THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par l'Université Toulouse 1 Capitole

Présentée et soutenue par

Ismat Ara RESHMA

Le 20 septembre 2021

Influence de la distribution des classes et évaluation en apprentissage profond - Application à la détection du cancer sur des images histologiques

Ecole doctorale : **EDMITT - Ecole Doctorale Mathématiques, Informatique et Télécommunications de Toulouse**

Spécialité : **Informatique et Télécommunications**

Unité de recherche :
IRIT : Institut de Recherche en Informatique de Toulouse

Thèse dirigée par
Hervé LUGA et Josiane MOTHE

Jury

Mme Su RUAN, Rapporteure
M. Henning MÜLLER, Rapporteur
M. Philippe MULHEM, Examinateur
M. Hervé LUGA, Directeur de thèse
Mme Josiane MOTHE, Co-directrice de thèse

Dedicated to my parents

Contents

List of Abbreviations

AUC Area Under the Curve

CAD Computer-Aided Diagnosis

CNN Convolutional Neural Network

FCNN Fully Convolutional Neural Network

FN False-Negative

FP False-Positive

FPR False-Positive Rate

PPV Positive Predictive Value

PR Precision-Recall

PR-AUC Area Under the PR Curve

PR-T Threshold-based Precision Recall

ROC Receiver Operating Characteristics

ROC-AUC Area Under the ROC Curve

ROI Region of Interest

TN True-Negative

TP True-Positive

TPR True-Positive Rate

WSI While Slide Image

Acknowledgement

The completion of this thesis is not a result of my individual effort but is an aggregate of co-operation of many other people. I want to acknowledge a selected few, although many others remain unacknowledged but greatly appreciated.

I was fortunate enough to get multiple advisors in this sophisticated journey of my PhD without whom this thesis would not be possible. My first and foremost gratitude goes to them.

At first, I would like to mention my first advisor in IRIT, Prof. Josiane MOTHE. My bonding with her is older than my PhD journey. It has started when I join her team as a research engineer. I can and will remember the day of submitting my first assignment to her after one week of my joining. On that day, one magical word from her -“EXCELLENT”- regain my confidence and boost up my enthusiasm to pass through this sophisticated journey. I truly appreciate her encouraging attitude. I also appreciate all her contributions, including knowledge sharing, ideas, guidance, continuous and tireless effort in improving my scientific writing, and funding to make my PhD experience productive and stimulating. Her qualities of hard work, sincerity, patience, parallel management of multidimensional tasks have motivated me a lot and will motivate me in my entire life. As a woman, I always desire to have such kind of qualities. Along with being a good teacher, she has been a good human being as well. I could share all of the problems that I faced during the last three years and always got an optimal solution from her.

Along with her, I would like to thank my other advisors Prof. Herve LUGA and Sylvain CUSSAT-BLANC for believing in me and giving me the opportunity to continue my research on my topic of interest but guided me in the right direction whenever needed. I appreciate all their supports, guidelines, patience, humbleness, cooperative attitude throughout this journey. Their constructive comments and suggestions have a great impact on this thesis.

My sincere gratitude also goes to Prof. Su RUAN from the Université de Rouen and Prof. Henning MÜLLER from the University of Applied Sciences Western Switzerland for accepting to review my thesis manuscript in their busy schedule and for their valuable remarks. I also would like to thank all other internal and external members of my defense committee for their valuable roles in my thesis defense.

I would like to thank Prof. Pierre BROUSSET, Dr. Camille FRANCHET, and Dr. Margot GASPARD from the Institut Universitaire du Cancer Toulouse Oncopole (IUCT-Oncopole) for their excellent collaboration on this thesis. We had several meetings, and I received valuable comments and suggestions from them at various steps of the work from data acquisition and annotation to verifying the outcome of the projects. One of the major data sets (the MLNTO) utilized in this thesis is annotated and provided by them. I also would like to thank Prof. Radu Tudor IONESCU from the University of Bucharest for his impactful collaboration in the project of class distribution analysis. I really appreciate his valuable comments and suggestions during our meeting and writing articles.

My special thanks and sincere gratitude go to my Master's advisor Prof. Masaki AONO from the Toyohashi University of Technology, Japan, who constructed the base of my research career in machine learning and image processing. In addition, I would like to thank all of my teachers from primary school to the university level. Their support, inspiration, guideline, and advice must have a strong influence on my entire life.

I would like to thank Prof. Olivier TESTE, the team leader of the SIG team, and all other team members for providing such a welcoming environment to work with. I am also grateful to all my past and present colleagues in the IRIT lab for their friendships, stimulating discussions, and assistance when needed. Especially Nathalie NEPTUNE and Hamdi Ben HAMADOU, thank you very much; you have been the best translators, problem solvers, and friends in our life in France. Without you, life could have been difficult as a person from a different culture. Special mention also goes to Sonia MEJBRI to share her project code with me at the beginning of my joining as a research engineer.

I would also like to show my gratitude to all the staff of IRIT, UT1, and EDMITT who helped me with the different official tasks. Especially, it is my pleasure to mention Jacques THOMAZEAU, who helped me a lot in solving technical problems with OSIRIM during the implementation of the projects. My special thanks also go to Chloé BOURBON for being so cooperative in any official task and being such a good friend.

I would like to acknowledge the general support of the Ministre de l'Enseignement Supérieur, de la Recherche et de l'Innovation (MESRI) for providing me the scholarship for pursuing my PhD.

I want to pay my deep gratitude to all my family members who have been the source of inspiration and support to pass this long academic journey.

My parents formed the base to reach until here, especially, my mother. She is my first teacher, motivator, and source of all positive energy till now. Being a single mother, we all four siblings are the witness of her struggle, sacrifices to bring us in our current positions after passing away of my father at our early age. No word is enough to appreciate her. I would like to appreciate my maternal uncles, who always have been a great support to us and dreamt to achieve this prestigious designation. My deepest love and thanks go to my unborn baby in my womb to bring me all the happiness in the toughest time of my PhD. You have been the mummy's tonic to remove all the stresses during the last eight months. Last but not the least, my heartiest appreciation goes to my lifeline, my husband. Thank you very much for all the delicious foods you cooked for me whenever I was unable due to the heavy workload of PhD; thank you very much for working as a substitute for Jacques in solving OSIRIM problems and directing me to the right way to solve technical problems sometimes; thank you very much for holding my hand tight whatever the situation is and motivating me in a very positive way. Although you did not teach me how to live without you, you teach me how to solve any academic problem without you; I really appreciate that also.

Thank you very much all.

Abstract

Cancer is a fatal disease considered the second leading cause of death. Any advances in diagnosis and detection of cancer are thus crucial to save lives. The analysis of histological images -also known as Whole Slide Images (WSIs)- is considered as the gold standard in cancer diagnosis and staging. The pathologists' manual analysis of WSIs is still the primary diagnosis process. It is time-consuming, laborious, prone to error, and difficult to grade in a reproducible manner. Computer-aided diagnosis techniques can assist pathologists in their workflow. Machine learning techniques, specifically deep learning algorithms, such as Convolutional Neural Networks (CNNs), are widely used in various domains that involve image analysis. The success of CNN models, however, depends on several hyper-parameter settings, such as the network architecture, the data used to train the model, and the class distribution of the training data. To the best of our knowledge, among the hyper-parameters, the class distribution of the training data is not studied yet in the literature for the WSI data, while it could be one of the most important criteria to regulate the model performance. One of the aims of this thesis is to study in-depth the impact of class distribution both at the training stage and at the test or forecasting stage. Another aim of this thesis is related to evaluation in a broader sense. We studied ways of evaluating the results that fit more the pathologist's goals and solve the issues of current metrics that suffer from their incapacity to distinguish models in many cases, lacking information regarding false predictions and being optimistic in the case of imbalanced data. Considering both the class distribution and the evaluation for cancer detection from WSIs, the specific contributions of this thesis are as follows:

The first main contribution of this thesis is to investigate the effectiveness of the balanced distribution in automatic cancer detection which is used in many studies. We propose a systematic approach to analyze the class distribution of the WSI data in the training set; put forward different hypotheses on the class distribution and test those hypotheses using three data sets and two CNN architectures, the U-net and the group equivariant convolutional network (G-CNN). We also introduce a patch-based (i.e., image region-based) evaluation method over the usual pixel-based one to obtain a better match in comparison to how a pathologist checks images. We found that the balanced distribution is not optimal for CNN training for cancer detection from WSI, rather with the class-biased distribution, it is possible to inflect the model toward the desired accuracy (e.g., toward recall or precision). These results are a step forward to understand the model behavior towards the different distributions of classes in the training set.

The second main contribution of this thesis is to develop a continuous threshold-based representation of precision and recall (PR-T) curves as an alternative to the Receiver Operating Characteristics (ROC) and Precision-Recall (PR) curves, the state-of-the-art evaluation metrics in binary classification as is cancer detection. Additionally, we developed end-to-end algorithms to compute the mean PR curve and the mean Area Under the Curve (PR-AUC).

Keywords: Class distribution analysis, Evaluation method, Histological image, Deep learning, Cancer diagnosis, Image segmentation and classification

Résumé

Le cancer est une maladie mortelle considérée comme la deuxième cause de décès. Toute avancée dans le diagnostic et la détection du cancer est donc cruciale pour sauver des vies. L'analyse d'images histologiques - également appelées Whole Slide Images (WSI) - est considérée comme la référence dans le diagnostic et l'étude du stade du cancer. L'analyse manuelle de ces images par les pathologistes reste le principal processus de diagnostic. Il prend du temps, est laborieux, sujet aux erreurs et difficile à évaluer de manière reproductible. Les techniques de diagnostic assisté par ordinateur peuvent aider les pathologistes dans leur travail. Les techniques d'apprentissage automatique, en particulier les algorithmes d'apprentissage profond, tels que les réseaux de neurones convolutifs (CNN), sont largement utilisés dans divers domaines dont l'analyse d'images. Le succès des modèles CNN dépend cependant de plusieurs hyper-paramètres, tels que l'architecture du réseau, les données utilisées pour entraîner le modèle et la distribution des données d'entraînement. A notre connaissance, parmi les hyper-paramètres, la distribution des données d'entraînement n'est pas encore étudiée dans la littérature pour les données WSI, alors qu'elle pourrait être l'un des critères les plus importants pour réguler les performances du modèle. L'un des objectifs de cette thèse est d'étudier en profondeur l'impact de la répartition des classes tant au stade de l'apprentissage qu'au stade du test ou de la prévision. Un autre objectif de cette thèse est lié à l'évaluation au sens large. Nous avons étudié des moyens d'évaluer les résultats qui correspondent davantage aux objectifs du pathologiste et résolvent les problèmes des métriques actuelles qui souffrent de leur incapacité à distinguer les modèles dans de nombreux cas, manquent d'informations concernant les fausses prédictions et sont optimistes dans le cas de données déséquilibrées. Considérant à la fois la distribution des classes et l'évaluation de la détection du cancer à partir des WSI, les contributions spécifiques de cette thèse sont les suivantes:

La première contribution principale de cette thèse est d'étudier l'efficacité de la distribution équilibrée dans la détection automatique du cancer qui est utilisée dans de nombreuses études. Nous proposons une approche systématique pour analyser la distribution des classes des données WSI dans l'ensemble d'apprentissage, pour proposer différentes hypothèses sur la distribution des classes et tester ces hypothèses en utilisant trois ensembles de données et deux architectures CNN, le réseau U-net et le réseau convolutif équivariant de groupe (G-CNN). Nous introduisons également une méthode d'évaluation basée sur les régions de l'image alternative à la méthode habituelle basée sur les pixels. Elle permet d'obtenir une meilleure correspondance par rapport à la façon dont un pathologiste vérifie les images. Nous avons constaté que la distribution équilibrée n'est pas optimale pour l'entrainement d'un CNN, et qu'avec la distribution biaisée par classe, il est possible d'infléchir le modèle vers la précision souhaitée (par exemple, vers le rappel ou la précision). Ces résultats constituent une avancée pour comprendre le comportement du modèle vis-à-vis des différentes distributions de classes dans l'ensemble d'apprentissage.

La deuxième contribution principale de cette thèse est de développer une représentation continue basée sur un seuil des courbes de précision et de rappel (PR-T) comme alternative aux courbes de caractéristiques de fonctionnement du récepteur (ROC) et de précision-rappel (PR), les métriques d'évaluation usuelles en classification binaire. De plus, nous avons développé des algorithmes de bout en bout pour calculer la courbe PR moyenne et la moyenne de l'aire sous la courbe (PR-AUC).

Mots clés : Analyse de la distribution des classes, Méthode d'évaluation, Image histologique, Apprentissage profond, Diagnostic du cancer, Segmentation et classification d'images

Publications

The research reported in this thesis has resulted in the following publications:

1. Sonia Mejbri, Camille Franchet, **Ismat Ara Reshma**, Josiane Mothe, Pierre Brousset, and Emmanuel Faure. Deep Analysis of CNN Settings for New Cancer whole-slide Histological Images Segmentation: the Case of Small Training Sets. In: 6th International Conference on Bioimaging; pages 120-128; Prague, Czech Republic; 22-24 February 2019 [Mejbri et al., 2019]. Access: `https://hal.archives-ouvertes.fr/hal-02092926/document`.

2. **Ismat Ara Reshma**, Margot Gaspard, Camille Franchet, Pierre Brousset, Emmanuel Faure, Sonia Mejbri, and Josiane Mothe. Training Set Class Distribution Analysis for Deep Learning Model – Application to Cancer Detection. In: 1st International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI); pages 123-127; Barcelona, Spain; 20-22 March 2019 [Reshma et al., 2019]. Access: `https://hal.archives-ouvertes.fr/hal-02891748/document`.

3. **Ismat Ara Reshma**, Sylvain Cussat-Blanc, Radu Tudor Ionescu, Hervé Luga, and Josiane Mothe. Natural vs Balanced Distribution in Deep Learning on Whole Slide Images for Cancer Detection. In: Proceedings of the 36th Annual ACM Symposium on Applied Computing (ACM SAC); rank: B; pages 18-25; Virtual Event Republic of Korea; 22-26 March 2021 [Reshma et al., 2021].
   Access: https://dl.acm.org/doi/pdf/10.1145/3412841.3441884.

4. **Ismat Ara Reshma**, Camille Franchet, Margot Gaspard, Radu Tudor Ionescu, Josiane Mothe, Sylvain Cussat-Blanc, Hervé Luga, and Pierre Brousset. Finding a Suitable Class Distribution for Building Cancer Histological Images Data Sets Used in Deep Model Training. In: Journal of Digital Imaging (JDI); IF: 4.056 (2020); publisher: Springer. (Submitted on 23th of December 2020: waiting for the first decision)

Chapter 1
Introduction

Cancer is one of the deadly diseases and considered as the second leading cause of death globally [Deniz et al., 2018; Cortes et al., 2020]. More than 100 types of cancer, such as, breast, skin, lung, colon, prostate, and lymphoma, are identified and categorized according to the tissue of origin [Deniz et al., 2018; Grixti and Ayers, 2020]. Once a type of cancers affects the human body, it can spread through the blood or lymphatic system and can damage other organs of the human body [Padera et al., 2016; Pang et al., 2016; Deniz et al., 2018]. According to the International Agency for Research on Cancer, GLOBOCAN cancer statistics[1] for the year 2020, there were an estimated 19.3 million new cases of cancer (18.1 million excluding non-melanoma skin cancer) and almost 10.0 million deaths from cancer (9.9 million excluding non-melanoma skin cancer) worldwide [Ferlay et al., 2021]. Any advances in diagnosis and detection of cancer are thus crucial for saving lives.

