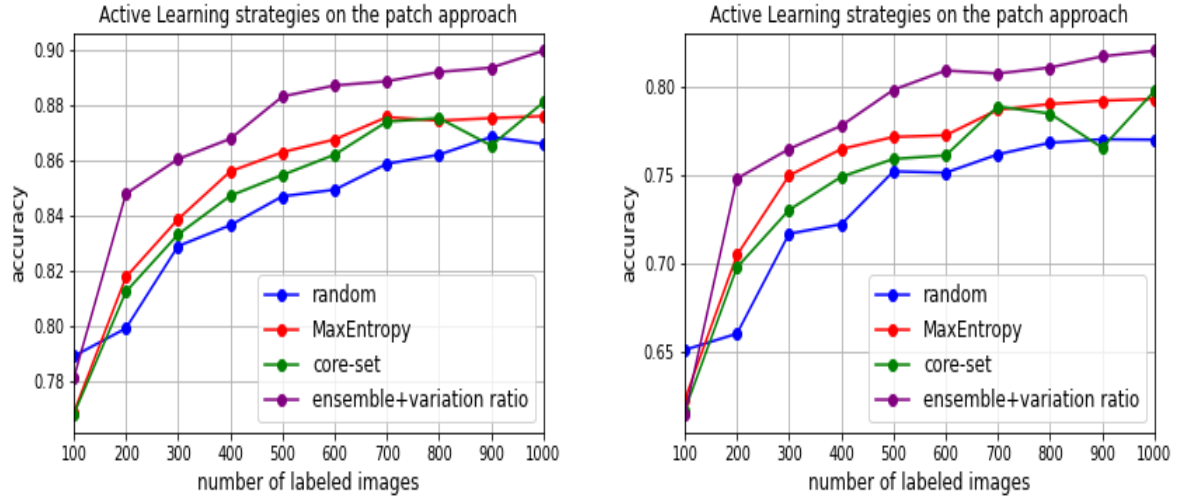


Figure 4.14: Active Learning strategies on the patch classifier.

As for the MNIST dataset, the ensemble with variation ratio performs better than the other strategies. Moreover, we observe that the core-set approach is not performing as well as in experiment 1 compared to the other methods. This could be explained by the fact that the dataset is relatively small and, therefore, there are fewer redundant data points. Indeed, as explained in Chapter 1, masses can take various shapes, contours, and sizes so that patches presenting abnormalities can be very different from each other. Uncertainty measures seem to better select interesting samples in this configuration.

4.2.2 Combining uncertainty and diversity

We combine uncertainty and diversity in a mixed method selecting firstly twice as many data points than the budget allows with the MaxEntropy measure. Afterwards, we select the most representative samples with the core-set approach on this subset. The idea is to select the most informative samples with the uncertainty measurement in a first step, and then avoid redundancy in the data points by taking the most representative ones. Results are shown in Figure 4.15. The combined method outperforms most of the time slightly the uncertainty sampling strategy. The improvement is small but indicates that a more thoughtful combined approach might be worth investigating.

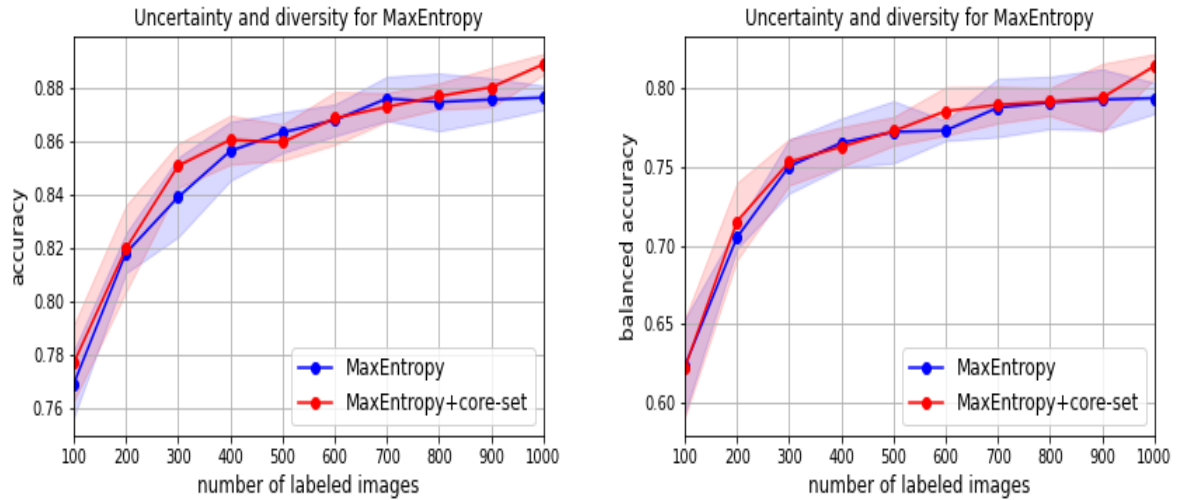


Figure 4.15: Active Learning strategy combining uncertainty (MaxEntropy) and diversity (core-set), mean accuracy with standard deviation.

This combined approach is tested with the ensemble and variation ratio measure. The features are extracted from the three models of the ensemble (as the method which extract features from only one model of the ensemble performs worse). Results in Figure 4.16 shows that the combined method is not able to achieve similar performance as the simple ensemble method.

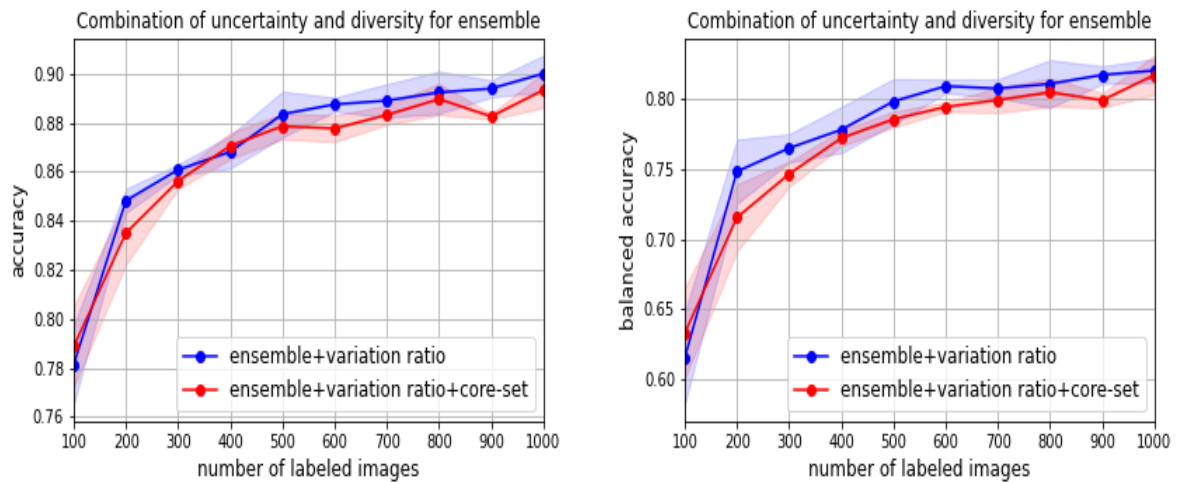


Figure 4.16: Active Learning strategy combining ensemble+variation ratio and diversity (core-set).

Despite this rather disappointing result, other diversity measures that take into account the feature representation of the ensemble may perform better and should be explored in future work, see discussion in Chapter 6.

Finally, we assess our MaxEntropy combined method for a more important budget. We set the budget at 300 data points per AL iteration. Results in Figure 4.17 show a small improvement in accuracy for the third iteration but lower accuracy for the second iteration. This improvement is meager, and because of the highly variable results when examining standard deviation, we cannot draw a definitive conclusion.

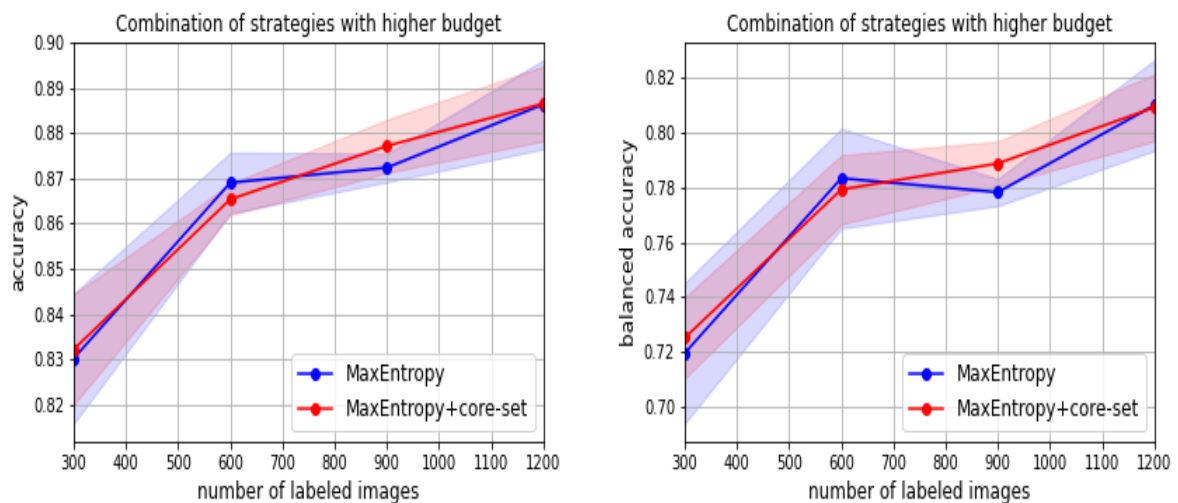


Figure 4.17: MaxEntropy and MaxEntropy+core-set for a budget of 300 data points.

4.2.3 Summary of experiment 2

We conclude that both uncertainty and diversity measure are useful depending on how many questions need to be asked at once and depending on the dataset properties. If the task needs to annotate a lot of data at once or if the unlabeled dataset presents very similar data points, there can be a gain in introducing diversity measures in the Active Learning process. On the contrary, if the task allows to ask annotations to experts several times, uncertainty measure seems to perform better. In our case, the dataset seems too small for the diversity measure to perform as well as the uncertainty measure. Indeed, if the data points are very diverse, uncertainty appears to be a more suitable choice as a selection metric. The combination of the two measures may lead to a significant gain in case of high redundancy in the dataset, although this matter needs to be further explored in future work. Even if

our patch approach only deals with patches of fixed size, we can establish a link with our Faster R-CNN model which also deals with the classification of regions of interest. A last experiment applying uncertainty measure to our Faster R-CNN is presented in Appendix C.

Chapter 5

Future work

This section presents a new and more complete way of dealing with the Active Learning matter. By combining Reinforcement Learning and coalition theory, we propose a more flexible strategy that considers how to apply Active Learning in practice.

We do not want our model's annotation query strategy to be based on a predefined heuristic (as is often the case in Active Learning). Instead, we would like a model that is able to learn its own query strategy and to adapt to several medical tasks. The main idea to refine our Active Learning approach is to introduce a second well-studied paradigm: Reinforcement Learning (RL). RL is a machine learning area where agents need to take actions in a particular environment to maximize their reward. Therefore, the idea is to build a model that will improve itself through carefully asked annotations and enhance its sampling strategy continuously, making it much more flexible.

The principle of Reinforcement Learning is to let the agents learn the best strategy to reach their goal. The term reinforcement comes from the fact that, during training, the agents receive rewards when they make good decisions and sometimes penalties when they make the wrong ones. The agent must choose which action to execute given a particular observation of its environment and a specific state: a position at a certain time step within the environment. It also includes a value function providing a prediction of how good a state and/or an action is. We distinct the model-free RL, in which this function is known, and the model-based RL, for which only a model of the environment is available (e.g., the agent does not know how the environment is changing given its actions and states). The final goal remains to determine the policy providing the greatest cumulated reward. Reinforcement Learning has been shown to solve challenging problems for which the correlation between immediate actions and their returns is complex. François-Lavet et al. [25] present an overview of Reinforcement Learning and how it can be combined with the generalization ability of Deep Learning. One of the most famous applications

of this combination is DeepMind’s AlphaGo [90], the first computer program to defeat a professional Go player in 2015.

The quality of annotation is decisive for a DL model to ensure its predictive power. As experienced radiologists make fewer mistakes than novice ones, our system should assess the quality of the annotations it receives. We must also think about rewarding the radiologists according to their involvement and accuracy as well as motivating them to participate. For these reasons, we intend to introduce a mathematical domain called Game Theory, which studies the interactions between two or more decision-makers, in our work. We propose to start from the TCLearn architecture [57] that studies how to handle different data owners who jointly train a machine learning model (by trusting or not each other) and how to evaluate each member’s proposal of model improvement. This would be stimulating for all the users and also benefit younger radiologists thanks to cross-validation of their propositions on complex cases.

5.1 Deep Reinforcement Active Learning

Research on Active Learning and Reinforcement Learning is relatively sparse and needs to be further developed. To our knowledge, this combination has never been explored for mammography. Detecting a tumor needs to point out one or more region(s) of interest, which is a complex operation. Most models proposed to radiologists nowadays provide a diagnosis with a degree of certainty; they do not use the radiologist to initiate a dialogue and improve themselves. We believe this new approach could have a significant impact on how image-based diagnostic AI’s are designed.

Some authors try to combine Reinforcement Learning and Active Learning, sometimes called Deep Reinforcement Active Learning. They start from the fact that most Active Learning methods are based on heuristics, which will work more or less well depending on the task. Therefore, they propose to allow the model to create its own sampling strategy by Reinforcement Learning. Fang et al. [24] apply their Policy-based Active Learning to natural language processing tasks. In 2017, Woodward & Finn [109] submit a one-shot Active Learning framework for image classification. One-shot means that the model has only one opportunity to see each sample and to decide whether to predict it or query a label. The model receives a reward for a correct prediction, a penalty for a wrong prediction, and another penalty (given the cost of labeling) for a label query. Liu et al. [54] apply Reinforcement Learning to person re-identification (search the same people from multiple views), with an agent selecting instances from a pool and binary feedbacks

based on a similarity measure between the queried samples and the already labeled ones. The framework is shown in Figure 5.1 with a gallery pool containing the unlabeled data.

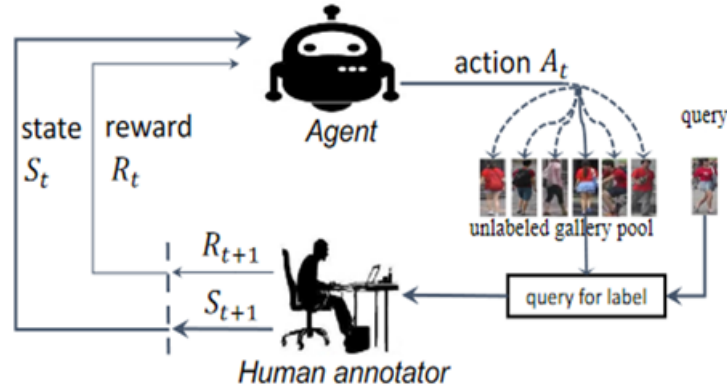


Figure 5.1: Example of Active Reinforcement Learning framework. Source: Liu et al. [54].

5.2 Coalition theory

In an actual deployment, several users will be asked to annotate mammograms to train the same shared models. One single organization gathering such a vast amount of data is complex in terms of legality, ethics, or storage. Moreover, several concerns can arise in such a setting: the risk of corruption (like unintentional errors), privacy leaks, but also the need to have an authority monitoring each training increment.

Secure Architecture Implementing Trusted Coalitions for Blockchain Distributed Learning (TCLearn) proposed by Lugan et al. [57] is a distributed learning framework aiming to preserve privacy but also data quality across each training sequence. The architecture is shown in Figure 5.2. More precisely, a CNN model is shared among the coalition members and optimized in an iterative sequence. Each member of the coalition updates it sequentially with new batches of their own local data. Each iteration of the shared model is validated by a process involving the coalition members and then stored in a blockchain. As a result, every single step can be retrieved and quality of training can be guaranteed. Federated Byzantine agreement is used to validate the new incoming data with validators chosen within the consortium.

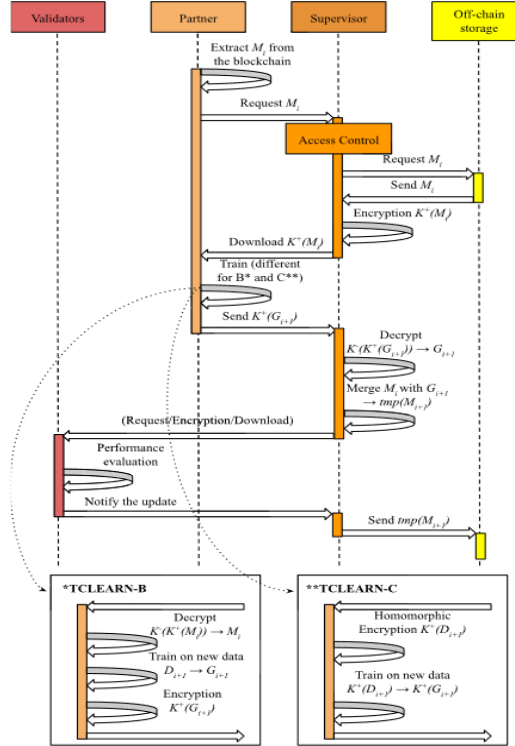


Figure 5.2: Federated Byzantine agreement with model privacy guaranteed by encryption. Source: Lugan et al. [57].