If any suspected case is found, a doctor must diagnose whether it is due to cancer or any other reason. The diagnosis process includes analysis of personal and family medical history, lab tests (e.g., blood, urine, or other body fluids test), imaging tests (e.g., X-ray, ultrasound, computed tomography, nuclear medicine including Positron Emission Tomography (PET), fluoroscopy, and Magnetic Resonance Imaging (MRI)), and biopsy[2] [Schiffman et al., 2015]. Among all the tests, the most common method for cancer diagnosis is based on tissue biopsy[3] [Li et al., 2021b]. The tissue biopsy or simply biopsy is a procedure in which a sample of tissue is extracted from the suspected area, and a pathologist looks at the tissue through a microscope by placing it on a glass-slide after some pre-processing and runs other tests to see if the tissue is cancer[2]. The process of examining the tissue to study the manifestation of the disease is called histological analysis [Slaoui and Fiette, 2011; He et al., 2012; Kandel and Castelli, 2020]. It plays an important role in cancer diagnosis and

[1] https://gco.iarc.fr/today/home

[2] https://www.cancer.gov

[3] https://training.seer.cancer.gov/disease/diagnosis/

staging [Gupta and Madoff, 2007; He et al., 2012].

About a decade ago, histological analysis of the glass-slide of a tissue sample with a microscope was the primary diagnosis method for cancer [Barisoni et al., 2020]. The digitization of the glass-slide of the tissue sample, which was introduced in 1999 after the invention of the whole slide scanner [Bera et al., 2019], was considered as a fairly novel technology and of insufficient maturity for primary diagnostic purposes [Barisoni et al., 2020]. The digitized version is called the histological image, which is also known as Whole Slide Image (WSI) or virtual slide [Naylor et al., 2017; Barisoni et al., 2020; Otálora et al., 2021]. The whole process of creating a WSI is given in the Figure 1.1. The biopsied tissue sample is placed on a glass-slide after doing some preprocessing, including fixation, dehydration, cleaning, infiltration, embedding, sectioning, staining. The processed glass-slide of the tissue sample is then scanned with a whole slide scanner to produce a WSI [He et al., 2010].

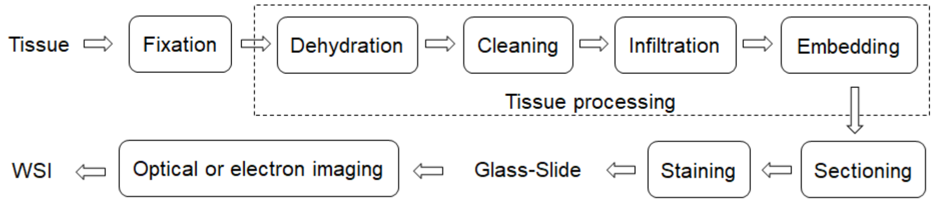


Figure 1.1: Histological tissue preparation and image production in tissue biopsy (Fig. 2 from [He et al., 2010]).

At the beginning of WSI technology, WSI was limited for educational uses, and its role in clinical practice was not apparent given the preference of conventionally trained pathologists to work with the conventional glass-slide of a tissue sample under microscope [Barisoni et al., 2020]. The status, however, has been changed by the emergence of rapidly evolving technologies in combination with developments in computational image analysis [Naylor et al., 2017; Barisoni et al., 2020; Otálora et al., 2021]. This new approach is being increasingly adopted in clinical trials [Barisoni et al., 2012] and for clinical research [Wang et al., 2016a; Liu et al., 2017, 2019; Lin et al., 2018; Veeling et al., 2018; Fan et al., 2019a], and with the approval of the FDA (US Food and Drug Administration)[4], it has even been using as a tool for primary diagnosis from 2017 [Evans et al., 2018; Barisoni et al., 2020].

[4] https://www.fda.gov/news-events/press-announcements/fda-allows-marketing-first-whole-slide-imaging-system-digital-pathology

WSIs, however, have some limitations: they are very large, usually gigapixels; their appearance can be different in color intensities depending on the staining process, operator ability, and scanner specifications [Komura and Ishikawa, 2018; Dimitriou et al., 2019; Salvi et al., 2020]. Moreover, through the collaboration with practicing pathologists, we come to know that sometimes more than 30 WSIs are required to be analyzed by pathologists to diagnose cancer on one patient. Analyzing this number of gigantic-sized WSIs without any assistance is time-consuming, laborious, and prone to error or misinterpretation. Additionally, it depends heavily on the expertise, experience, and fatigue level of pathologists, being difficult to grade in a reproducible manner, and empirically, it is known that there are substantial intra- and inter-observation variations among experts [He et al., 2012]. To assist the pathologists in this regard, computer-aided diagnosis (CAD) techniques can be added to the pathologists' workflow [Antropova et al., 2017; Liu et al., 2019]. Machine learning techniques, both classical and deep learning, are utilized in developing CAD systems [Gurcan et al., 2009; Bejnordi et al., 2017; Litjens et al., 2017; Hu et al., 2018; Komura and Ishikawa, 2018]. The deep learning algorithms, e.g., Convolutional Neural Networks (CNNs), are the current state-of-the-art methods [Otálora et al., 2021] in this case following their incredible success in visual recognition tasks [Krizhevsky et al., 2012a; LeCun et al., 2015; Rawat and Wang, 2017].

The CAD systems with CNNs for WSI, however, face some difficulties as well, e.g., the heterogeneity and high complexity of tissue area, the inter-class similarities and intra-class difference, the fact that advanced image processing systems for other imaging applications (such as, computed tomography, MRI, X-ray) cannot be directly adapted to the WSI due to the different imaging characteristics, and the scarcity of enough ground truth cancerous WSIs. All these difficulties makes the algorithm evaluation largely subjective or dependable to the minimal confidence testing [He et al., 2012; Komura and Ishikawa, 2018; Otálora et al., 2021]. The success of CNN models also depends on several hyper-parameter settings [Hinz et al., 2018; Bacanin et al., 2020], e.g., network architecture, data used to train the model, the class distribution of the training data. To address the increasing demand for experts' time to interpret WSIs for cancer diagnosis, the CAD system can play a vital role. It can provide rapid and consistent results compared to the manual system [He et al., 2012]. With the advances in scanning technology paired with the increasing ability of computing resources, pathologists in a collaboration with technology experts, including data scientists, computational engineers, imaging physicists, explore the potentiality of converting the pathology specialty to fully digital workflows and clinically applicable CADs [Barisoni et al., 2020].

To our knowledge, among the aforementioned hyper-parameters regulating the performance of CNN models, the class distribution of the training data is not studied yet in the literature for the WSI data -which is imbalanced usually [Wang et al., 2016a; Lin et al., 2018]. Class distribution is one of the important hyper-parameters, as the training data provides the supervision for all learning-based systems [Cracknell and Reading, 2014; Crawford, 2016; Deisenroth et al., 2020; Sarker, 2021].

Generally, in machine learning an imbalanced data distribution has been shown as problematic [Chawla et al., 2002; Khan et al., 2017], and hence, a lot of efforts have been put in to develop methods to overcome the problem. Buda et al. [2018] have reviewed the popular methods, such as oversampling, undersampling, thresholding, cost-sensitive learning, one-class classification, and various hybrids. These studies may indeed lay the path for balanced distribution to become the default choice as a deep learning state-of-the-art method [Bejnordi et al., 2017; Liu et al., 2017; Halicek et al., 2019], although Prati et al. [2015], for example, have shown that it is not optimal in all cases. A lot of studies make the imbalance data balanced, but, very few analytically consider the performance impact of different distributions [Weiss and Provost, 2001, 2003; Prati et al., 2015]. The available studies have been mainly conducted on toy data sets, even though real data sets may be very different and more complex. Thus, there is no evidence from the conclusions of these studies that they would be appropriate for cancer WSIs.

The outcomes of available studies are contradictory: some support an imbalanced distribution [Weiss and Provost, 2001, 2003] while others support a balanced distribution [Prati et al., 2015]. It is thus not straightforward to decide on a specific class distribution for all types of tasks. We believe that elaborate domain-specific analysis is required for each specialized task, especially for a comparatively new and sensitive case, such as cancer detection from WSIs. It is also worth knowing which distribution produces fewer false positives with high sensitivity and why. It would help in choosing training examples and their ratios for building robust training data sets and utilizing the existing ones optimally; this is specifically crucial because of the cost and expertise needed to annotate WSI.

Considering the evaluation of the models, we found that the state-of-the-art evaluation methods (e.g., the Receiver Operating Characteristics, ROC Curve) have some limitations, e.g, the inability of model performance separation, not being much informative regarding false predictions, being optimistic to imbalanced data. This thesis also contributes to this topic.

The contributions resulting from this thesis are as follows.

- To conduct an analytical study on different class distributions or a comparative study on different evaluation metrics, we require a general methodological framework to generate predictive models. We thus develop a methodological framework consisting of four major steps, including pre-processing of the data, training of models, inferring class probabilities of unseen data samples or test set, and evaluating models. In the pre-processing steps, we propose a patch (i.e., image-block/region) extraction and categorization method to address the size problem of WSIs. It facilitates coverage of all tissue areas without repetition and the creation of different class distributions in data for training a model. Moreover, in the evaluation step, we propose to use a patch-based (i.e., image-block-based) evaluation methodology rather than the usual pixel-based one. Indeed it corresponds more to pathologists' evaluation where they look into image blocks (i.e., patches) rather than pixels. The framework is applicable for both image segmentation[5] and classification[6] task.

- We investigate the effectiveness of the default choice, the balanced distribution, in cancer detection from WSI. More precisely, we addressed two research questions: *is the most adopted balanced distribution optimal for the cancer detection from WSI? If not, which class the training set should be biased toward?*

  To answer the questions, we conduct a preliminary study on the model performances when trained with different distributions including the natural (i.e., the original distribution of the training set), cancer-biased, non-cancer-biased, and balanced distributions. We design and run several experiments with the mentioned class distributions. We found that the balanced distribution is not optimal for CNN training for cancer

[5] The task of assigning a class label to each pixel of the input image.

[6] The task of assigning class label(s) to the entire input image.

detection. Rather using natural distribution leads to the best recall-precision trade-off; when training with class-biased distribution, it is possible to inflect the model toward the desired accuracy (e.g., toward recall or precision). Moreover, we found that most of the errors in this task come from False-Positive (FP) -the number of samples that are falsely predicted as positive/cancer- rather than False-Negative (FN) -the number of samples that are falsely predicted as negative/non-cancer-, and reducing FP is more difficult than reducing the FN. In the cancer detection task, pathologists' goal is to reduce FN; our finding will be useful to reset the goal of this task since FN reduction is already easy, while FP reduction is not. This work was presented in a paper accepted by the International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI 2019) [Reshma et al., 2019].

Moreover, we compute the evaluation with patch dimension from 1×1 pixel (i.e., pixel-level) to 1000×1000 pixels to observe the result at different levels of patch dimension. We found that recall increases while precision decreases when the patch dimension increases.

This preliminary study has some limitations, e.g., not all the experiments have the equal number of training examples, which hampers the fair comparison. We thus formalize the analysis in our next deeper analysis.

We present a hypothesis-driven deeper analysis, which determines the performance impact of different class distributions on training data. We derive several hypotheses with regard to WSIs used for cancer detection and test them with two commonly used target applications: image segmentation and classification. We employ two data sets for training: one is multi-class (annotated with cancer, non-cancer, and other histological structures), and is applicable to the segmentation setting, whereas the other is binary-class (annotated with cancer and non-cancer classes), and is applicable to the classification setting. With both data sets, we conduct a series of experiments and analyze the results in detail to be able to provide comprehensive conclusions. Additionally, we also discuss the case of test data distribution. To this end, we use the test set from an additional data set along with the two previous data sets.

Our experimental results are consistent with our preliminary findings. Moreover, we observe several interesting findings regarding different

WSI regions, test set distribution, and classification types. Our approach was presented in a paper accepted by the 36th Annual ACM Symposium on Applied Computing (ACM SAC) [Reshma et al., 2021]. An extended version was submitted to the Journal of Digital Imaging (JDI) on 23 December 2020; we are waiting for the first review.

- Another contribution of this thesis is developing an evaluation metric that addresses the limitations of the state-of-the-art metrics -the Receiver Operating Characteristics (ROC) curve and Precision-Recall (PR) curve- popularly utilized in binary classification (e.g., cancer detection task).

  In an imbalanced data set, the outnumbered class is usually easy to classify, which causes inflation of the ROC curve [Fu et al., 2017; Saito and Rehmsmeier, 2017; Sofaer et al., 2019; Cook and Ramadas, 2020]. Because of this inflation, the ROC curve can hardly distinguish the performances of two different models. We show that for a large test set, the PR curve -which is usually used as an alternative to the ROC curve for imbalance data [Brodersen et al., 2010; Keilwagen et al., 2014; Ozenne et al., 2015; Fu et al., 2017; Sofaer et al., 2019]- have the same problem. The first challenge in binary classification evaluation is thus *to define a metric that is not falsely inflated and cannot mislead about the actual performance of the model due to data imbalance.* More importantly, the measure should also be able to separate models that differ from each other.

  Some applications need to minimize false negatives (e.g., medical diagnosis) while others need to minimize false positives (e.g., fraud detection). From ROC and PR curves, especially from inflated ones, it is difficult to know the type of false prediction the model produces the most. The second challenge we address is thus *to develop an evaluation metric that allows the model designer to analyze properly the model errors.*

  PR curves also face the difficult problem of interpolation, which is however important during mean curve and the Area Under Curve (AUC) computations. While it is straightforward for the ROC because FPR and TPR are linearly correlated, it is not for the PR curve since precision does not monotonically change with recall. Therefore, it requires non-linear interpolation. Existing methods to non-linearly interpolate a point on the PR space [Davis and Goadrich, 2006; Boyd et al., 2013; Keilwagen et al., 2014] are not enough to compute the mean PR curve (and PR-AUC) since the start-, intermediate-, and end-point with the tied x-axis values need to be treated as well. The third challenge we address is thus *to propose an evaluation metric that neither requires interpolation nor needs to handle any cases specially.*

With the aim to target the three challenges above-presented, we present a novel metric, named the PR-T curves, which present precision and recall in two separate curves as functions of the threshold used on the predicted scores. Both theoretically and experimentally, we show that PR-T curves eliminate the above-mentioned limitations of ROC and PR curves. Moreover, by considering the third challenge in the state-of-the-art metric, the difficulty in mean curve computation of the PR curve, we propose new methods of mean PR curve and PR-AUC computations by handling all special cases. Additionally, we develop a method to non-linearly interpolate a point on the PR space, which is easy to implement and computation-friendly.

At the time of writing this manuscript, we are preparing an article on the contribution of proposing the PR-T curves to submit to a journal and another article on the contribution of proposing algorithms for mean PR curve and PR-AUC to submit to a conference.