5.3 Summary of Chapter 5

We propose to combine Reinforcement Learning, Active Learning, and the distributed learning framework of TCLearn. Active Learning and Reinforcement Learning ensure a model that improves over time, while the TCLearn architecture preserves the model from corrupted data and guarantees data confidentiality, which is a critical concern in medical applications.

Chapter 6

Discussion and conclusion

6.1 Discussion on breast cancer screening

The real impact of early screening is difficult to measure precisely and can only be estimated through statistical analysis and assumptions (like the fact that breast cancer rate did not evolve too much since the introduction of widespread screening mammography). Gilbert Welch et al. [106] estimate that at least two-thirds of the reduction in breast cancer mortality must be attributed to better treatment (USA). After the introduction of widespread screening mammography the number of detected large tumors has decreased (from 145 to 115 cases per 100,000 women) but the number of small tumors has increased in a more significant way (from 82 to 244 cases per 100,000 women), suggesting that screening allows to detect tumors before they grow too large, but also detects cancer that would not need any treatment otherwise. As a result, it is estimated that 132 cases of breast cancer per 100,000 women are over-diagnosed. Moreover, as suggested by Figure 6.1, the number of metastatic breast cancer remains stable. Ghajar & Bissel [29] present evidence that tumor cells can leave the primary site very early, thus highlighting the inability of mammography to detect this type of cancer in time. Currently, the 5-year survival rate for women with metastatic breast cancer is estimated at 28% [13].

It is important to mention that screening is not only limited to mammography or more generally to imaging-based tests. Other measurable factors can influence a woman's lifetime risk of developing cancer. For example, BRCA1 and BRCA2 are two genes that are known to highly increase the risk of cancer when they present mutations. Even if genetic testing is not recommended for the general public at present time, it constitutes another tool for people who may be at higher risk of carrying a harmful BRCA1 or BRCA2 variant (such as people who have a family history of certain cancers) [63].

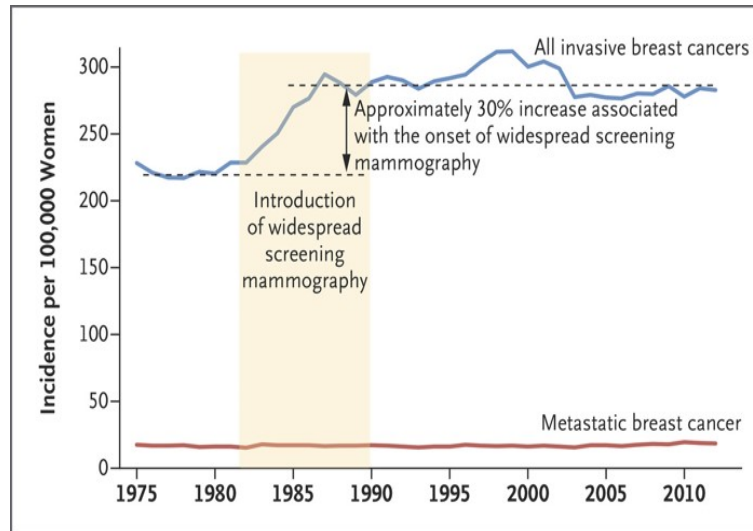


Figure 6.1: Incidence of invasive breast cancer and metastatic breast cancer among women 40 years and older from 1975 to 2021. Source: Gilbert Welch et al. [106].

6.2 Limitations, improvements of the results and future work

The results from Chapter 2 could gain in value with a more formal approach implying statistical proves of performance. K-fold cross validation with multiple repetitions could give a better nsight into the distribution of accuracy, AUC, mAP and FROC metrics. This has not been achieved due to the time limit and the limited available computation power. Moreover, the combination of mammography with other information as family history or age was not studied.

Active learning methods can be tedious to assess, as they require training a large number of models with different training set sizes. More solid results could be obtained again by k-fold cross-validation to better estimate the standard deviation of accuracy of each method. Indeed, a method that performs slightly worse on average with much more stable results might be preferred in practice over a method that performs better but with much more variable outcomes. Moreover, the computational overhead could be reduced by not retraining the model from scratch at each Active Learning iteration but by continuously fine-tuning the same model. This approach needs careful reuse of previous labeled data, as the model will be trained much more often with the first labeled samples than the last selected ones. A combination of random selection of previously labeled data with newly labeled data could be interesting to study. More intelligent techniques could select more samples from classes for which the model performs worse than for the others. The

experiments were carried out on two different datasets, but as suggested in Chapter 4, some methods could perform better on a given dataset depending on the nature of the images, the number of available images, or the similarities between the images.

Some methods like the ensemble one could be improved in terms of computation time. Some authors propose to use snapshot ensemble [40], horizontal ensemble [111], vertical ensemble [111] or stochastic gradient descent with warm restarts [55] to remove a part of the burden of training multiple models. Moreover, only one prediction combination was studied: the average of the predictions of the ensemble members. More complex combinations are possible like a weighted average ensemble, stacked generalization [108], or model weight averaging [43].

Certain approaches have not been studied, such as those using bootstrapping, data augmentation or semi-supervised learning. To give a better overview of all the possible tools to select the samples, it would be interesting to explore them.

The combined method presented in Chapter 4 needs improvement, in terms of comparison with other methods and by finding the good balance between uncertainty and diversity. As suggested at the end of Chapter 4, both measures could be useful depending on the task to perform and the nature of the dataset. However, this matter was not studied in this master thesis.

Finally, a complete active system based on our Faster R-CNN and Active Learning strategies has not been implemented. This would be helpful to get senologists' feedback and assess its practical feasibility. We believe that this approach combined with Deep Reinforcement Active Learning and coalition theory, as presented in Chapter 5, may worth exploring, and could lead to better interactions, decisions, performance, and ultimately better acceptance by senologists.

Acknowledgments

First of all, I would like to thank Benoit Macq for his supervision. I appreciated the fact that he challenged me to explore new ideas such as introducing Active Learning in this thesis. I also thank him for putting me in contact with senologists of the Clinique Saint-Pierre of Ottignies, Vincent Nicolas (CEO of Intuitim) and professors of the UMonS university.

I would like to thank Vincent Nicolas for his availability and his knowledge sharing about mammography screening and the current products assessing mammograms. The monthly meetings with Benoit Macq, Vincent Nicolas, Andre-Robert Grivegne, Saïd Mahmoudi, Sidi Mahmoudi, and Xavier Lessage were the occasion to think outside the box and push me to get regular results.

Finally, I want to thank Elliott Brion for his careful proofreadings and for helping me being more rigorous in my work through his constructive criticism.

Bibliography

- [1] A. c. society. "How common is breast cancer?". <https://www.cancer.org/cancer/breast-cancer/about/how-common-is-breast-cancer.html>. Accessed: 2021-04-4.
- [2] Richa Agarwal et al. "Automatic mass detection in mammograms using deep convolutional neural networks". In: *Journal of Medical Imaging* 6 (Feb. 2019), p. 1. DOI: 10.1117/1.JMI.6.3.031409.
- [3] Ayelet Akselrod-Ballin et al. "Deep Learning for Automatic Detection of Abnormal Findings in Breast Mammography". In: Sept. 2017, pp. 321–329. ISBN: 978-3-319-67557-2. DOI: 10.1007/978-3-319-67558-9_37.
- [4] William Beluch et al. "The Power of Ensembles for Active Learning in Image Classification". In: June 2018, pp. 9368–9377. DOI: 10.1109/CVPR.2018.00976.
- [5] "Bi-Rads.". <https://www.acr.org/Clinical-Resources/Reporting-and-Data-Systems/Bi-Rads>. Accessed: 2021-07-4.
- [6] Paz Boix-Montesinos et al. "The past, present, and future of breast cancer models for nanomedicine development". In: *Advanced Drug Delivery Reviews* 173 (2021), pp. 306–330. ISSN: 0169-409X. DOI: <https://doi.org/10.1016/j.addr.2021.03.018>. URL: <https://www.sciencedirect.com/science/article/pii/S0169409X2100096X>.
- [7] Freddie Bray et al. "Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries". In: *CA: A Cancer Journal for Clinicians* 68.6 (2018), pp. 394–424. DOI: <https://doi.org/10.3322/caac.21492>. eprint: <https://acsjournals.onlinelibrary.wiley.com/doi/pdf/10.3322/caac.21492>. URL: <https://acsjournals.onlinelibrary.wiley.com/doi/abs/10.3322/caac.21492>.
- [8] *Breastcancer.org* "Dense Breasts". https://www.breastcancer.org/risk/factors/dense_breasts. Accessed: 2021-29-4.

- [9] *Breastcancer.org "statistics"*. https://www.breastcancer.org/symptoms/understand_bc/statistics. Accessed: 2021-10-4.
- [10] *Britannica "Mammary gland"*. <https://www.britannica.com/science/mammary-gland>. Accessed: 2021-29-4.
- [11] Brent Burbridge. *Undergraduate Diagnostic Imaging Fundamentals*. Nov. 2017. ISBN: 978-0-88880-611-6.
- [12] *Cancer center "Breast cancer types"*. <https://www.cancercenter.com/cancer-types/breast-cancer/types>. Accessed: 2021-04-8.
- [13] *Cancer.net "Breast Cancer - Metastatic: Statistics"*. <https://www.cancer.net/cancer-types/breast-cancer-metastatic/statistics>. Accessed: 2021-19-8.
- [14] Nicolas Carion et al. "End-to-End Object Detection with Transformers". In: *CoRR* abs/2005.12872 (2020). arXiv: 2005.12872. URL: <https://arxiv.org/abs/2005.12872>.
- [15] Kashyap Chitta, Jose M. Alvarez, and Adam Lesnikowski. *Large-Scale Visual Active Learning with Deep Probabilistic Ensembles*. Nov. 2018.
- [16] Hiba Chougrad, Hamid Zouaki, and Omar Alheyane. "Deep Convolutional Neural Networks for breast cancer screening". In: *Computer Methods and Programs in Biomedicine* 157 (2018), pp. 19–30. ISSN: 0169-2607. DOI: <https://doi.org/10.1016/j.cmpb.2018.01.011>. URL: <https://www.sciencedirect.com/science/article/pii/S0169260717301451>.
- [17] Pedro Costa et al. "End-to-End Adversarial Retinal Image Synthesis". In: *IEEE Transactions on Medical Imaging* 37.3 (2018), pp. 781–791. DOI: 10.1109/TMI.2017.2759102.
- [18] Zhipeng Deng et al. "Multi-scale object detection in remote sensing imagery with convolutional neural networks". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 145 (May 2018). DOI: 10.1016/j.isprsjprs.2018.04.003.
- [19] Neeraj Dhungel, Gustavo Carneiro, and Andrew P. Bradley. "A deep learning approach for the analysis of masses in mammograms with minimal user intervention". In: *Medical Image Analysis* 37 (2017), pp. 114–128. ISSN: 1361-8415. DOI: <https://doi.org/10.1016/j.media.2017.01.009>. URL: <https://www.sciencedirect.com/science/article/pii/S136184151730018X>.
- [20] Kaiwen Duan et al. "CenterNet: Keypoint Triplets for Object Detection". In: *CoRR* abs/1904.08189 (2019). arXiv: 1904.08189. URL: <http://arxiv.org/abs/1904.08189>.

- [21] Melanie Ducoffe and Frederic Precioso. “Adversarial Active Learning for Deep Networks: a Margin Based Approach”. In: (Feb. 2018).
- [22] Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Monographs on Statistics and Applied Probability 57. Boca Raton, Florida, USA: Chapman & Hall/CRC, 1993.
- [23] Sina Famouri et al. “Breast Mass Detection With Faster R-CNN: On the Feasibility of Learning From Noisy Annotations”. In: *IEEE Access* PP (Apr. 2021), pp. 1–1. DOI: 10.1109/ACCESS.2021.3072997.
- [24] Meng Fang, Yuan Li, and Trevor Cohn. “Learning how to Active Learn: A Deep Reinforcement Learning Approach”. In: *CoRR* abs/1708.02383 (2017). arXiv: 1708.02383. URL: <http://arxiv.org/abs/1708.02383>.
- [25] Vincent François-Lavet et al. “An Introduction to Deep Reinforcement Learning”. In: *CoRR* abs/1811.12560 (2018). arXiv: 1811.12560. URL: <http://arxiv.org/abs/1811.12560>.
- [26] Yarin Gal and Zoubin Ghahramani. *Bayesian Convolutional Neural Networks with Bernoulli Approximate Variational Inference*. 2016. arXiv: 1506.02158 [stat.ML].
- [27] Yarin Gal and Zoubin Ghahramani. *Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning*. 2016. arXiv: 1506.02142 [stat.ML].
- [28] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. “Deep Bayesian Active Learning with Image Data”. In: *CoRR* abs/1703.02910 (2017). arXiv: 1703.02910. URL: <http://arxiv.org/abs/1703.02910>.
- [29] Cyrus Ghajar and Mina Bissell. “Metastasis: Pathways of parallel progression”. In: *Nature* 540 (Dec. 2016). DOI: 10.1038/nature21104.
- [30] Ross Girshick. “Fast R-CNN”. In: *2015 IEEE International Conference on Computer Vision (ICCV)*. 2015, pp. 1440–1448. DOI: 10.1109/ICCV.2015.169.
- [31] Ross Girshick et al. “Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation”. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 580–587. DOI: 10.1109/CVPR.2014.81.
- [32] Daniel Gissin and Shai Shalev-Shwartz. “Discriminative Active Learning”. In: *CoRR* abs/1907.06347 (2019). arXiv: 1907.06347. URL: <http://arxiv.org/abs/1907.06347>.
- [33] Chuan Guo et al. *On Calibration of Modern Neural Networks*. 2017. arXiv: 1706.04599 [cs.LG].

- [34] Jennifer S. Haas. “The Complexity of Achieving the Promise of Precision Breast Cancer Screening”. In: *JNCI: Journal of the National Cancer Institute* 109.5 (Jan. 2017). djw301. ISSN: 0027-8874. DOI: 10.1093/jnci/djw301. eprint: <https://academic.oup.com/jnci/article-pdf/109/5/djw301/17497530/djw301.pdf>. URL: <https://doi.org/10.1093/jnci/djw301>.
- [35] Ali Harakeh, Michael Smart, and Steven Waslander. “BayesOD: A Bayesian Approach for Uncertainty Estimation in Deep Object Detectors”. In: May 2020, pp. 87–93. DOI: 10.1109/ICRA40945.2020.9196544.
- [36] Kaiming He et al. “Deep Residual Learning for Image Recognition”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.
- [37] Kaiming He et al. “Mask R-CNN”. In: *2017 IEEE International Conference on Computer Vision (ICCV)*. 2017, pp. 2980–2988. DOI: 10.1109/ICCV.2017.322.
- [38] William Hsu and Anne C Hoyt. “Using Time as a Measure of Impact for AI Systems: Implications in Breast Screening”. In: *Radiology. Artificial intelligence* 1.4 (2019), e190107. ISSN: 2638-6100. DOI: 10.1148/ryai.2019190107. URL: <https://europepmc.org/articles/PMC8017416>.
- [39] Gao Huang et al. “Densely Connected Convolutional Networks”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 2261–2269. DOI: 10.1109/CVPR.2017.243.
- [40] Gao Huang et al. *Snapshot Ensembles: Train 1, get M for free*. 2017. arXiv: 1704.00109 [cs.LG].
- [41] Jonathan Huang et al. “Speed/accuracy trade-offs for modern convolutional object detectors”. In: *CoRR* abs/1611.10012 (2016). arXiv: 1611.10012. URL: <http://arxiv.org/abs/1611.10012>.
- [42] Sergey Ioffe and Christian Szegedy. “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift”. In: *CoRR* abs/1502.03167 (2015). arXiv: 1502.03167. URL: <http://arxiv.org/abs/1502.03167>.
- [43] Pavel Izmailov et al. “Averaging Weights Leads to Wider Optima and Better Generalization”. In: *CoRR* abs/1803.05407 (2018). arXiv: 1803.05407. URL: <http://arxiv.org/abs/1803.05407>.
- [44] Alex Kendall and Yarin Gal. “What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?” In: *CoRR* abs/1703.04977 (2017). arXiv: 1703.04977. URL: <http://arxiv.org/abs/1703.04977>.