The structure of this thesis is as follows:

Chapter 1 is this introduction in which the research questions and main contributions have been presented.

Chapter 2 presents the basic context of this research to familiarize the reader with general terminologies. Specifically, we discuss the WSI -the data to be analyzed- in more detail, task description, the techniques of machine learning we utilized, data sets, and evaluation metrics popularly utilized in the cancer detection task.

Chapter 3 reviews the literature on cancer researches utilizing medical images (both radiology and histology images), image processing, and machine learning techniques. It also describes our general methodological framework and evaluation technique for cancer detection in WSI, which is utilized throughout this thesis.

Chapter 4 discusses the problem of finding the optimal class distribution of a training set to train an optimal model that detects cancer in WSIs. At first, we conduct a preliminary study. Based on the findings from that study, we formulate several hypotheses, which are then tested and discussed based on the test result.

Chapter 5 proposes a continuous threshold-based evaluation metrics to evaluate model on binary classification problem (e.g., cancer detection) by addressing limitations of state-of-the-art metrics, ROC and PR curves. In addition, it proposes a method to correctly compute the mean PR curve and PR-AUC since, in literature, there are no precise algorithms to compute them by handling all special cases that could happen with the PR curve.

Chapter 6 concludes this thesis, discusses the main contributions and limitations of our work, and proposes some future directions.

Chapter 2
Context

Contents

To enhance the readability of the thesis, it requires describing its materials and related contexts. This chapter serves that purpose. We include the technical details of elements of this thesis, including the main data (i.e., the WSI), the task description (i.e., the cancer detection), the techniques of machine learning we utilized, the data sets, and evaluation metrics with a short survey about each of them. Along with the motivation presented in the introduction chapter, the surveys help us deciding the materials or elements for our task of interest. The description will help the reader to familiarize themselves with the general terminologies of this thesis. The included contents, however, are limited. We assume that the readers are already familiar with general image processing and machine learning, specifically deep learning techniques.

The structure of this chapter is as follows: In Section 2.1, we detail the WSI, including its acquisition, manipulation, regions, and pixel class distribution. Section 2.2 defines our task of interest cancer detection and its research dimensions. Usually, machine learning techniques are utilized in cancer research that we define and describe in Section 2.3. We present a mini-survey on one of the important elements of a machine learning system, the data set utilized in cancer detection from WSIs in Section 2.4 and also present the descriptions of our selected data sets in this thesis in the same section. Section 2.5 presents another survey on evaluation measures, which is another important element of a machine learning system. The same section also defines our selected evaluation measures based on the survey and the purpose of the thesis. Finally, Section 2.6 concludes the chapter.

2.1 Whole Slide Image (WSI)

WSI is a special kind of image data. To understand and analyze this data, it requires to know about its technical details. Moreover, it requires to know about its different regions and distribution of pixels to utilize these data to solve a problem, e.g., cancer detection, automatically. This section discusses these two topics.

2.1.1 Technical detail: acquisition and manipulation of WSI by pathologists

Whole slide images (WSIs) are the digital conversion of conventional histological glass slides containing tissue samples [Farahani et al., 2015; Kumar

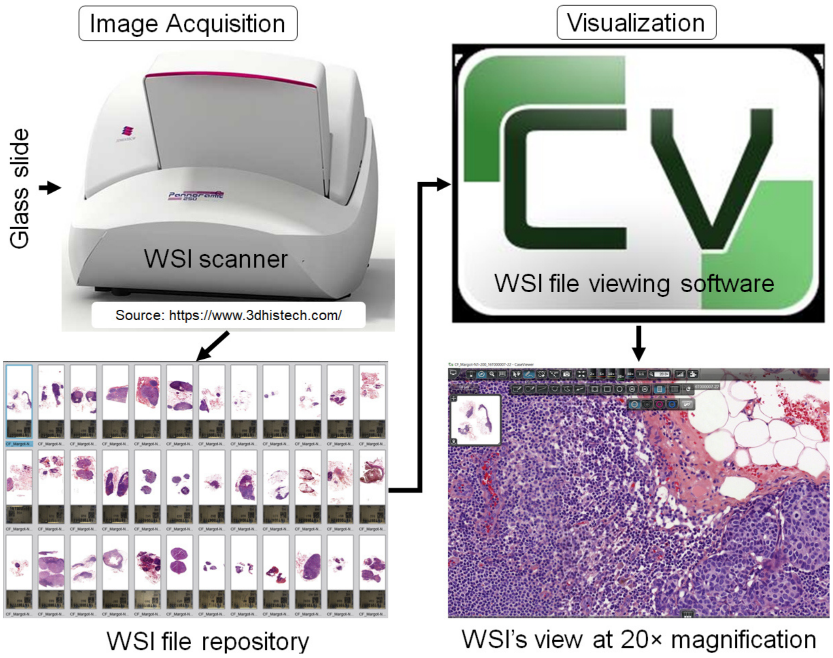


Image Acquisition
Visualization
Glass slide
WSI scanner
Source: https://www.3dhistech.com/
WSI file viewing software
WSI file repository
WSI's view at 20× magnification

Figure 2.1: **A schematic view of analyzing WSI by a pathologist**. It consists of two steps: image acquisition (left part) and visualization (right part). The WSI file is acquired by a WSI scanner from a conventional glass slide, which is then stored in a repository. A WSI file viewing software is used to visualize or manipulate the file. An example software is CaseViewer.

et al., 2020]. WSIs can be generated by taking biopsied tissue samples from the various suspected anatomical site of the body depending on the purpose of examination. The lymph node WSIs, for example, are examined to see if the cancer is metastasized [1] or not; it helps to decide the stage and treatment of a variety of cancers [Liu et al., 2019]. In general, analysis of WSIs by a pathologist includes two processes: image acquisition and visualization [Melo et al., 2020; Kumar et al., 2020] (Figure 2.1).

For the image acquisition (Figure 2.1: left part), specialized hardware known as digital slide scanner or WSI scanner[2] is required along with some pre-processing (as shown in Figure 1.1). Modern scanners can obtain WSIs

[1] Metastasized cancer or metastatic cancer or simply metastasis is the spread of malignant cells from its primary tumor site to distant sites, which poses the biggest problem to cancer treatment and is the main cause of death of cancer patients [Geiger and Peeper, 2009].

[2] A WSI scanner is a trinocular microscope with robotic control of illumination intensity, mechanical stage, objectives, and coarse and fine focusing facilities and is equipped with a high-resolution camera connected to a computer [Bueno et al., 2014].

from slides stained with hematoxylin & eosin (H&E), special stains (such as, Acid Fast Bacilli, Alcian Blue, Periodic Acid Schiff, Gram, Congo Red, and Toluidine Blue), immunohistochemistry stains, and fluorescence stains [Aeffner et al., 2018]. WSIs can vary depending on their acquisition process on a scanner. There are several different manufacturers with varying designs of WSI scanners (e.g., 3D Histech, Leica, Hamamatsu, Olympus, TissueGnostics, and Sakura). These scanners have various magnifications, scanning methodologies, hardware, and software employed to convert a glass slide into a WSI [Indu et al., 2016; Abels and Pantanowitz, 2017; Kumar et al., 2020].

Table 2.1: **File size of WSI change with the change of magnification**: data is collected from CaseViewer software for different magnification levels of an example WSI, which was originally scanned at magnification $20\times$ with a 3DHISTECH Pannoramic 250 digital slide scanner.

Magnification	Dimension in pixel	File size
$20\times$	83200×123136	2.52 GB
$10\times$	41728×61696	1.06 GB
$5\times$	20992×30976	355 MB

Unlike the still microscopic images, WSI scanners capture sequential images either in a tiled or line-scanning manner which are subsequently assembled or stitched into a virtual slide that mimics the glass slide under a microscope [Indu et al., 2016; Aeffner et al., 2018; Kumar et al., 2020]. Apart from the x- and y-axes, some scanners capture several scans along the z-axis and assemble them on top of each other ("z stacking") to allow digital fine focusing on an area of interest [Aeffner et al., 2018; Kumar et al., 2020]. Z-stack is identified as the nominal physical height (in μm) of image focus above the glass slide, which is used for relative spatial positioning of image planes[3] [Garcia-Rojo et al., 2016]. Independent of the scanning and focusing methodology, most scanners allow for scanning at multiple magnifications. Most commonly, this includes $20\times$ (typically at resolution of 0.5 $\mu m/pixel$) and $40\times$ (typically at resolution of 0.25 $\mu m/pixel$) magnification. Scans at 20× magnification are sufficient for standard viewing and interpretation, while higher magnification, e.g., $40\times$, is required for the cytological analysis [Wright et al., 2013; Aeffner et al., 2018; Kumar et al., 2020]. Note that

[3] http://dicom.nema.org/Dicom/DICOMWSI/

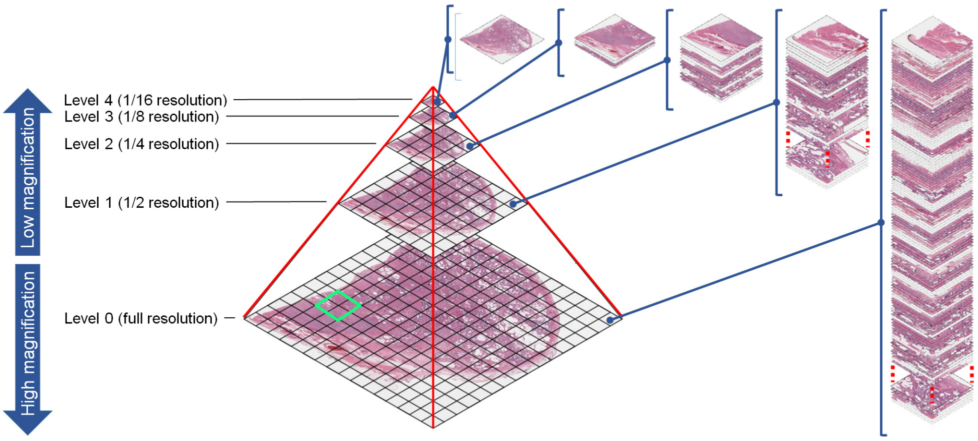


Figure 2.2: **A “pyramid" structure of WSI file with a thumbnail on top and high resolution on the bottom**. In this example, 5 levels of resolutions are illustrated. The highest magnification is considered as level 0, while the lowest magnification is considered as level 4, which is 16 times downsampled from level 0. Each level is composed of tiles. (Figure taken and modified from [McClintock, 2018])

the higher the resolution used for scanning, the larger the data file created for each slide, which then needs much storage to be archived and stored. Doubling the magnification from a 20× scan to 40× increases the file size by approximately 2 times (see Table 2.1). A WSI with the highest resolution can be several gigapixels in size.

Unlike a normal image, a WSI is stored in a pyramid structured file containing multiple digital slides of multiple magnifications or resolutions (Figure 2.2). Each level of resolution is stored in a separate “page” of the WSI and composed of a set of tiles (Figure 2.2: right part). Level 0 or page 0 contains the image at the full resolution. Other levels contain downsampled images with lower resolutions. The factor between two consecutive levels of resolution is traditionally of 2 (see Figure 2.2: left part), but it may vary. Depending on the scanner specification, the pyramidal WSI file can contain 7 to 10 levels of resolution. The pyramidal file structure allows the extraction of the WSI image at any available resolution and takes low to high magnification levels into account. Libraries such as openslide or libvips allow to process such WSI formats and provide methods to directly access the downsampled levels, optimizing the processing time spent on the WSI. Moreover, this pyramidal structure gives a high level of flexibility for the user to zoom and pan in the image, and so mimic the behavior of slide review under a microscope [Guet, 2021]. For example, when a WSI viewing software (Figure 2.1: right part) is connected to the WSI repository, the software requests a particular part of the WSI file (for example, the green marked area in Figure 2.2) at a particular zoom level; the repository then sends only the tiles needed to fulfill the request instead of sending the full file, which allows fast viewing and saves memory [McClintock, 2018].

For the visualization (Figure 2.1: right part), a specialized digital slide viewing software (e.g., Surface slide, Aperio ImageScope, PathXL, Definiens, CaseViewer) is required [Webster and Dunstan, 2014; Melo et al., 2020]. This kind of software is used to mimic the process of viewing a glass slide under a microscope digitally [Abels and Pantanowitz, 2017]. Due to the large amount of information on a WSI, pathologists cannot view an entire sample at high resolution. Instead, they pan through the slide with a viewing software at a relatively low resolution -typically $5\mu m/pixel$ ($2\times$ magnification) or $2.5\mu m/pixel$ ($4\times$ magnification)- and then zoom in to higher resolution for selected regions of diagnostic interest in a similar manner microscopists examine a sample under a microscope[3]. When multiple Z-stacks are captured, viewers also provide a rapid change of Z-stack selection. Viewing softwares often able to annotate the image and export it to other file formats [Kumar et al., 2020].

2.1.2 Regions and pixel class distributions in WSIs

In the previous section, we discuss how a pathologist utilizes WSIs manually through viewing software. In this section, we discuss the regions and class distribution in WSIs, which is beneficial to know to develop a CAD with machine learning algorithms. While applying machine learning, two regions in a WSI are considered:

- Regions of interest (ROIs) are the regions that alert pathologists to check for abnormalities. In the case of lymph node WSIs, for example, the regions containing lymph nodes are ROIs, since the health of lymph node tissue is what pathologists observe. The ROIs can in turn be divided into two classes: positive and negative. Metastasis is considered as positive class, *cancer* (denoted by $\mathbb{C}$ in the rest of the manuscript), while any remaining ROIs are considered as negative class, *non-cancer* (denoted by $\neg\mathbb{C}$).

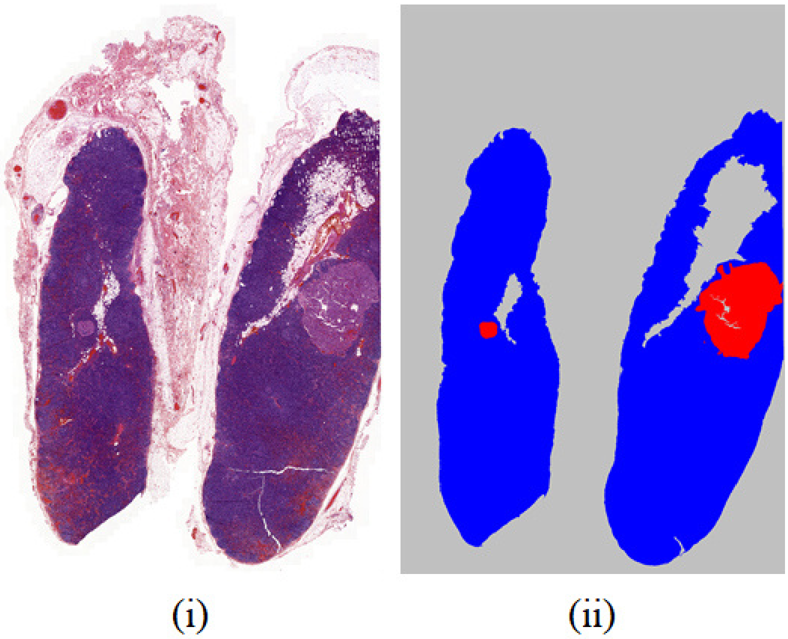


Figure 2.3: An example (i) metastatic lymph node WSI and (ii) its annotation. In (ii), the color red represents the cancer class ($\mathbb{C}$), blue represents the non-cancer class ($\neg\mathbb{C}$), and gray represents *other* class, mainly the background ($\mathbb{O}$). Both $\mathbb{C}$ and $\neg\mathbb{C}$ are ROIs, while $\mathbb{O}$ is not.

- Other regions (non-ROIs) are mainly background and histological structures other than lymph node tissue. The non-ROIs, that is to say, non-lymph nodes, are considered as belonging to the negative class *other* (denoted by $\mathbb{O}$).

In other words, three classes are usually considered in the utilization of a WSI: the positive ROI class, *cancer* ($\mathbb{C}$), the negative ROI class, *non-cancer* ($\neg\mathbb{C}$), and the negative non-ROI class, *other* ($\mathbb{O}$). For the binary classification, both $\mathbb{O}$ and $\neg\mathbb{C}$ are merged into one class and simply considered as being the negative class $\neg\mathbb{C}$.

Figure 2.3 shows an example of a metastatic lymph node WSI with its corresponding regions. In a WSI data set, the non-ROI class is usually overrepresented, while the two ROI classes are balanced or imbalanced, depending on which WSIs have been included in the data set (more details are provided in Section 2.4). On average, a WSI contains 70 to 80% of non-ROI pixels [Wang et al., 2016a; Lin et al., 2018]. We consider this usual bias towards non-ROIs as the *natural* distribution.

2.2 Cancer detection

Cancer, also called malignancy and neoplasms, is an abnormal growth of cells in a multistage process that generally progresses from a pre-cancerous lesion to a malignant tumor, which can then invade adjoining parts of the body and spread to other organs [Coleman and Tsongalis, 2010]. Any checking for abnormalities in cells that might be cancer or might become cancer in the future is cancer detection. As a life-threatening disease, cancer gets the attention of many researchers, including computer scientists, to investigate it. For the research purpose, different modalities of data have been using, including texts, images, or tissue samples of the confirmed or probable affected area for cancer detection and other related fields [Schiffman et al., 2015].

The research with imaging modalities (e.g., computed tomography, MRI, X-ray, WSI) is a popular track for cancer research in the computer science community. Among them, the researches with radiology[4] images have become more advanced than that of the histology images [Barisoni et al., 2020]. The research outcomes on the radiology images reach such a height that some developers have sought commercialization of their models [Yu et al., 2020]. The radiology imaging tests are particularly important to identify cancer in its earliest stage when patients not experiencing any symptoms, look for a lump or tumor inside the body, predict if the tumor is potentially cancerous, decide on if a biopsy is needed, guide the biopsy needle, see if cancer spreads to other body parts, plan treatment, check if the treatment working, and identify if cancer returned after the treatment[5]. Numerous systems and products have been developed with these image modalities [McAuliffe et al., 2001; Ibanez et al., 2003; Schneider et al., 2012; Roth et al., 2015; Shen et al., 2015; Lian et al., 2016; Sun et al., 2017b,a; McQuin et al., 2018; Zhou et al., 2021]. These systems, however, are not directly applicable to histology images (i.e., WSIs) because of the huge difference in image characteristics [He et al., 2012].

For precision oncology, however, the analysis of WSI from biopsy plays a key role [Djuric et al., 2017], which is comparatively newer and fewer than the radiology one in computational medical research [Barisoni et al., 2020]. In the recent decade, with the advances of imaging technologies, huge storage

[4] The study of high-energy radiation used to examine and diagnose internal structures of the body [Christensen et al., 2019]

[5] https://www.cancer.org/treatment/understanding-your-diagnosis/tests/imaging-radiology-tests-for-cancer.html

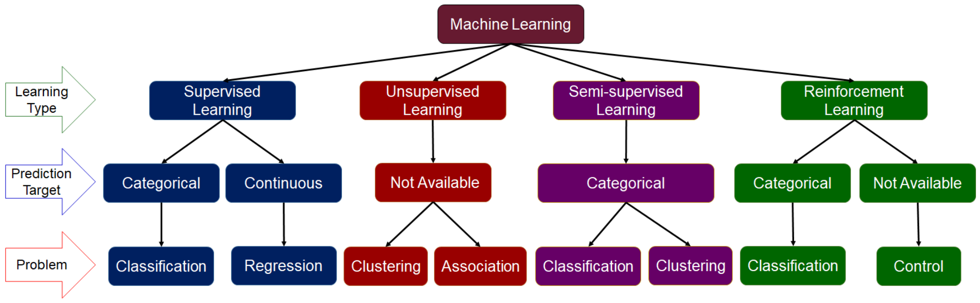


Figure 2.4: Various machine learning types, their prediction target types, and applicable problems to solve. (Figure is taken and modified from [Sarker, 2021]).

devices, and high-performance computing devices, the exploration of WSIs for cancer research is becoming popular. Like other images, the machine learning algorithms, both classical [DiFranco et al., 2011; Chang et al., 2013a,b; Sharma et al., 2015; Vu et al., 2015; Lu and Mandal, 2015] and deep learning [Wang et al., 2016a; Liu et al., 2017, 2019; Lin et al., 2018; Veeling et al., 2018; Fan et al., 2019a; Han et al., 2020; Alzubaidi et al., 2020; Otálora et al., 2021; Li et al., 2021a] are used either to classify or segment the WSIs. Deep learning algorithms produce state-of-the-art results [Otálora et al., 2021].

2.3 Machine learning

Machine learning is the field of study in computer science, specifically in Artificial Intelligence (AI), that deals with the automated detection of meaningful patterns in data [Shalev-Shwartz and Ben-David, 2014]. It is one of the fastest-growing areas of computer science, which is the foundation of countless important applications, e.g., web search, email anti-spam, speech recognition, product recommendations, and more, that require information extraction from large data sets [Shalev-Shwartz and Ben-David, 2014; Ng, 2019]. It is also widely used in other scientific fields, such as bioinformatics, medicine, and astronomy.

What is a machine learning algorithm? According to Goodfellow et al. [2016], it is an algorithm that is able to learn from data. To explain “learning" in this definition, Mitchell et al. [1997] provides a brief definition: “A computer program is said to learn from experience **E** with respect to some class of tasks **T** and performance measure **P**, if its performance at tasks in **T**, as measured by **P**, improves with experience **E**." The algorithm gathers experience from the data set for a particular task (e.g., classification) by tuning its parameters -which is known as training-, and its performance is evaluated on an unseen set of data -which is known as validation and test sets- by a performance measure (e.g., accuracy, error rate). The experience **E**, task **T**, and performance measure **P** can be of wide variety [Goodfellow et al., 2016]; in this chapter, we discuss those that belong to our topic of interest.

Machine Learning algorithms can be mainly divided into four types according to the type of experience (**E**) they are allowed to have during the learning process: Supervised, Unsupervised, Semi-supervised, and Reinforcement learning [Sarker, 2021], which are used to solve various problems/tasks of classification, regression, clustering, association, and control (Figure 2.4). Sarker [2021] provides comprehensive definitions of all these types. In this thesis, we focus on the classification problem with supervised learning, where the algorithm gathers the experience from WSI data for the cancer detection task. This type is described in the following section.

2.3.1 Supervised learning: regression, classification, and segmentation

According to Liu [2011], supervised learning is a machine learning paradigm for acquiring the input-output relationship information of a system based on a given set of paired input-output training samples. As the output is regarded as the label of the input data or the supervision, an input-output training sample is also called labeled training data, or supervised data. The supervised learning problem can be further divided into two categories: regression and classification.

In the supervised learning paradigm, the goal is to infer a function, $f : \mathscr{X} \to \mathscr{Y}$ from a sample data or training set, $\mathscr{A}_n = \{(x_1, y_1), ..., (x_n, y_n) \in (\mathscr{X} \times \mathscr{Y})^n\}$. Typically, $\mathscr{X} \subset \mathbb{R}^d$, $\mathscr{Y} \subset \mathbb{R}$ for regression problems, and $\mathscr{Y} \subset \mathbb{N}_0$ for classification problem [Cunningham et al., 2008]. $x_i \in \mathscr{X}$ is a feature[6] set of dimension d, and $y_i \in \mathscr{Y}$ is usually either 1 or 0 for the binary classification and any natural number -where the number belongs to a class label- for multi-class classification.

[6] Each piece of information included in the representation of data is known as a feature [Goodfellow et al., 2016].

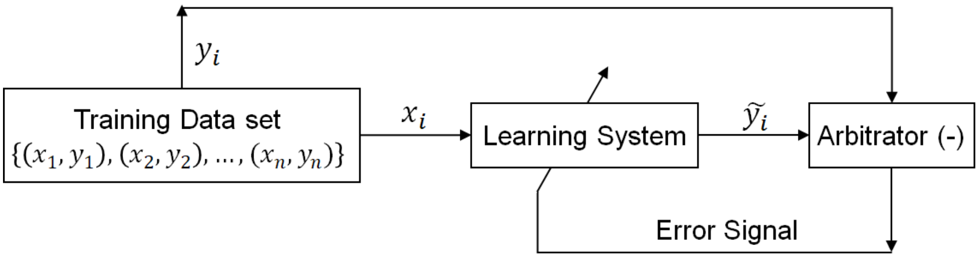


Figure 2.5: Block diagram of the supervised learning (Figure 1 from [Liu, 2011]).

Figure 2.5 shows a block diagram of supervised learning. During a supervised learning process, a training input x_i is fed to the learning system that generates an output $\widetilde{y}_i$. The learning system output $\widetilde{y}_i$ is then compared with the ground truth label y_i. The difference, termed error signal in this diagram, is then sent to the learning system for adjusting the parameters of the learner. The goal of this learning process is to obtain a set of optimal learning system parameters that can minimize the differences between $\widetilde{y}_i$ and y_i for all i. A minimum training error does not necessarily indicate good performance in testing [Liu, 2011]. The reason for this is mainly due to the possible overfitting to the training data. This issue is referred to as generalizability. A good learning algorithm must have a good generalizability [Liu, 2011; Vapnik, 2013].

A special type of classification problem is segmentation (although segmentation can also be achieved by clustering, discussion about this is out of the scope of this thesis) [Farmer and Jain, 2005; Dean, 2014]. It is the process of separating the data into distinct groups. A well-defined segment is one in which the members of the segment are similar to each other and also are different from members of other segments based on their features [Dean, 2014]. In the case of image processing -the topic of interest of this thesis-, the goal of segmentation is to segment predefined objects from the input image [Everingham et al., 2010; Noh et al., 2015; Khan et al., 2020] for the purposes of localization. This is performed by labeling each pixel of the corresponding object with a predefined color that corresponds to a class label. Image classification, on the other hand, is the process of assigning a class label (or labels for a multi-class problem) to the entire input image (Figure 2.6).

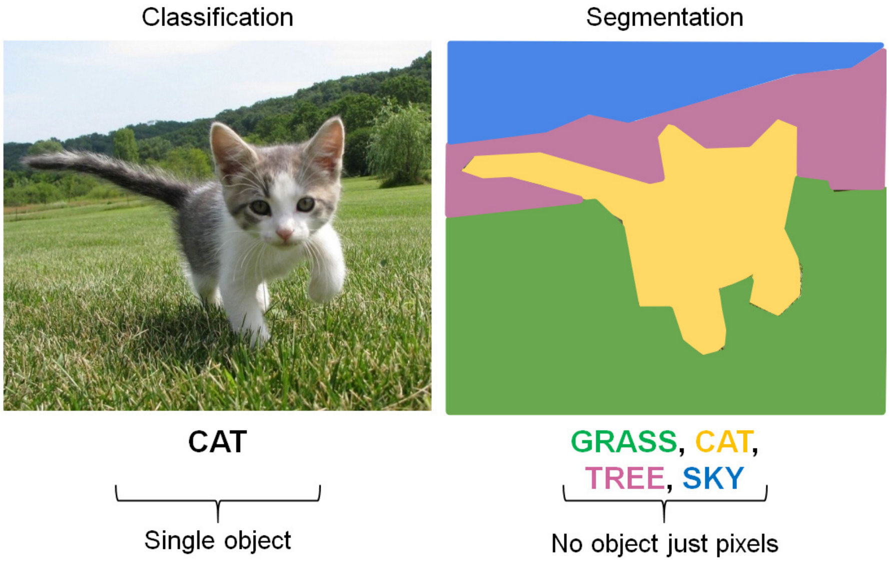


Figure 2.6: Difference between classification and segmentation tasks. In classification whole image belongs to a label (single-label problem) or multiple labels (multi-label problem), while in segmentation, each pixel or region of the image belongs to a label.

Source (modified): Detection and Segmentation
`http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf`

2.3.2 Types of machine learning algorithms based on learning strategy

Algorithms to solve machine learning problems can also be grouped into two types based on their learning strategies: traditional/classical and deep learning. They mainly differ from each other in terms of the representation of the data (i.e., feature set) they are given to tune their parameters during training.

2.3.2.1 Traditional/classical learning

The traditional learning algorithm is the one that needs to break down the problem statements into two parts: the features engineering by the domain expert and model selection or parameters tuning.

Some of the renowned traditional learning algorithms are naive bayes, linear discriminant analysis (LDA), linear regression, polynomial regression, LASSO and ridge regression, logistic regression, k-nearest neighbors (k-NN), support vector machine (SVM), decision tree, random forest, adaptive boosting (AdaBoost), extreme gradient boosting (XGBoost), k-means clustering, mean-shift clustering, gaussian mixture models (GMMs). All of the algorithms require pre-extracted hand-craft features from the data to train and test a model [O'Mahony et al., 2019]. Indeed, the efficiency of these algorithms highly relied on the goodness of the representation of the input data or features. A bad data representation often leads to lower performance compared to a good data representation. Therefore, feature engineering is an important research step in classical machine learning. It is often very domain-specific and requires significant human effort. It becomes more and more challenging and cumbersome as the number of classes to classify increases [Pouyanfar et al., 2018; Indolia et al., 2018; O'Mahony et al., 2019]. Numerous research studies have defined efficient feature descriptors. For the image processing or computer vision task, some of the traditional feature descriptors are histogram of oriented gradients(HOG), local binary pattern (LBP), scale invariant feature transform (SIFT), speeded-up robust features (SURF), and binary robust independent elementary features (BRIEF) [O'Mahony et al., 2019].

Moreover, selection of appropriate features for a given problem is also important for reducing computation time, improving prediction performance, and a better understanding of the data [Chandrashekar and Sahin, 2014; Khalid et al., 2014; Indolia et al., 2018; O'Mahony et al., 2019]. In feature selection process, a subset from the input features data are selected for the model selection or parameter tuning of the learning algorithm [Chandrashekar and Sahin, 2014]. The selected subset should efficiently describe the input data while reducing effects from noise or irrelevant variables and still provide good prediction results. The best subset is the one with the least number of dimensions that most contribute to learning accuracy [Ladha and Deepa, 2011; Khalid et al., 2014]. A lot of efforts are also given for devising feature selection and feature extraction techniques [Khalid et al., 2014; Chandrashekar and Sahin, 2014]; some worth mentioning approaches are minimal redundancy and maximal relevance, conditional mutual information maximization, correlation coefficient, between-within ratio, genetic algorithm, recursive feature elimination, principal component analysis, non-linear principal component analysis, independent component analysis, and correlation based feature selection. Khalid et al. [2014] and Chandrashekar and Sahin [2014] detail different feature selection and feature extraction techniques.