- [45] Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. “BatchBALD: Efficient and Diverse Batch Acquisition for Deep Bayesian Active Learning”. In: *CoRR* abs/1906.08158 (2019). arXiv: 1906.08158. URL: <http://arxiv.org/abs/1906.08158>.
- [46] Ksenia Konyushkova, Raphael Sznitman, and Pascal Fua. “Introducing Geometry in Active Learning for Image Segmentation”. In: Dec. 2015, pp. 2974–2982. DOI: 10.1109/ICCV.2015.340.
- [47] Thijs Kooi et al. “Large scale deep learning for computer aided detection of mammographic lesions”. In: *Medical Image Analysis* 35 (2017), pp. 303–312. ISSN: 1361-8415. DOI: <https://doi.org/10.1016/j.media.2016.07.007>. URL: <https://www.sciencedirect.com/science/article/pii/S1361841516301244>.
- [48] Florian Kraus and Klaus Dietmayer. “Uncertainty Estimation in One-Stage Object Detection”. In: *CoRR* abs/1905.10296 (2019). arXiv: 1905.10296. URL: <http://arxiv.org/abs/1905.10296>.
- [49] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Commun. ACM* 60.6 (May 2017), 84–90. ISSN: 0001-0782. DOI: 10.1145/3065386. URL: <https://doi.org/10.1145/3065386>.
- [50] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. *Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles*. 2017. arXiv: 1612.01474 [stat.ML].
- [51] Hei Law and Jia Deng. “CornerNet: Detecting Objects as Paired Keypoints”. In: *CoRR* abs/1808.01244 (2018). arXiv: 1808.01244. URL: <http://arxiv.org/abs/1808.01244>.
- [52] Rebecca Lee et al. “A curated mammography data set for use in computer-aided detection and diagnosis research”. In: *Scientific Data* 4 (Dec. 2017), p. 170177. DOI: 10.1038/sdata.2017.177.
- [53] Wei Liu et al. “SSD: Single Shot MultiBox Detector”. In: *CoRR* abs/1512.02325 (2015). arXiv: 1512.02325. URL: <http://arxiv.org/abs/1512.02325>.
- [54] Zimo Liu et al. “Deep Reinforcement Active Learning for Human-in-the-Loop Person Re-Identification”. In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 6121–6130. DOI: 10.1109/ICCV.2019.00622.
- [55] Ilya Loshchilov and Frank Hutter. “SGDR: Stochastic Gradient Descent with Restarts”. In: *CoRR* abs/1608.03983 (2016). arXiv: 1608.03983. URL: <http://arxiv.org/abs/1608.03983>.

- [56] William Lotter, G. Sorensen, and D. Cox. “A Multi-scale CNN and Curriculum Learning Strategy for Mammogram Classification”. In: *DLMIA/ML-CDS@MICCAI*. 2017.
- [57] Sébastien Lugan et al. “Secure Architectures Implementing Trusted Coalitions for Blockchain Distributed Learning (TCLearn)”. In: *CoRR* abs/1906.07690 (2019). arXiv: 1906.07690. URL: <http://arxiv.org/abs/1906.07690>.
- [58] Dhruv Mahajan et al. *Exploring the Limits of Weakly Supervised Pretraining*. May 2018.
- [59] Andrew McCallum and Kamal Nigam. “A Comparison of Event Models for Naive Bayes Text Classification”. In: *Work Learn Text Categ* 752 (May 2001).
- [60] Scott Mayer McKinney et al. “International evaluation of an AI system for breast cancer screening”. In: *Nature* 577 (2020), pp. 89–94.
- [61] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. “DeepFool: a simple and accurate method to fool deep neural networks”. In: *CoRR* abs/1511.04599 (2015). arXiv: 1511.04599. URL: <http://arxiv.org/abs/1511.04599>.
- [62] Ines Moreira et al. “INbreast: Toward a Full-field Digital Mammographic Database”. In: *Academic radiology* 19 (Nov. 2011), pp. 236–48. DOI: 10.1016/j.acra.2011.09.014.
- [63] *National Cancer Institute*. “BRCA Gene Mutations: Cancer Risk and Genetic Testing”. <https://www.cancer.gov/about-cancer/causes-prevention/genetics/brca-fact-sheet>. Accessed: 2021-20-8.
- [64] Houssami Nehmat. “Overdiagnosis of breast cancer in population screening: does it make breast screening worthless?” In: *Cancer Biology Medicine* 14 (Mar. 2017), pp. 1–8. DOI: 10.20892/j.issn.2095-3941.2016.0050.
- [65] Luke Oakden-Rayner. “The Rebirth of CAD: How Is Modern AI Different from the CAD We Know?” In: *Radiology: Artificial Intelligence* 1.3 (2019), e180089. DOI: 10.1148/ryai.2019180089. eprint: <https://doi.org/10.1148/ryai.2019180089>. URL: <https://doi.org/10.1148/ryai.2019180089>.
- [66] Mei-Sing Ong and Kenneth D. Mandl. “National Expenditure For False-Positive Mammograms And Breast Cancer Overdiagnoses Estimated At \$4 Billion A Year”. In: *Health Affairs* 34.4 (2015), pp. 576–583. DOI: 10.1377/hlthaff.2014.1087. eprint: <https://doi.org/10.1377/hlthaff.2014.1087>. URL: <https://doi.org/10.1377/hlthaff.2014.1087>.

- [67] Ian Osband. “Risk versus Uncertainty in Deep Learning : Bayes , Bootstrap and the Dangers of Dropout”. In: 2016.
- [68] Hieu Pham et al. “Meta Pseudo Labels”. In: *CoRR* abs/2003.10580 (2020). arXiv: 2003.10580. URL: <https://arxiv.org/abs/2003.10580>.
- [69] *Radiology Assistant "Bi-RADS for Mammography and Ultrasound 2013"*. <https://radiologyassistant.nl/breast/bi-rads/bi-rads-for-mammography-and-ultrasound-2013>. Accessed: 2021-04-8.
- [70] Farheen Ramzan et al. “A Deep Learning Approach for Automated Diagnosis and Multi-Class Classification of Alzheimer’s Disease Stages Using Resting-State fMRI and Residual Neural Networks”. In: *Journal of Medical Systems* 44 (Dec. 2019). DOI: 10.1007/s10916-019-1475-2.
- [71] Vijay Rao et al. “How Widely Is Computer-Aided Detection Used in Screening and Diagnostic Mammography?” In: *Journal of the American College of Radiology : JACR* 7 (Oct. 2010), pp. 802–5. DOI: 10.1016/j.jacr.2010.05.019.
- [72] Joseph Redmon and Ali Farhadi. “YOLO9000: Better, Faster, Stronger”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 6517–6525. DOI: 10.1109/CVPR.2017.690.
- [73] Joseph Redmon and Ali Farhadi. “YOLOv3: An Incremental Improvement”. In: *CoRR* abs/1804.02767 (2018). arXiv: 1804.02767. URL: <http://arxiv.org/abs/1804.02767>.
- [74] Joseph Redmon et al. “You Only Look Once: Unified, Real-Time Object Detection”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 779–788. DOI: 10.1109/CVPR.2016.91.
- [75] Shaoqing Ren et al. “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks”. In: *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*. NIPS’15. Montreal, Canada: MIT Press, 2015, 91–99.
- [76] Mauricio Reyes et al. “On the Interpretability of Artificial Intelligence in Radiology: Challenges and Opportunities”. In: *Radiology: Artificial Intelligence* 2 (May 2020), e190043. DOI: 10.1148/ryai.2020190043.
- [77] Dezső Ribli et al. “Detecting and classifying lesions in mammograms with Deep Learning”. In: *arXiv preprint arXiv:1707.08401* (2017).
- [78] Nicholas Roy and Andrew McCallum. “Toward optimal active learning through monte carlo estimation of error reduction”. In: (Jan. 2001).

- [79] Nicholas Roy and Andrew McCallum. “Toward Optimal Active Learning through Sampling Estimation of Error Reduction”. In: *Proceedings of the Eighteenth International Conference on Machine Learning*. ICML ’01. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001, 441–448. ISBN: 1558607781.
- [80] Debbie Saslow et al. “American Cancer Society Guidelines for Breast Screening with MRI as an Adjunct to Mammography”. In: *CA: A Cancer Journal for Clinicians* 57.2 (2007), pp. 75–89. DOI: <https://doi.org/10.3322/canjclin.57.2.75>. eprint: <https://acsjournals.onlinelibrary.wiley.com/doi/pdf/10.3322/canjclin.57.2.75>. URL: <https://acsjournals.onlinelibrary.wiley.com/doi/abs/10.3322/canjclin.57.2.75>.
- [81] Jürgen Schmidhuber. “Deep Learning in Neural Networks: An Overview”. In: *CoRR* abs/1404.7828 (2014). arXiv: 1404.7828. URL: <http://arxiv.org/abs/1404.7828>.
- [82] “Screening for Breast Cancer: U.S. Preventive Services Task Force Recommendation Statement”. In: *Annals of Internal Medicine* 164.4 (2016). PMID: 26757170, pp. 279–296. DOI: 10.7326/M15-2886. eprint: <https://www.acpjournals.org/doi/pdf/10.7326/M15-2886>. URL: <https://www.acpjournals.org/doi/abs/10.7326/M15-2886>.
- [83] Ramprasaath R. Selvaraju et al. “Grad-CAM: Why did you say that? Visual Explanations from Deep Networks via Gradient-based Localization”. In: *CoRR* abs/1610.02391 (2016). arXiv: 1610.02391. URL: <http://arxiv.org/abs/1610.02391>.
- [84] Ozan Sener and Silvio Savarese. *Active Learning for Convolutional Neural Networks: A Core-Set Approach*. 2018. arXiv: 1708.00489 [stat.ML].
- [85] Burr Settles. *Active Learning Literature Survey*. Computer Sciences Technical Report 1648. University of Wisconsin–Madison, 2009. URL: <http://axon.cs.byu.edu/~martinez/classes/778/Papers/settles.activelearning.pdf>.
- [86] Burr Settles and Mark Craven. “An Analysis of Active Learning Strategies for Sequence Labeling Tasks”. In: *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*. Honolulu, Hawaii: Association for Computational Linguistics, Oct. 2008, pp. 1070–1079. URL: <https://aclanthology.org/D08-1112>.