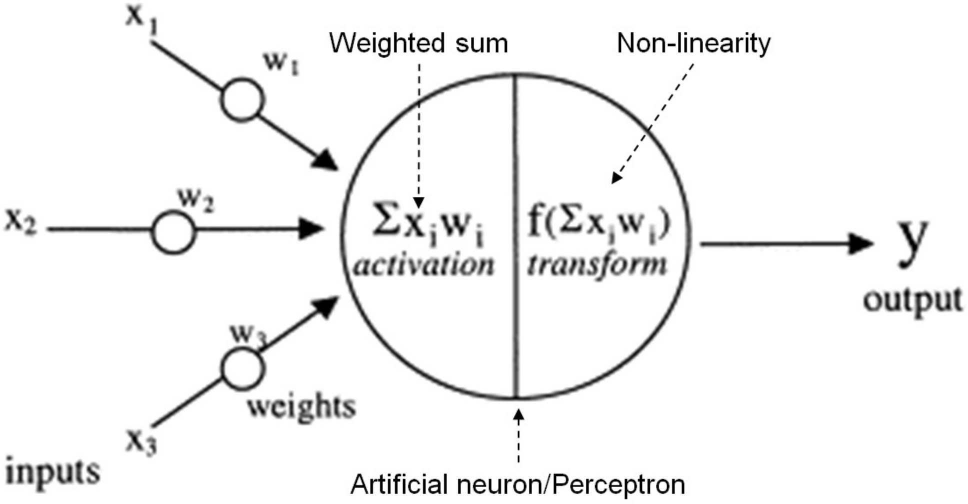


Figure 2.7: The structure of artificial neuron/perceptron used in the ANN [Agatonovic-Kustrin and Beresford, 2000].

At the beginning of the WSI technology, classical machine learning algorithms with feature engineering were mainly utilized [Gurcan et al., 2009; Komura and Ishikawa, 2018], although the trend changes very fast with the success of deep learning in the visual recognition task.

2.3.2.2 Deep learning

Deep learning is also known as deep structured learning and hierarchical learning that consists of multiple layers of abstraction which includes non-linear processing units for features transfer and extraction [Dargan et al., 2019]. The learning process takes place either in a supervised or unsupervised way [Dargan et al., 2019]. Unlike classical learning algorithms, these algorithms do not require to separate the feature engineering part from the parameter tuning part [Indolia et al., 2018]. The algorithms include both feature extraction and parameters tuning in end-to-end learning architectures. They were originally inspired by AI simulating the neurons of the human brain and nervous system. The brain can automatically extract data representation from different scenes and situations through the neurons and can process as per need. A deep learning algorithm mimics these criteria of the human brain thus works on the so-called artificial neural network system (ANN) [Pouyanfar et al., 2018].

The fundamental computational unit of ANN is an artificial neuron or perceptron that takes multiple signals as input then integrates these signals linearly with the weight -which represents the strength of interconnection between neurons of two adjacent layers- and transfers the combined signals over the nonlinear function named activation function to produce outputs [Dargan et al., 2019] (Figure 2.7).

A typical ANN has single input and output layers along with multiple hidden layers that can learn various features of data through multiple levels of abstraction [Pouyanfar et al., 2018; Indolia et al., 2018; Dargan et al., 2019]. Every subsequent layer takes the results from the previous layer as the input. Multiple layers naturally allow the network to integrate low/mid/high-level features and the "levels" of features can be enriched by the number of stacked layers (i.e., by depth) [Zeiler and Fergus, 2014; He et al., 2016a; Indolia et al., 2018]; the low-level features (e.g., edge) are extracted from the earlier layers of the networks, while more abstract features (e.g., shape) are extracted from the later layers [Pouyanfar et al., 2018; Liu et al., 2019]. Indeed, in "deep learning", the term "deep" comes from the concept of numerous layers through which the data is transformed [Dargan et al., 2019].

The deep learning algorithms are preferred to the classical ones since learned features from deep learning often produce much better performance than can be obtained with hand-craft features. This process of feature extraction (i.e., the one provided by the ANN system) is also rapidly adaptable to new tasks, with minimal human intervention [Hu et al., 2018]. It can discover a good set of features for a simple task in minutes, or for a complex task in hours to months; in contrast, manually designing features for a complex task requires a great deal of human time and effort, sometimes it take decades for an entire community of researchers [Goodfellow et al., 2016]. Although these algorithms are successful in many application domains, it is worth noting that there are also some disadvantages. Hu et al. [2018] enlisted some of them: deep learning models often require a large amount of training data for convincing performance, the training process is extremely computationally expensive and it is quite time-consuming to train a deep and complex model even with the support of the most powerful GPU hardware, the body of trained deep learning model is like a black box, we still lack the perfect methodology to fully comprehend its deep structure.

Some of the basic types of deep learning algorithms are: Auto-Encoder (AE), Convolutional Neural Network (CNN), Restricted Boltzmann Machine (RBM), Deep Stacking Network (DSN), Long Short Term Memory (LSTM)/

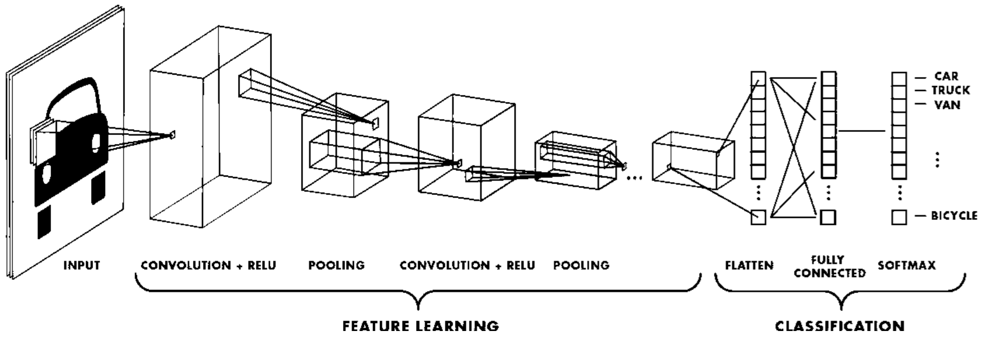


Figure 2.8: A general architecture of CNN. [Krizhevsky et al., 2012b].

Gated Recurrent Unit (GRU) Network, and Recurrent Neural Network (RNN). Out of these, CNN is the fundamental and the most commonly used approaches to solve complex problems [Pouyanfar et al., 2018; Indolia et al., 2018; Dargan et al., 2019]. CNNs are widely being used in various domains due to their remarkable performances, such as image classification, object detection, face detection, speech recognition, natural language processing, vehicle recognition, facial expression recognition, cancer detection, health care, and many more [Indolia et al., 2018; Dargan et al., 2019]. CNNs have three main advantages, namely, parameter sharing, sparse interactions, and equivalent representations [Pouyanfar et al., 2018]. A generic CNN architecture consists of a number of convolutional layers -which ensure parameter sharing- followed by activation functions (e.g., RELU) and pooling/subsampling layers -which ensure dimensionality reduction and sparse interaction-, and in the final stage, fully connected layers -which generate the high-level abstraction from the data- followed by an activation function, e.g., Softmax or SVM -which generate the classification scores [Pouyanfar et al., 2018] (Figure 2.8). If the fully connected layers in a CNN architecture are replaced with upsampling layers and deconvolutional layers, then it is called a Fully Convolutional Neural Network (FCNN), which is usually utilized for segmentation tasks. Instead of predicting one probability score to each class to classify the whole image/patch, FCNNs classify each pixel of the image/patch [Hu et al., 2018]. Pouyanfar et al. [2018], Indolia et al. [2018], and Tabian et al. [2019] provide detail descriptions of each components of CNNs. In this thesis, we utilize a CNN and an FCNN to classify patches and pixels of WSIs (describe in the next chapter).

2.4 Data sets

Learning algorithms gather experience (**E**) from data sets, as mentioned earlier. There are several data sets of WSIs available online. Those are usable for research purposes, where some of them require registration. Hu et al. [2018] and Komura and Ishikawa [2018] list up them. Most of the data sets are variable-sized patch-based (i.e., image region-based instead of full WSI), e.g., BreakHis[7] [Spanhol et al., 2016], KIMIA960[8] [Kumar et al., 2017], Biosegmentation[9] [Gelasca et al., 2008], Bioimaging challenge 2015[10] [Araújo et al., 2017], GlaS[11] [Sirinukunwattana et al., 2017], MITOS-ATYPIA-14[12], MITOS 2012[13] [Roux et al., 2013], and Patch Camelyon[14] (PCam) [Veeling et al., 2018]. A few data sets contain full WSIs with multi-resolution, e.g., CAMELYON16[15] [Litjens et al., 2018], CAMELYON17[16], The Cancer Genome Atlas (TCGA)[17] [Weinstein et al., 2013], Genotype-Tissue Expression (GTEx)[18] [Lonsdale et al., 2013]. There are also some data sets not available publicly due to different constraints on using private information of patients, e.g., in [Mejbri et al., 2019; Mejbri, 2019], authors proposed such type of data set for different types of breast cancer.

The data sets are designed for various kinds of tasks, e.g., disease classification, gland segmentation, mitosis detection, nuclear segmentation, and tumor detection. Some of them are multi-class, and some of them are binary class and specific to a particular type of cancer, e.g., breast, colon, lung cancer. Most of the data sets mentioned here, including the patch-based ones, require pre-processing (except KIMIA960, PCam) since they are not in a manageable size by the CNNs with existing computational devices. Among the full WSI-based data sets, CAMELYON data sets (both 16 and 17) are annotated at

[7] https://web.inf.ufpr.br/vri/databases/breast-cancer-histopathologicaldatabase-breakhis/

[8] http://kimia.uwaterloo.ca/kimia_lab_ data_Path960.html

[9] http:// bioimage.ucsb.edu/research/bio-segmentation

[10] https://rdm.inesctec. pt/dataset/nis-2017-003

[11] https://warwick.ac.uk/fac/sci/dcs/ research/tia/glascontest/

[12] https://mitos-atypia-14.grand-challenge.org/dataset/

[13] http://ludo17.free.fr/mitos_ 2012/index.html

[14] https://github.com/basveeling/pcam

[15] https://drive.google.com/drive/folders/0BzsdkU4jWx9Bb19WNndQTlUwb2M

[16] https://camelyon17.grand-challenge.org

[17] https://portal.gdc.cancer.gov/ legacy-archive/

[18] https://brd.nci.nih.gov/brd/image-search/ searchhome

pixel-level, others are annotated in slide-level/case-level or are weakly annotated. In the later case, their use requires extra efforts from domain experts for model training as well as for evaluation [Hu et al., 2018; Komura and Ishikawa, 2018]. We utilized PCam and CAMELYON16 from the patch-based and full WSI-based data sets, respectively, and a full WSI-based private data set fully annotated by expert pathologists. This section describes the data sets we utilized in this thesis.

2.4.1 Multi-class data set: the MLNTO data set

We used the Metastatic Lymph Node data set from the Institut Universitaire du Cancer Toulouse Oncopole (IUCT-Oncopole), which is abbreviated as MLNTO. It is a private data set composed of WSIs of metastatic lymph nodes from various primary tumors, such as melanoma, adenocarcinoma, and squamous cell carcinoma relating to various anatomical sites. The data set contains 61 WSIs of lymph nodes with 34 WSIs in the training set and 27 in the test set. All the WSIs are pre-extracted from the multi-resolution pyramid structure at level 3, i.e., the WSIs are 8-times downsampled from the highest resolution. The average size of a training WSI is $9,488 \times 14,648$ pixels (at level 3).

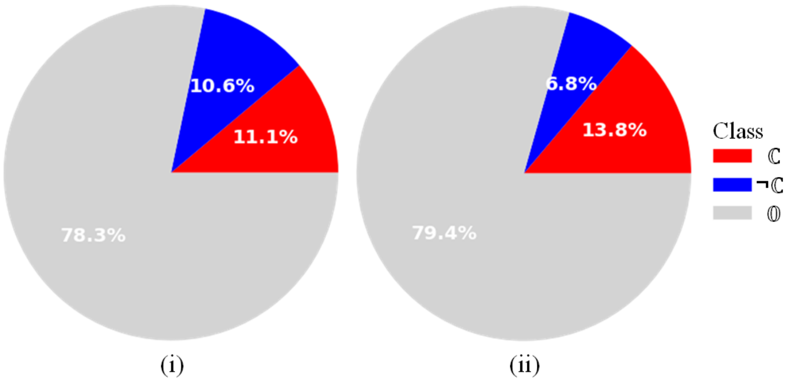
In the preparation process of the WSIs, the glass slides of the histological biopsy were stained with H&E and digitized with a 3DHISTECH Pannoramic 250 digital slide scanner at 0.243 μm per pixel resolution. Two expert pathologists provided the ground-truth annotation masks for all the WSIs manually while the WSIs were downsampled 8 times (i.e., level 3) with respect to the highest resolution (level 0). The manual annotation processed by pathologists (by Dr. Camille FRANCHET and Dr. Margot GASPARD from IUCT-Oncopole under the supervision of Prof. Pierre BROUSSET) was done in two steps:
- The WSIs were first analyzed using *Definiens* [19] *Developer XD* software. Superpixels were generated using the multi-resolution segmentation function provided by the image analysis environment. As a result, images were roughly segmented.
- Each superpixel was then annotated by the pathologists who defined three classes: "*metastasis/cancer*" ($\mathbb{C}$), "*lymph node/non-cancer*" ($\neg\mathbb{C}$), and so-called "*other*" ($\mathbb{O}$) which can be either background or histological structures not included in the first two classes, such as adipose or fibrous tissue (Figure 2.3).

[19] www.definiens.com : Definiens (the tissue phenomic company) provides image analysis tools, which is used in particular for tissue phenomics analysis.

Table 2.2: Pixel statistics reporting the average number (in millions and percentage) of pixels of each class per WSI in the training and test sets of MLNTO data set. Considered WSIs are downsampled by a factor of 8 from the original resolution.

		$\mathbb{C}$	$\neg\mathbb{C}$	$\mathbb{O}$
Train	Mean in millions	15.2	14.6	107.4
	Mean in %	11.1	10.6	78.3
Test	Mean in millions	20.0	9.8	114.8
	Mean in %	13.8	6.8	79.4

These manually annotated masks are used as the ground truth. On average, three hours per image were needed by an experienced pathologist for annotation. The ground-truth annotations make MLNTO suitable for the segmentation task, although it can be adapted for the classification task as well.



(i) (ii)

Figure 2.9: The natural class distributions of the pixels in the training (i) and test (ii) sets of the MLNTO data set.

With one exception, all the WSIs included in the MLNTO data set contain metastasis. However, the metastatic WSIs contain enough healthy regions to collect $\neg\mathbb{C}$ examples for training and testing. Table 2.2 and Figure 2.9 shows the natural class distributions of the pixels in the training and test sets. The statistics indicate that the natural data distribution is imbalanced with an over-representation of class $\mathbb{O}$ in comparison to classes $\mathbb{C}$ and $\neg\mathbb{C}$. In the training set, the percentage of class $\mathbb{O}$, $\mathbb{C}$, and $\neg\mathbb{C}$ are on average 78.3%, 11.1%, and 10.6%, respectively, while in the test set, they are 79.4%, 13.8%, and 6.8%. The imbalanced nature of the data convinced us to better understand the impact of a balanced versus an imbalanced distribution of the classes for the learned models, as well as the difficulty of choosing which WSIs to annotate when creating a training set.

2.4.2 Binary class data set: the CAMELYON16 and Patch Camelyon

We also have utilized two well-known binary class benchmark data sets, namely CAMELYON16[15] [Litjens et al., 2018], and PCam[14] [Veeling et al., 2018].

The CAMELYON16 data set is a breast cancer data set consisting of 399 WSIs of sentinel lymph nodes, with 270 WSIs in the training set and 129 WSIs in the test set. There are 111 training WSIs and 49 test WSIs containing tumors (i.e., class $\mathbb{C}$). All the WSIs are in a multi-resolution pyramid structure, generally with 10 levels of resolution. In the ground truth, only class $\mathbb{C}$ regions are annotated at pixel-level. Unlike the MLNTO data set, the non-lymph node areas (i.e., class $\mathbb{O}$) are not separated from the lymph node ones (i.e., class $\neg\mathbb{C}$), and both classes are considered as normal/non-cancer/$\neg\mathbb{C}$ class. It is a large data set with full-sized WSIs and required huge pre-processing to use for CNN training; we thus use another data set named PCam -which is a subset of CAMELYON16 and already pre-processed- described in the following paragraph. It helps us concentrate on the main focus of this thesis, the class distribution analysis. Although we did not use the CAMELYON16 training set to avoid huge pre-processing, we utilized its test set as an example of an imbalance test set. We describe how we utilize it after describing the PCam data set since both are linked.

The PCam data set is a new benchmark generated from CAMELYON16. Unlike the MLNTO and CAMELYON16 data sets, PCam is a ready-to-go data set, as the patches have already been separated. PCam contains 327,680 patches of 96 × 96 pixels at level 2 (i.e., 4-times downsampled from the highest resolution). In PCam, the ratio of training, validation, and test data is 6:1:1. The data set is balanced, i.e. the ratio of $\mathbb{C}$ and $\neg\mathbb{C}$ patches is 1:1 (Figure 2.10(i)). The data set considers 32 pixels at the edge as border pixels, while the central region is 32×32 pixels. The patch-level labels are established as follows: if there are any class $\mathbb{C}$ pixels in the central region, the whole

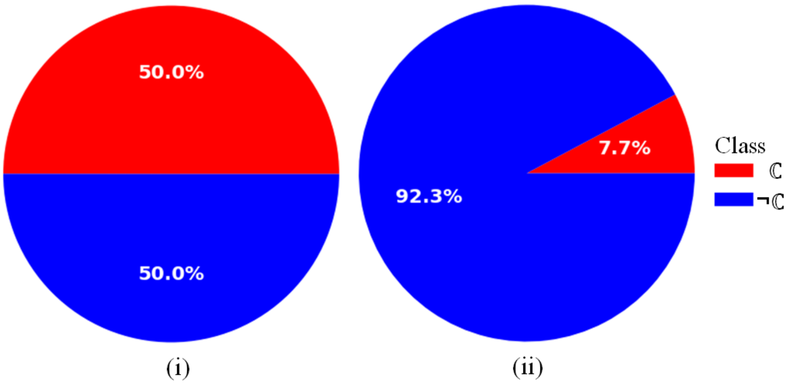


Figure 2.10: Class distributions of the PCam data set (i) and the extracted patches from CAMELYON16 test set (ii).

patch is labeled as category ℂ; otherwise, the patch-level label is ¬ℂ.

By using the same patch extraction process of the PCam data set, the number of extracted test patches from the CAMELYON16 test set is 10,487,709. Specifically, the patches were extracted after filtering out the background of WSIs at level 2, the stride being 32 and the central region being 32×32 pixels. Figure 2.10(ii) shows the class distribution of the extracted patches, where the ratio of ℂ to ¬ℂ is 1:12.

2.5 Evaluation measures

According to the definition of learning system given by Mitchell et al. [1997], the remaining element in a learning system is the performance measure (**P**)/evaluation measure/evaluation metric. This section covers that topic for our task of interest, cancer detection from WSIs. We conducted two different surveys on this topic: one by reading articles on cancer detection task, another by the PubMed[20] search.

First survey. We conducted our first survey on evaluation metrics by reading articles on cancer detection from image data, including WSIs. Specifically, Hu et al. [2018] surveyed 79 deep learning-based methods for cancer detection from image data, and we did our survey on the evaluation metrics authors of those methods utilized, in addition to 16 more papers that we find interesting but not included in those 79 papers. The statistics of different evaluation metrics are given in Table 2.3. Most of the papers we surveyed utilized more

[20] https://pubmed.ncbi.nlm.nih.gov/

Table 2.3: **Evaluation metric statistics collected from the deep learning based methods for cancer detection from image data**. The first column belongs to the metric name and the second column indicates the number of papers from our survey that utilized the metric.

Metric	#times utilized
Accuracy	40
Recall, Sensitivity, True-Positive Rate (TPR)	36
ROC/ROC-AUC	24
Dice coefficient/Dice coef./Dice index	23
Precision, positive predictive value (PPV)	20
F1-measure	16
Specificity	14
Hausdorff distance/Hausdorff	8
Jaccard index/Jaccard coef.	5
False-Positive Rate (FPR)	5
Free response operating characteristic curve (FROC)	5
Average surface distance	4
Competition Performance Metric (CPM)	3
Volume intersection ratio/ average volume intersection	2
Negative predictive value	1
Overlapping ratio	1
True-Negative Rate (TNR)	1
Root mean square (RMS) error	1
False-Positive (FP)	1
True-Positive (TP)	1
False-Negative (FN)	1
Volume error	1
Concordance index	1
PR curve	1

than one metric depending on the work they did. According to our survey, the most utilized metrics are accuracy, recall, ROC, dice coefficient, and precision.

The evaluation metrics can mainly be divided into two categories: *single threshold-based / single-valued scalar metrics* (e.g., accuracy, recall, precision, dice-coefficient) and *multi-threshold-based / threshold-free / multidimensional graphical metrics* (e.g., ROC, FROC, PR curves). The evaluation metrics that require selecting a threshold of the predicted score (e.g., predicted probability) to compute the metric values are categorized as single threshold-based metrics otherwise, they are threshold-free. Threshold-free metrics can also be defined as multi-threshold metrics since they consider all possible thresholds instead of selecting one. They are named as threshold-free since they do not require any particular threshold selection (like single threshold-based ones) in their computation.

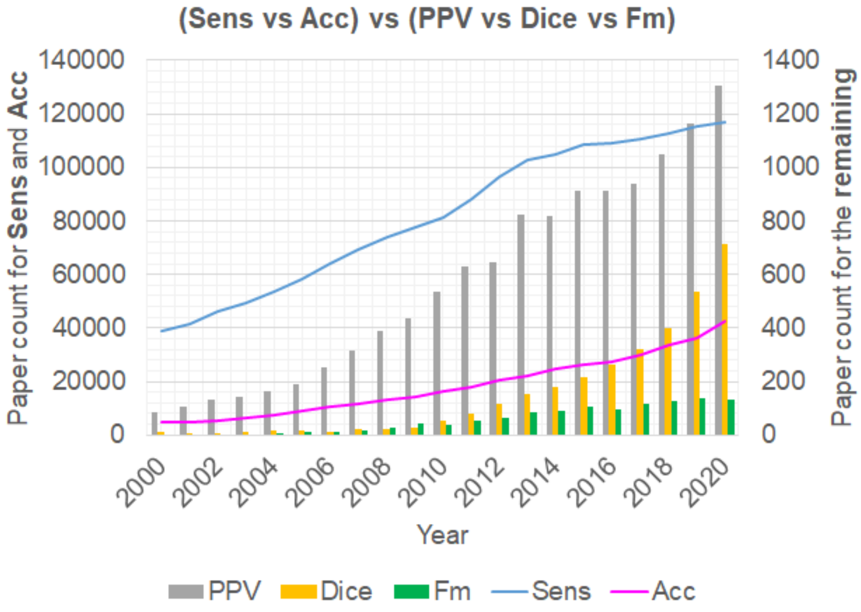


Figure 2.11: **Paper count per year for the single threshold-based evaluation metrics**: we collected the statistics from the *PubMed* Search for the terms “Sensitivity”, “Accuracy”, "Positive predictive value (PPV)", "Dice coefficient", and "F-measure" for the last 20 years. Here, Sens: sensitivity/recall, Acc: accuracy, PPV: precision, Dice: dice coefficient, and Fm: F-measure. The paper count per year for the sensitivity and accuracy are much higher than the remaining metrics. We thus present them in two different scales in the y-axis: left side for the sensitivity and accuracy, and right side for the remaining.

Second survey. Since we conducted our first survey on limited articles,

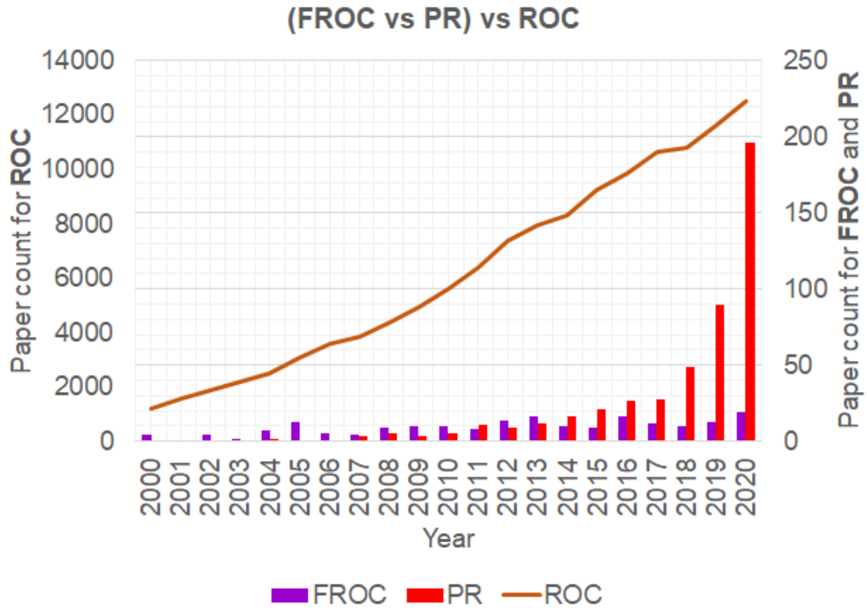


Figure 2.12: **Paper count per year for the threshold-free / multi-threshold evaluation metrics**: we collected the statistics from the *PubMed* Search for the terms “FROC”, “Precision-Recall”, and “ROC” for the last 20 years. The paper count per year for the ROC curve is much higher than the two other metrics. We thus present them in two different scales in the y-axis: left side for the ROC and right side for the FROC and PR.

we conducted a second one on broader range by utilizing the result of our first survey. Our second survey was based on the articles in PubMed for the most frequently used metrics according to our first survey (see Table 2.3 for our first survey result). We present the second survey statistics in Figure 2.11 and 2.12 for the single threshold-based and multi-threshold-based metrics, respectively. These statistics are almost consistent with our previous survey presented in Table 2.3: the most utilized single threshold-based metrics are recall/sensitivity, accuracy, and precision, while the most utilized multi-threshold-based metrics are ROC and PR curves. Although different papers might use a different name for the same metric, we consider only the most popular name used in the biomedical field to search for the paper count for that metric in PubMed; the actual paper counts might be thus slightly different from the presented ones. Nevertheless, these surveys present a concise idea about the popular metrics utilized in the biomedical field.

Based on these mini-surveys, we selected our evaluation metrics suitable

Table 2.4: **Confusion matrix for binary classification problem**. Here, TP, FP, TN, and FN are the number of True-Positive, False-Positive, True-Negative, and False-Negative, while neg and pos are the total number of negative and positive samples in the test set.

	Predicted negative	Predicted positive	
Actual negative	TN	FP	$neg = FP + TN$
Actual positive	FN	TP	$pos = TP + FN$

for our task of interest. We considered several metrics in our evaluation because no single metric can serve all our purposes [Bylinskii et al., 2018]. For all the considered evaluation metrics, a pre-requisite is to compute a confusion matrix. It is a table that presents the statistics of the true and false predictions by a model. In this thesis, we consider binary classification, even for the multi-class data set, we convert the problem to a binary one by considering the class of interest as positive and the remaining classes as negative. For a binary class problem, a confusion matrix has four elements: the number of TP, FP, TN, and FN (see Table 2.4). All the considered metrics can be defined by the elements of the confusion matrix.

Short descriptions of our selected metrics are given in the following sections.

2.5.1 Single threshold-based evaluation metrics

We utilized the popular metrics, recall (also known as sensitivity or TPR) and precision (also known as PPV), as single threshold-based evaluation measures. The recall is useful to reflect the TP, while precision is useful to reflect the FP. Since recall and precision varies in reverse order, it is important to report both. To evaluate the model performance by considering both recall-precision at the same time, we also reported F-measure. They are computed as follows:

$$Recall = \frac{TP}{TP + FN} \tag{2.1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2.2}$$

$$F - measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{2.3}$$

In Equation 2.1, the denominator, $(FN + TP)$ is the number of positive samples (pos, second row in Table 2.4) in the test set while in Equation 2.2, the denominator, $(FP + TP)$ is the total number of predicted positives (second column in Table 2.4) by the respective model.

The major limitation of using single threshold-based metrics is their values depend on pre-selected thresholds, and they do not provide the opportunity to users of selecting an optimal threshold by providing a range of options. Nevertheless, they are useful to present the result in a compact way.

2.5.2 Threshold-free / multi-threshold-based evaluation metrics

We utilized ROC and PR curves as threshold-free/multi-threshold-based evaluation metrics since they are popular in the literature according to our survey (see Figure 2.12). The ROC curve expresses the trade-off between the TPR and the FPR, while the PR curve expresses the trade-off between precision and recall. Among the elements of ROC and PR, TPR/recall (Equation 2.1) and precision (Equation 2.2) have been defined previously; the remaining element FPR is defined by the following equation:

$$FPR = \frac{FP}{FP + TN} \tag{2.4}$$

where the denominator, $(FP + TN)$ is the number of negative samples in the test set.

By changing the threshold on predicted scores, it is then possible to match the desired FPR (resp. recall) [Freeman and Moisen, 2008; Grau et al., 2015], find the associated TPR (resp. precision), and plot the corresponding point on the ROC (resp. PR) curve. Each point on a ROC curve is thus calculated with a different threshold. Following the definition given by Boyd et al. [2013], mathematically,

$$ROC_{curve} = \{(fpr(t), tpr(t)) : -\infty < t < \infty\} \tag{2.5}$$

$$PR_{curve} = \{(recall(t), prec(t)) : -\infty < t < \infty\} \tag{2.6}$$

where $fpr(t)$, $tpr(t)$, $recall(t)$, and $prec(t)$ are the FPR, TPR, recall, and precision computed at a threshold t. From a ROC or PR curve of a single model, it is possible to know the threshold information, t; however, there is no guarantee to have it from a mean ROC or PR curves of multiple models (we detail it in Chapter 5).