- [87] Burr Settles, Mark Craven, and Soumya Ray. “Multiple-Instance Active Learning”. In: *Advances in Neural Information Processing Systems*. Ed. by J. Platt et al. Vol. 20. Curran Associates, Inc., 2008. URL: <https://proceedings.neurips.cc/paper/2007/file/a1519de5b5d44b31a01de013b9b51a80-Paper.pdf>.
- [88] Li Shen et al. “Deep Learning to Improve Breast Cancer Detection on Screening Mammography”. In: *Scientific Reports* 9.1 (2019). ISSN: 2045-2322. DOI: 10.1038/s41598-019-48995-4. URL: <http://dx.doi.org/10.1038/s41598-019-48995-4>.
- [89] D’Orsi CJ Bassett LW et al. ACR BI-RADS® Mammography. In: ACR BI-RADS® Atlas Breast Imaging Reporting Sickles EA and American College of Radiology; 2013. Data System. Reston VA.
- [90] David Silver et al. “Mastering the game of Go with deep neural networks and tree search”. In: *Nature* 529 (Jan. 2016), pp. 484–489. DOI: 10.1038/nature16961.
- [91] K. Simonyan and Andrew Zisserman. “Very Deep Convolutional Networks for Large-Scale Image Recognition”. In: *CoRR* abs/1409.1556 (2015).
- [92] Jai Singh and Bagavathi Nagarajan. “Automatic diagnosis of mammographic abnormalities based on hybrid features with learning classifier”. In: *Computer methods in biomechanics and biomedical engineering* 16 (Jan. 2012). DOI: 10.1080/10255842.2011.639015.
- [93] Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. “Variational Adversarial Active Learning”. In: *CoRR* abs/1904.00370 (2019). arXiv: 1904.00370. URL: <http://arxiv.org/abs/1904.00370>.
- [94] Asim Smailagic et al. “MedAL: Accurate and Robust Deep Active Learning for Medical Image Analysis”. In: *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. 2018, pp. 481–488. DOI: 10.1109/ICMLA.2018.00078.
- [95] Asim Smailagic et al. “O-MedAL: Online active deep learning for medical image analysis”. In: *WIREs Data Mining and Knowledge Discovery* 10.4 (2020). ISSN: 1942-4795. DOI: 10.1002/widm.1353. URL: <http://dx.doi.org/10.1002/widm.1353>.
- [96] Nitish Srivastava et al. “Dropout: A Simple Way to Prevent Neural Networks from Overfitting”. In: *Journal of Machine Learning Research* 15.56 (2014), pp. 1929–1958. URL: <http://jmlr.org/papers/v15/srivastava14a.html>.

- [97] *Statistical Odds & Ends "What is balanced accuracy"*. <https://statisticaloddsandends.wordpress.com/2020/01/23/what-is-balanced-accuracy/>. Accessed: 2021-18-6.
- [98] Christian Szegedy et al. "Going deeper with convolutions". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 1–9. DOI: 10.1109/CVPR.2015.7298594.
- [99] Christian Szegedy et al. *Intriguing properties of neural networks*. 2014. arXiv: 1312.6199 [cs.CV].
- [100] Mingxing Tan and Quoc V. Le. "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks". In: *CoRR* abs/1905.11946 (2019). arXiv: 1905.11946. URL: <http://arxiv.org/abs/1905.11946>.
- [101] Ashish Vaswani et al. "Attention Is All You Need". In: *CoRR* abs/1706.03762 (2017). arXiv: 1706.03762. URL: <http://arxiv.org/abs/1706.03762>.
- [102] *W. C. R. Fund. "Breast cancer statistics"*. <https://www.wcrf.org/dietandcancer/cancer-trends/breast-cancer-statistics>. Accessed: 2021-04-6.
- [103] *W. H. Organization. "Breast cancer"*. <https://www.who.int/news-room/fact-sheets/detail/breast-cancer>. Accessed: 2021-04-4.
- [104] Keze Wang et al. "Cost-Effective Active Learning for Deep Image Classification". In: *CoRR* abs/1701.03551 (2017). arXiv: 1701.03551. URL: <http://arxiv.org/abs/1701.03551>.
- [105] Karl Weiss, Taghi Khoshgoftaar, and DingDing Wang. "A survey of transfer learning". In: *Journal of Big Data* 3 (May 2016). DOI: 10.1186/s40537-016-0043-6.
- [106] H. Gilbert Welch et al. "Breast-Cancer Tumor Size, Overdiagnosis, and Mammography Screening Effectiveness". In: *New England Journal of Medicine* 375.15 (2016). PMID: 27732805, pp. 1438–1447. DOI: 10.1056/NEJMoa1600249. eprint: <https://doi.org/10.1056/NEJMoa1600249>. URL: <https://doi.org/10.1056/NEJMoa1600249>.
- [107] *Wikipedia "Mammography"*. <https://en.wikipedia.org/wiki/Mammography>. Accessed: 2021-26-6.
- [108] David H. Wolpert. "Stacked generalization". In: *Neural Networks* 5.2 (1992), pp. 241–259. ISSN: 0893-6080. DOI: [https://doi.org/10.1016/S0893-6080\(05\)80023-1](https://doi.org/10.1016/S0893-6080(05)80023-1). URL: <https://www.sciencedirect.com/science/article/pii/S0893608005800231>.

- [109] Mark Woodward and Chelsea Finn. “Active One-shot Learning”. In: *CoRR* abs/1702.06559 (2017). arXiv: 1702.06559. URL: <http://arxiv.org/abs/1702.06559>.
- [110] Xing Wu et al. “COVID-AL: The diagnosis of COVID-19 with deep active learning”. In: *Medical Image Analysis* 68 (2021), p. 101913. ISSN: 1361-8415. DOI: <https://doi.org/10.1016/j.media.2020.101913>. URL: <https://www.sciencedirect.com/science/article/pii/S1361841520302772>.
- [111] Jingjing Xie, Bing Xu, and Chuang Zhang. “Horizontal and Vertical Ensemble with Deep Representation for Classification”. In: *CoRR* abs/1306.2759 (2013). arXiv: 1306.2759. URL: <http://arxiv.org/abs/1306.2759>.
- [112] Adam Yala et al. “A Deep Learning Mammography-based Model for Improved Breast Cancer Risk Prediction”. In: *Radiology* 292.1 (2019). PMID: 31063083, pp. 60–66. DOI: 10.1148/radiol.2019182716. eprint: <https://doi.org/10.1148/radiol.2019182716>. URL: <https://doi.org/10.1148/radiol.2019182716>.
- [113] Lin Yang et al. *Suggestive Annotation: A Deep Active Learning Framework for Biomedical Image Segmentation*. 2017. arXiv: 1706.04737 [cs.CV].
- [114] Donggeun Yoo and In So Kweon. “Learning Loss for Active Learning”. In: *CoRR* abs/1905.03677 (2019). arXiv: 1905.03677. URL: <http://arxiv.org/abs/1905.03677>.
- [115] Jin Yuan et al. “Multi-criteria active deep learning for image classification”. In: *Knowledge-Based Systems* 172 (2019), pp. 86–94. ISSN: 0950-7051. DOI: <https://doi.org/10.1016/j.knosys.2019.02.013>. URL: <https://www.sciencedirect.com/science/article/pii/S0950705119300747>.
- [116] Chiyuan Zhang et al. “Understanding deep learning requires rethinking generalization”. In: *CoRR* abs/1611.03530 (2016). arXiv: 1611.03530. URL: <http://arxiv.org/abs/1611.03530>.
- [117] Tong Zhang and Frank J. Oles. “A Probability Analysis on the Value of Unlabeled Data for Classification Problems”. In: *Proc. 17th International Conf. on Machine Learning*. Morgan Kaufmann, San Francisco, CA, 2000, pp. 1191–1198.
- [118] Zhong-Qiu Zhao et al. “Object Detection with Deep Learning: A Review”. In: *CoRR* abs/1807.05511 (2018). arXiv: 1807.05511. URL: <http://arxiv.org/abs/1807.05511>.
- [119] Fedor Zhdanov. *Diverse mini-batch Active Learning*. 2019. arXiv: 1901.05954 [cs.LG].