Along with the ROC and PR curves, we also utilized precision and FPR curves -where the precision and FPR are presented as a function of continuous threshold in two separate curves- to serve our purpose of reflecting FP in a range of thresholds. The precision curve thus can be defined as the plot of precision at different thresholds of the predicted score, and similarly, the FPR curve is the plot of FPR at different thresholds of the predicted score. Mathematically,

$$Precision_{curve} = \{(t, prec(t)) : -\infty < t < \infty\} \tag{2.7}$$

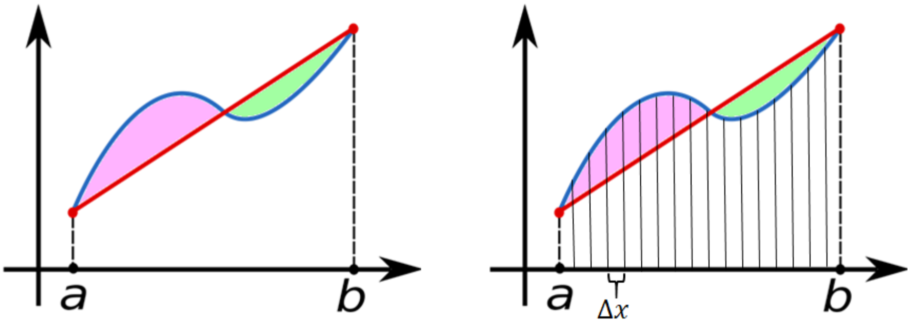
$$FPR_{curve} = \{(t, fpr(t)) : -\infty < t < \infty\} \tag{2.8}$$

where $fpr(t)$ and $prec(t)$ are the FPR and precision, respectively computed at a threshold t.

The presentations of precision and FPR as functions of threshold can provide several advantages over the ROC and PR that we cover with examples in Chapter 4. Moreover, by observing the advantages of these curves, we formally present a new metric named PR-T curves as an alternative to the PR curve. The PR-T curves provide the opportunity for threshold selection even in the case of mean curves. We detail it in Chapter 5.

2.5.3 The area under the curve (AUC)

The area under a curve (AUC) can be calculated for any curve-based metric, including the ones presented in Section 2.5.2. The AUC has great importance in expressing the model performance with a single scalar value. It helps to express the model performance without plotting any curve [Brodersen et al., 2010; Yu, 2012]. We thus define it separately in this section.



(i) (ii)

Figure 2.13: **AUC approximation**. (i) If we directly apply the trapezoidal rule to approximate the AUC of curve function (in blue), the AUC will be equal to the area under the linear function (in red). The pink area from the AUC will be then excluded, while the green area will be extra added. (ii) If we divide the AUC into small enough portions and apply trapezoidal rule to each portion, the problem can be solved.

It is defined as the definite integral, $\int_a^b f(x)\,dx$ [Tallarida and Murray, 1987]. Here, $f(x)$ is the function of x, where $[a, b]$ is the range of the x. x and $f(x)$ are plotted on the x- and y-axis of the curve space. In the ROC curve, for example, the TPR (i.e., $f(x)$) is presented as the function of FPR (i.e., x) and the range of the TPR value is $[0, 1]$. If $f(x)$ is a linear function (Figure 2.13: red line), the area under the function $f(x)$ can be correctly approximated by the trapezoidal rule (Equation 2.9).

$$\int_a^b f(x)\,dx \approx (b-a)\frac{f(a)+f(b)}{2} \tag{2.9}$$

If $f(x)$ is a curve function (Figure 2.13: blue curve), however, the area under the function $f(x)$ (i.e., AUC) cannot be correctly approximated by the trapezoidal rule given in Equation 2.9 since the pink area from the AUC will be excluded, while the green area will be extra added in that case. According to the general rule of integral calculus, we have to divide the AUC into small enough portions and apply trapezoidal rule to each portion to compute the area of that portion, then summing up all the computed areas approximates the AUC value (Figure 2.13: ii). We can achieve it by dividing the range $[a, b]$ into small enough N intervals, i.e., making an interval, $\triangle x$ ($= x_{k+1} - x_k$) $\longrightarrow 0$, and applying the trapezoidal rule as follows gives the approximation of AUC:

$$\int_a^b f(x)\,dx \approx \sum_{k=1}^{N}(x_{k+1} - x_k)\frac{f(x_k) + f(x_{k+1})}{2} \tag{2.10}$$

2.6 Conclusion

To summarize, the whole slide image is a very special kind of image data currently utilized by researchers to develop computer-aided cancer detection systems. They are complex regarding file structure, size, and pattern, which makes them different from other images. The current advances in other imaging modalities are thus not directly applicable to them. Machine learning algorithms -both classical and deep learning- are utilized to develop computer-aided detection systems, where deep learning, specifically, CNNs produce state-of-the-art results. Any such learning-based system has three crucial elements: the data set from where the system gathers experience, the task for which the system is built for, and the performance measures that evaluate the system efficiency. We surveyed and discussed all these three elements. Based on the surveys, we determine CAMELYON16, PCam, and MLNTO as the data sets and recall, precision, F-measure, ROC, PR, precision, and FPR curves as the evaluation metrics from the state-of-the-art for our target task cancer detection.

Chapter 3

Machine Learning Model for Cancer Detection

Contents

Conducting an analytical study (e.g., on class distribution and evaluation metric) requires designing a general methodological framework that generates predictive models for a targeted task (e.g., cancer detection). The framework should have all the elements that facilitate the desired analysis. In this chapter, we propose our ones. At first, we review the literature on cancer researches utilizing medical images (both radiology and histology images), image processing, and machine learning techniques, then describe our developed framework for cancer detection in WSI. The framework is applicable to both classification and segmentation tasks. In our framework, we proposed a patch extraction and categorization method that overcomes the limitation of usually adopted random patch sampling. Moreover, the framework includes an evaluation methodology for segmentation tasks which is more relevant to the pathologists' way of checking a WSI.

The structure of this chapter is as follows: Section 3.1 reviews the related work on cancer research utilizing both non-WSI and WSI data. Section 3.2 articulates our methodological framework and hyper-parameter settings to produce CNN-based models to detect cancer on WSIs. We present our patch-based evaluation method for the segmentation task, which is a part of our proposed framework, in a separate Section 3.3. Finally, Section 3.4 concludes the chapter.

3.1 Related work

3.1.1 Cancer research based on non-WSI data

Articles published on non-WSI images for cancer research are numerous. For example, Roth et al. [2015] proposed a two-tiered coarse-to-fine cascade framework for bone lesions, enlarged lymph nodes, and colonic polyps classification from computed tomography. In the first tier, they utilized an existing detection system as a candidate generation system at sensitivities of 100% but at high FP levels from where the coordinates of regions or volumes of interest were generated. The candidates were the input for the second tier. In the second stage, they generated 2D or 2.5D views via sampling through 4 different scale transformations, 5 different random translations, and 5 different rotations. These random views were used to train deep CNN classifiers (modified AlexNet). In testing, CNNs assigned class probabilities for a new set of random views (4*5*5=100) that were then averaged to compute a final per-candidate classification probability. They showed that the proposed method improved the sensitivity from 57% to 70%, 43% to 77%, and 58% to 75% at 3 FPs per patient for sclerotic metastases, lymph nodes, and colonic polyps, respectively. However, it was not clear enough whether they re-trained the model with candidate FPs only or the data-augmented version of all training examples.

In [Shen et al., 2015], authors proposed a multi-scale Convolutional Neural Networks (MCNN) architecture to extract multi-scale features from nodules in computed tomography with different sizes. In their method, they concatenate the features from three parallel CNNs. The features were extracted from three different sizes of patches, which were reshaped into the smallest patch size to train CNNs. The features from CNNs were concatenated to construct the final feature vector. The extracted CNN features were used to train SVM and random forest classifiers then the results were compared with the corresponding results for histogram of oriented gradients(HOG) and local binary pattern (LBP) features. According to their experiments, MCNN outperformed HOG and LBP descriptors with 10.91% and 13.17%, respectively, and the MCNN feature was noise tolerant.

Lian et al. [2016] proposed a framework for predicting the outcome of cancer therapy from multi-sources of information, including radiomics in FDG-PET images and clinical characteristics. As the main contribution, they improved the features selection method, EFS [Lian et al., 2015] to select features from uncertain, small-sized, and imbalanced data set. They experimentally proved the effectiveness of their method on two real data sets.

In [Sun et al., 2017b], the authors proposed a new multi-channel ROI combining the original ROI, nodule ROI, and gradient ROI as RGB channel, which was applied for lung cancer diagnosis in computed tomography images. They also compared different deep learning-based features with traditional hand-craft features. According to them, their multi-channel ROI won over original ROI by 2.8%; deep learning feature won over traditional feature by 2.3%.

In [Sun et al., 2017a], authors hypothesized that combining a small amount of labeled data with a large amount of unlabeled data could be a solution to collect enough data to train the deep learning algorithms. To test this hypothesis, they designed a graph-based semi-supervised learning scheme consisting of three modules: data weighing, feature selection, and a newly proposed dividing co-training data labeling, and utilized a deep CNN. They also compared three different dimension reduction methods and three different weighing functions. For the breast cancer mammogram data, the difference of accuracy between the originally labeled data and their semi-supervised data was 3.75% at voting threshold 7. They concluded that the unlabeled data could not replace labeled data, and using unlabeled data is only a supplement.

In [Zhou et al., 2021], the authors proposed a novel segmentation architecture by including a correlation model and attention-based fusion modules to combine multiple U-net architectures for multiple imaging modalities. According to them, this was the first segmentation method, which is capable of describing the latent multi-source correlation representation among modalities and allows to help segmentation on missing modalities. They tested the efficiency of the proposed architecture on two data sets of multi-modal MRI of brain tumor data and got competitive results with state-of-the-art methods.

In summary, different dimensions have been investigated, such as FP reductions, features engineering, combining multi-modal data, utilizing unlabeled data and proposed different interesting methods in the domain of radiology images. The studies, however, are not directly applicable to histology images, WSIs, because of the huge difference in image characteristics [He et al., 2012]. The FP reductions and efficient data utilization could also be interesting tracks to investigate with WSI. This thesis contributes to similar problems, FP reductions, and data utilization for WSI.

3.1.2 Cancer research based on WSI data

During last decades, several studies were done to facilitate computer-aided diagnosis for cancer detection from WSIs [Barisoni et al., 2012; Wang et al., 2016a; Bejnordi et al., 2017; Liu et al., 2017, 2019; Evans et al., 2018; Khan et al., 2019; Bera et al., 2019; Dimitriou et al., 2019; Barisoni et al., 2020; Zhou et al., 2020]. Although at the early age of WSIs, classical machine learning algorithms with feature engineering were mainly utilized [Gurcan et al., 2009; Komura and Ishikawa, 2018], in recent years, the deep learning algorithms with the representation learning are mainly emphasized [Hu et al., 2018; Li et al., 2021a]. Several deep learning systems have been proposed during the last few years, specifically from 2015 onward, and has shown incredible performance levels [Liu et al., 2019]. The recommended methods, however, have not been adopted at a clinical level, as the performance threshold needed to gain the trust of pathologists has not yet been met [Bera et al., 2019]. Therefore, developing efficient methods for cancer detection from WSIs remains an active area of research [Fan et al., 2019b].

Different dimensions, such as pre-processing of WSIs [Khan et al., 2014; Vahadane et al., 2016; Zheng et al., 2019; Tellez et al., 2019; Salvi et al., 2020; Alzubaidi et al., 2020], transfer-learning [Chen et al., 2016; Khan et al., 2019; Han et al., 2020; Alzubaidi et al., 2020; Otálora et al., 2021], developing end-to-end systems [Wang et al., 2016a; Liu et al., 2017, 2019; Lin et al., 2018; Veeling et al., 2018; Fan et al., 2019a], proposing new network architectures, or utilizing the existing ones differently [Chen et al., 2016; Lin et al., 2018; Alzubaidi et al., 2020; Toğaçar et al., 2020; Shirazi et al., 2020], post-processing of the predictions [Liu et al., 2019; Kaushal and Singla, 2020] are being explored.

For example, Vu et al. [2015] proposed a cancer detection system utilizing the classical machine learning algorithm, SVM. They proposed an automatic feature extraction method via learning class-specific dictionaries. In [Salvi et al., 2020], the authors proposed a fully automated stain separation and normalization approaches for H&E-stained WSIs named SCAN (Stain Color Adaptive Normalization). The proposed algorithm was based on segmentation and clustering strategies for cellular structure detection. According to them, SCAN was able to improve the contrast between histological tissue and background and preserve local structures without changing the color of the lumen and the background. They showed that the proposed method was both quantitatively and qualitatively superior to the state-of-the-art techniques, although some other studies claim that data augmentation is more preferable to color normalization for dealing with color variations [Liu et al., 2019; Otálora et al., 2021].

From 2015, Bejnordi et al. have organized a worldwide challenge known as CAMELYON to gather together different methods for cancer detection in WSIs; the overview of their first challenge, the CAMELYON16, is published in [Bejnordi et al., 2017]. Before this annual event, the use of WSIs in computational tasks was limited to patches that had been pre-extracted by the pathologist [Lin et al., 2018]. In the challenge, there were two sub-tasks, the metastasis/cancer identification and localization task (task 1) and the WSIs classification task (task 2). A total of 23 teams submitted 32 methods, and 21 teams described their methodologies. All methods proposed in the challenge followed a similar workflow: 1) pre-processing of WSIs, 2) training a machine learning model for detection of tumor regions, 3) producing tumor probability map for each test slide with the trained model, and 4) post-processing probability maps to produce tumor lesion locations and scores, and a score for the entire slide. As a machine learning algorithm, most of the proposed methods utilizing deep learning: the variation in the participants' results was induced by hyper-parameter settings and data pre-processing. The methods that placed within rank 10 for at least one sub-task are summarized in Table 3.1. In the table, we present only some of the promising hyper-parameters: the training set distribution, the network architecture, if the methods used transfer learning, data augmentation, stain normalization or not. The result indicates that hyper-parameter settings contribute to the models' performances, although the fair comparison is not ensured because of the variable settings of different methods.

Table 3.1: **Top 10 methods in CAMELYON16 challenge.** Here, First column shows the team name and submission ID in the challenge, Distribution: the training set size with class distribution, ML algo.: machine learning algorithm, TL: transfer learning, DA: data augmentation, SN: stain normalization, Y: yes, and N: no.