- [120] Zongwei Zhou et al. “Active, continual fine tuning of convolutional neural networks for reducing annotation efforts”. In: *Medical Image Analysis* 71 (2021), p. 101997. ISSN: 1361-8415. DOI: 10.1016/j.media.2021.101997. URL: <http://dx.doi.org/10.1016/j.media.2021.101997>.
- [121] Zongwei Zhou et al. “Fine-Tuning Convolutional Neural Networks for Biomedical Image Analysis: Actively and Incrementally”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 4761–4772. DOI: 10.1109/CVPR.2017.506.

Appendix A: Active Learning algorithms

Python implementations are available with the link: <https://github.com/MaxZanella/Active-Learning-for-Mammography>.

Algorithm 2 Random sampling

Require: model, unlabeled_dataset, budget
selected_indices $\leftarrow$ random_choice([0...size(unlabeled_dataset)], budget)
return selected_indices

Algorithm 3 MaxEntropy uncertainty measure

Require: model, unlabeled_dataset, budget
predictions $\leftarrow$ predict(model, unlabeled_dataset)
entropy_measures $\leftarrow$ entropy(predictions)
selected_indices $\leftarrow$ argmax(entropy_measures, budget)
return selected_indices

Algorithm 4 Monte Carlo dropout

Require: model, unlabeled_dataset, budget, nb_MC_forward
activate dropout
for i in 0...nb_MC_forward **do**
 predictions $\leftarrow$ predictions + predict(model, unlabeled_dataset)
end for
predictions_mean $\leftarrow$ predictions/nb_MC_forward
entropy_measures $\leftarrow$ entropy(predictions_mean)
selected_indices $\leftarrow$ argmax(entropy_measures, budget)
return selected_indices

Algorithm 5 Ensemble Active Learning

Require: models, unlabeled_dataset, budget, uncertainty_measure (either entropy or variation ratio)
for i in 0...size(models) **do**
 predictions $\leftarrow$ predictions + predict(models[i], unlabeled_dataset)
end for
predictions_mean $\leftarrow$ predictions/size(models)
uncertainty $\leftarrow$ uncertainty_measure(predictions_mean)
selected_indices $\leftarrow$ argmax(uncertainty, budget)
return selected_indices

Algorithm 6 Greedy core-set

Require: model, labeled_dataset, unlabeled_dataset
unlabeled_features $\leftarrow$ extract_last_layer(model, unlabeled_dataset)
labeled_features $\leftarrow$ extract_last_layer(model, labeled_dataset)
U, L = size(unlabeled_dataset), size(labeled_dataset)
distances $\leftarrow$ [∞_0 ... ∞_{N-1}]
for i in 0...U **do**
 for j in 0...L **do**
 distances[i] $\leftarrow$ min(distances[i], euclidian_distance(unlabeled_features[i], labeled_features[j]))
 end for
end for
selected_indices $\leftarrow$ []
indice $\leftarrow$ argmax(distances, 1)
selected_indices $\leftarrow$ append(selected_indices, indice)
for k in 1...budget **do**
 for i in 0...U **do**
 distances $\leftarrow$ min(distances[i], euclidian_distances(distances[i], unlabeled_features[indice]))
 indice $\leftarrow$ argmax(distances, 1)
 selected_indices $\leftarrow$ append(selected_indices, indice)
 end for
end for
return selected_indices

Algorithm 7 Discriminative Active Learning

Require: model, unlabeled_dataset, labeled_dataset, budget, n_sampling
sampling_size = budget/n_sampling
features $\leftarrow$ extract_last_layer(model, dataset)
selected_indices $\leftarrow$ []
for i in 0...n_sampling **do**
 unlabeled_dataset_subset $\leftarrow$ random_choice(unlabeled_dataset,
 10*size(labeled_dataset))
 unlabeled_features $\leftarrow$ extract_last_layer(model, unlabeled_dataset_subset)
 labeled_features $\leftarrow$ extract_last_layer(model, labeled_dataset)
 train_dataset $\leftarrow$ concatenate(unlabeled_features, labeled_features)
 discriminative_model $\leftarrow$ train_discriminative(train_dataset)
 predictions $\leftarrow$ predict(discriminative_model, unlabeled_features)
 indices $\leftarrow$ argmax(predictions, sampling_size)
 selected_indices $\leftarrow$ concatenate(selected_indices, indices)
 unlabeled_dataset $\leftarrow$ remove(unlabeled_dataset, indices)
 labeled_dataset $\leftarrow$ concatenate(labeled_dataset, unlabeled_dataset[indices])
end for
return selected_indices

Algorithm 8 Learning Loss

Require: model, learning_loss_module, unlabeled_dataset
model, learning_loss_module $\leftarrow$ train_learning_loss(model, learning_loss_module)
predictions_loss $\leftarrow$ predict_loss(model, learning_loss_module, unlabeled_dataset)
selected_indices $\leftarrow$ argmax(predictions_loss, budget)
return selected_indices

Appendix B: Details on the experiments

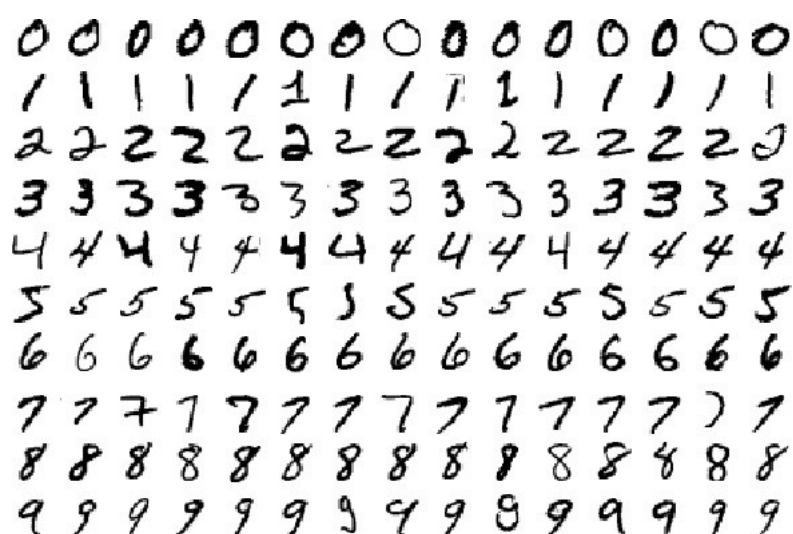


Figure B.1: A subset of the MNIST dataset. Source: Wikipedia

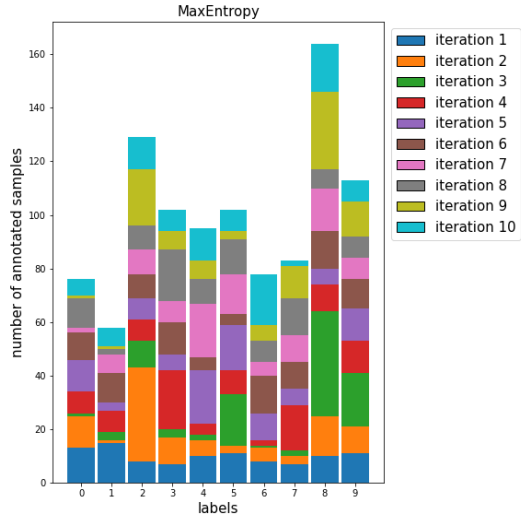


Figure B.2: Results of trial 3 for the deterministic MaxEntropy strategy.

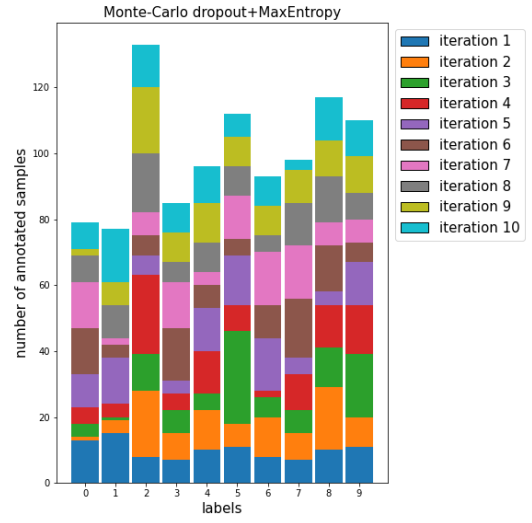


Figure B.3: Results of trial 3 for the Monte Carlo with MaxEntropy strategy.

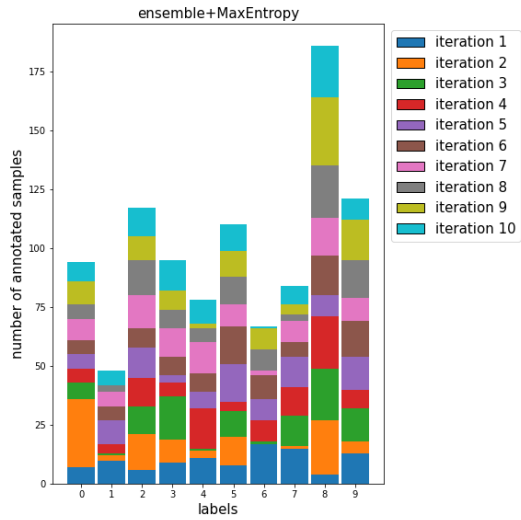


Figure B.4: Results of trial 1 for the ensemble method with MaxEntropy.

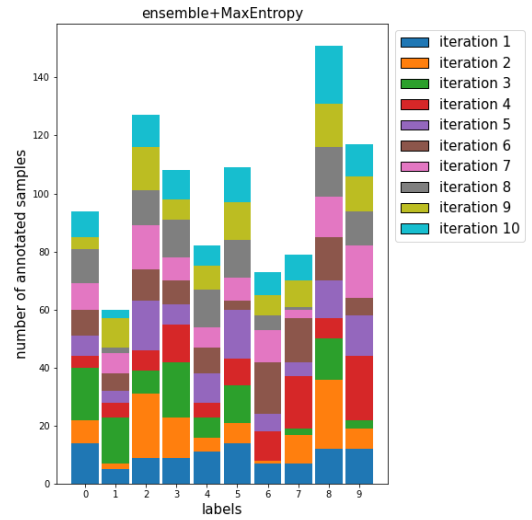


Figure B.5: Results of trial 2 for the ensemble method with MaxEntropy.

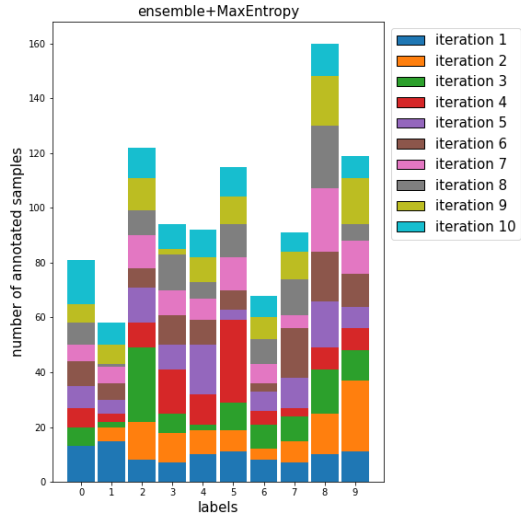


Figure B.6: Results of trial 3 for the ensemble method with MaxEntropy.

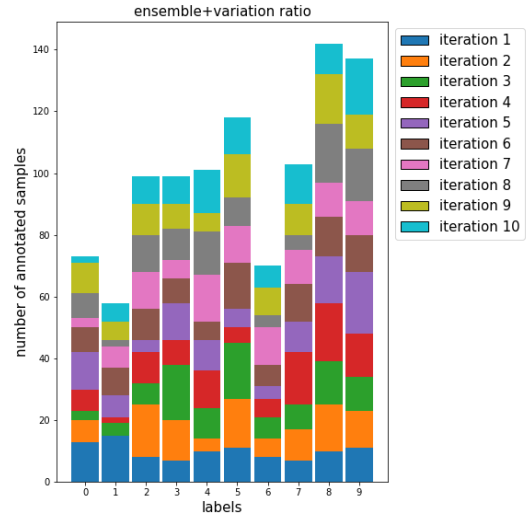


Figure B.7: Results of trial 3 for the ensemble method with variation ratio.

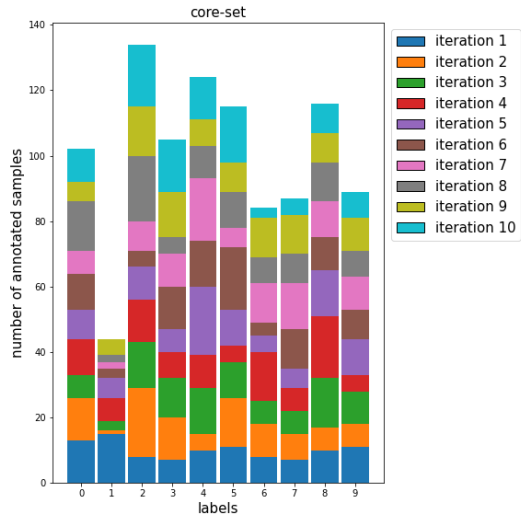


Figure B.8: Results of trial 3 for the core-set approach.

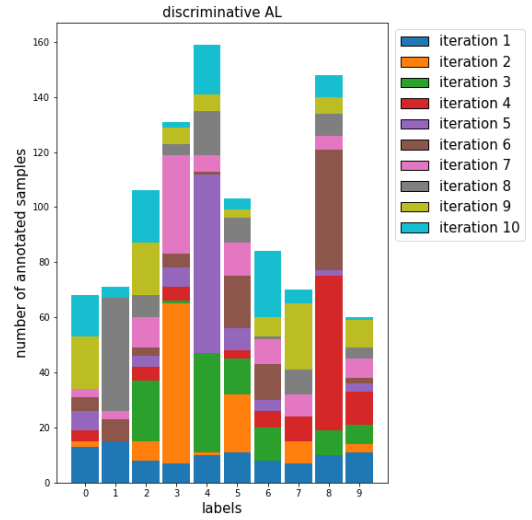


Figure B.9: Results of trial 3 for the discriminative AL strategy.

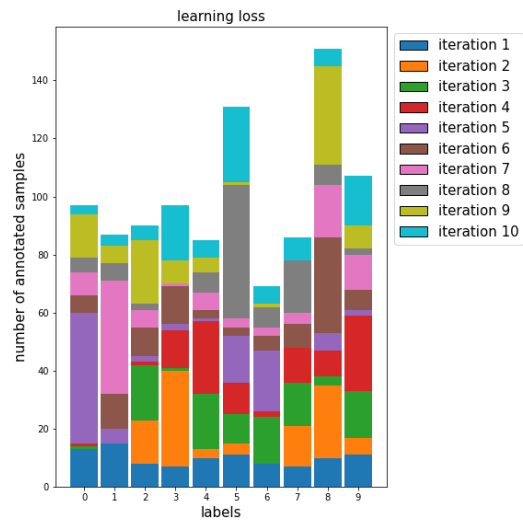


Figure B.10: Results of trial 3 for the learning loss strategy.

Appendix C: Active Learning for a detection model

This last experiment will use our Faster R-CNN trained in Chapter 2. It aims to show that Active Learning can be used in detection tasks and the feasibility of the **Active preliminary results** concept introduced in Section 1.6.

We measure the prediction’s uncertainty of the head classifier. The idea is that if the head has difficulties classifying a region proposed by the RPN, it could be an interesting area that contains an abnormality. We try this method with the MaxEntropy measure, limiting to one question per image and taking the entropy measures in decreasing order. We take the model trained in Chapter 2 and assess the method on the test set (348 images the model has never seen). A question is considered to indicate an abnormality if the region proposed by the RPN (and refined by the head) and the ground truth bounding box have an IoU of 0.25 or greater (we assume it is enough for the senologist to understand the requested annotations). Results are shown in Table C.1.

Number of questions	Number of abnormalities found
10	5
20	10
30	13
40	18
50	25
100	52
200	113
358	186

Table C.1: Results for MaxEntropy on Faster R-CNN.

We observe that on average half of the questions asked point out an abnormality, showing that this simple method may work. However, two major concerns arise.

- The model may not be able to ask annotations for all the regions of interest

of a mammogram at once, engendering concerns about how to train models with incomplete annotations.

- A very difficult to detect abnormality may never be pointed out even if it would be, by definition, very useful annotations to train on. This issue may diminish with increasing performance of the model.

We leave these reflections for further studies.

UNIVERSITÉ CATHOLIQUE DE LOUVAIN

École polytechnique de Louvain

Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve, Belgique | www.uclouvain.be/epl