Team name	Distribution	Patch sampling	ML algo.	TL?	DA?	SN?	Result	
							Task-1: FROC-AUC (rank)	Task-2: ROC-AUC (rank)
HMS & MIT II	Two million for each class. They mentioned it differently in different place; Maybe ¬ℂ-biased	Patches were sampled uniformly from positive and negative regions. Hard negative mining was performed after initial classification to augment the training set.	22-layers GoogLeNet	N	Y	Y	0.807 (1^{st})	0.994 (1^{st})
HMS & MGH III	400,000 with online sampling with approximately 25% positive samples (¬ℂ-biased)	Negative patches were sampled from both negative slides and normal regions in positive slides.	Fully Convolutional ResNet-101	Y	Y	N	0.760 (2^{nd})	0.976 (2^{nd})
HMS & MGH II	25% positive samples (¬ℂ-biased)	Same as HMS & MGH III	ResNet-101	Y	Y	N	0.729 (3^{rd})	0.908 (8^{th})

Continue on the next page

Table 3.1 Top 10 methods in CAMELYON16 challenge

Team name	Distribution	Patch sampling	ML algo.	TL?	DA?	SN?	Result	
CULab III	15 million (5% positive: ¬ℂ-biased)	Uniform sampling	Fully Convolutional VGG-16	Y	Y	N	0.703 (4^{th})	0.94 (4^{th})
HMS & MIT I	Two million for each class (balanced)	Same as HMS & MIT II	22-layers GoogLeNet (balanced)	N	Y	N	0.693 (5^{th})	0.923 (5^{th})
HMS & MGH I	Same as HMS & MGH II	Same as HMS & MGH II	GoogLeNet	Y	Y	N	0.596 (6^{th})	0.964 (3^{rd})
Radboud-UMC	150,000 positive and 150,000 negative (balanced)	Patches were sampled uniformly from positive and negative regions. Normal patches were sampled from negative slides as well as non-metastatic regions in tumor slides.	15-layers VGG-like	N	Y	N	0.575 (7^{th})	0.779 (17^{th})
CULab I	15 million (5% positive: ¬ℂ-biased)	Uniform sampling	VGG-16	Y	Y	N	0.544 (8^{th})	0.909 (7^{th})

Continue on the next page

Table 3.1 Top 10 methods in CAMELYON16 challenge

Team name	Distribution	Patch sampling	ML algo.	TL?	DA?	SN?	Result	
CULab II	Same as CULab I	Same as CULab I	Cascade of two CNNs: VGG-16, ResNet-152	Y	Y	N	0.527 (9^{th})	0.906 (9^{th})
ExB	1.6 million (Natural)	Training was started with a balanced sampling between the positive and negative class. As the training proceeded the distribution of positive/negative samples was slowly changed to match the original distribution in the images.	ResNet-34	N	Y	N	0.511 (10^{th})	0.915 (6^{th})

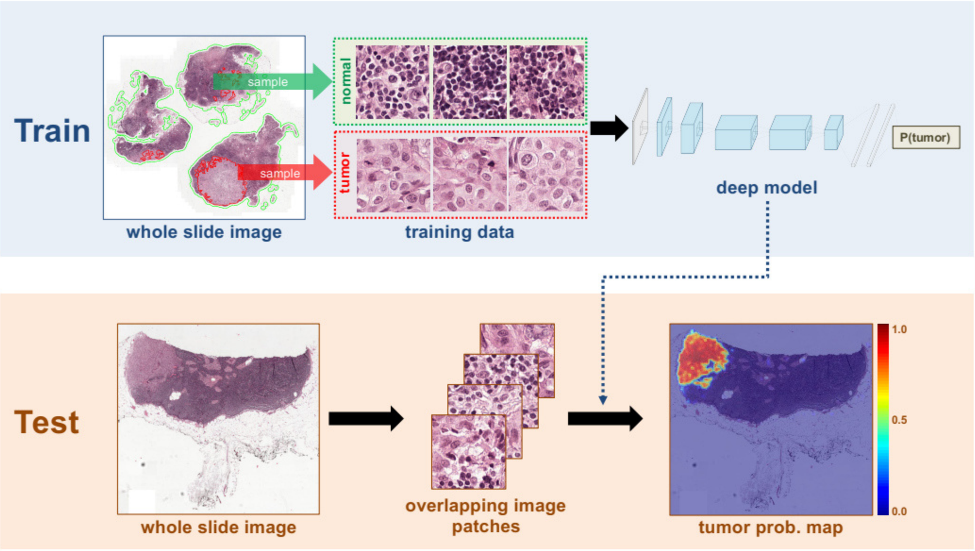


Figure 3.1: The framework for cancer detection from WSI utilized by the winner of CAMELYON16 (Figure 2 of [Wang et al., 2016a]).

The winning team [Wang et al., 2016a] in the CAMELYON16 challenge trained two 22-layer GoogleNets (V1), one with randomly sampled training patches (HMS & MIT I in Table 3.1) and another with additional hard negative examples–probably biased towards negative examples (HMS & MIT II in Table 3.1). The final decision was based on the combination of predictions from the two models. Additionally, they trained a random forest classifier with 28 handcrafted features extracted from the output heatmaps of the CNNs. They used color normalization to cope with color variation in WSIs while also applied rotation and color noise to the training patches for data augmentation. Figure 3.1 shows the basic framework they utilized, where they collected positive (metastatic cancer) and negative (normal tissue) samples randomly to train a deep model. The trained model was then used to predict the overlapping patches from unseen test WSIs, then stitched them together to generate the probability heatmap. They post-processed the heatmaps to generate the result for task1 (the metastasis/cancer identification and localization task) and task2 (the WSIs classification task).

Liu et al. [2017] (from GoogleAI) improved the result of CAMELYON16's winner. They used the GoogleNet (V3) [Szegedy et al., 2015] and applied a random patch sampling technique to obtain a balanced training set. Moreover, to deal with the scarcity of tumor patches, they applied several data augmentation techniques, including rotation, mirroring, and extensive color perturbation. However, in their extended work [Liu et al., 2019], they selected a training distribution biased towards the negative class by a factor of four. Lin et al. [2018] proposed a framework for fast and dense scanning during prediction. In their framework, they converted the modified VGG16 to a Fully Convolutional Neural Network (FCNN) -a CNN architecture, where dense layers in the classification part are replaced with the 1×1 convolutional layers-, which was followed by a patch reconstruction method. To train their model, they used FP generating patches from the training set. Veeling et al. [2018] proposed a rotation equivariance framework by adopting G-CNN architecture [Cohen and Welling, 2016]. To test the effectiveness of their method, they proposed a new data set known as PCam (see Section 2.4.2). In [Mejbri et al., 2019; Mejbri, 2019], authors proposed a new data set for different types of breast cancer, and an end-to-end deep learning framework for multilabel tissue segmentation utilizing their data set while network parameters were determined with a deep analysis.

In [Chen et al., 2016], authors proposed a deep cascaded neural network (CasNN) for mitosis detection in breast cancer WSIs, which is a combination of different existing methods. Their framework was composed of two models: a coarse retrieval model with FCNN to identify and locate the candidates of mitosis with high sensitivity, and three fine discrimination models with caffeNet [Jia et al., 2014] and transfer learning (from ImageNet data) to fine-tune the candidate. The average and threshold were used to determine the final output. In [Toğaçar et al., 2020], authors proposed a CNN architecture named BreastNet consisting of convolutional, dense, and residual blocks. They compared their method with the AlexNet, VGG-16, VGG-19. By utilizing the data augmentation and combining the data of different resolutions, the classification success with their method increased by 0.29% and obtained as 98.80% on BreakHis data set.

In [Khan et al., 2019], the authors proposed a deep learning-based framework for the detection and classification of breast cancer in WSI patches using transfer learning. They also provided a comparative analysis of three different deep learning architectures with respect to accuracy in the context of transfer learning. In their framework, different low level features were extracted separately by three well-known CNN architectures, GoogLeNet [Szegedy et al., 2015], VGGNet [Simonyan and Zisserman, 2014], and ResNet [He et al., 2016b] pre-trained with ImageNet [Deng et al., 2009]. They combined the extracted features using average pooling and fed them into a fully connected layer for the classification task. The images were normalization before feeding to networks. The data augmentation techniques including color processing, transformation (translating, scaling, and rotation), flipping, and noise perturbation were utilized. Compared to the previous methods, the proposed method improved the accuracy by .02%, and compared to the ResNet, it improved by 3.2%. Otálora et al. [2021] did a comparative study on fully-, weakly-supervised CNN models using two freely accessible data sets -tissue microarrays (TMAs) with strong annotations and WSIs with weak labels-, and transfer learning, targeting prostate cancer scoring. They trained models in three different ways: fully-supervised with strongly labeled TMA data set (let say, it is model m_s), feature transferred from m_s which is fine-tuned with weakly labeled WSI data set (model m_f), and weakly-supervised with weakly labeled WSI data set (model m_w), whereas all three models utilized a pre-trained model on ImageNet data set for weight initialization. Moreover, they adopted balanced distribution to the training set by applying data augmentation to under-represented classes. According to them, fine-tuned model, m_f, outperformed the remaining two for the WSI test set. Their result shows the importance of transferring features from the same base domain in the transfer learning scheme similar to the result presented in [Yosinski et al., 2014]. Nevertheless, fine-tuning with weakly labeled data might not be a rational choice for a medical domain since it produces many wrong predictions. Rather the weakly labeled data could be an option to pre-train a model to use as a base. The base should be later fine-tuned with a strongly labeled data set -at least with a small one- to correct the existing probable errors in the base model.

In [Kaushal and Singla, 2020], the authors proposed a post-processing technique on the segmented output of WSIs. Their method is based on the one-dimensional energy curve [Singla and Patra, 2017] which includes neighborhood information. They showed the effectiveness of their method over existing state-of-the-art methods.

Some other studies on image-based cancer research have been covered comprehensively in [Bera et al., 2019; Zhou et al., 2020]. Although different dimensions of research have been done on WSI, class distribution analysis has not been explored yet for it, while it is worth exploring. We investigate this topic as a part of this thesis. The investigation requires an end-to-end framework to generate predictive models for cancer detection; the survey helps us to design it. The framework we design is described in the following section.

3.2 A CNN-based model to detect cancer in WSI

In this section, we detail our approach of generating a cancer detection model for the analysis we conducted in this thesis.

3.2.1 The methodological framework

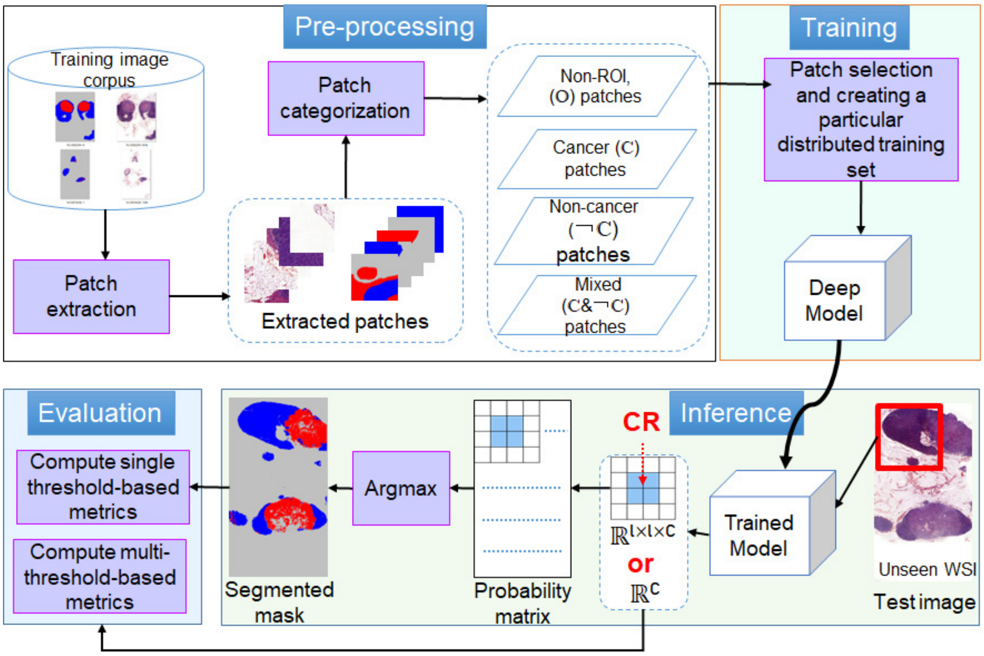


Figure 3.2: Our methodological framework for cancer detection from WSI. It is composed of 4 major steps: pre-processing, training, inference, and evaluation. Here, CR: central region.

To pursue the analytical study, we design an end-to-end cancer detection system (Figure 3.2). The system has four major steps: pre-processing, training, inference, and evaluation. All these steps are applicable for both segmentation at the pixel level and classification at the patch level settings: we mention some minor differences for each step. The framework is inspired by the one proposed by the winner of CALELYON16 challenge in [Wang et al., 2016a]. Unlike them, however, we propose to use systematic patch extraction rather than random sampling. The patch extraction method and its advantages are described in the following paragraph.

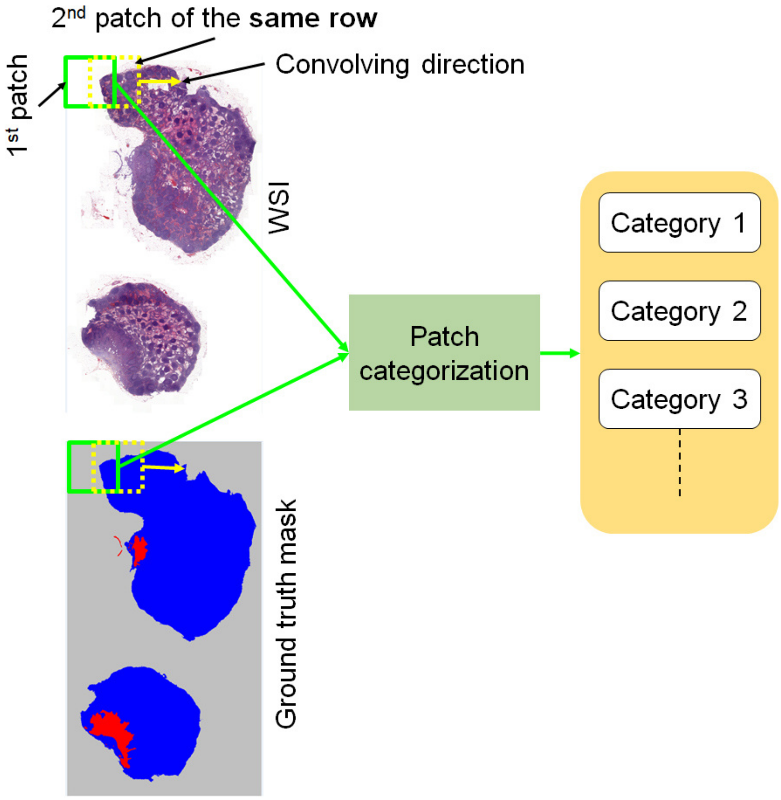


Figure 3.3: Systematic approach for extracting and categorizing patches from WSIs

Pre-processing. In the pre-processing step, we extract same-sized training patches from the training WSIs and their corresponding masks/labels from their ground truth (Figure 3.2: Pre-processing). It makes the gigantic WSIs adaptable to the CNN architecture with available memory. The dimension (l^2) of the patches is a system hyper-parameter. We extract overlapping patches.

More precisely, for a *multi-class data set* (e.g., MLNTO), the patches are extracted by convolving the WSI (and ground truth mask) from the (0,0) pixel coordinate to the last pixel coordinate with a stride of $l/2$ (making them overlapping patches). We consider overlapping patches to utilize all the tissue areas as center regions in extracted patches since there is a lack of sufficient contextual information from the border pixels of a patch [Kellenberger et al., 2018]. We did some preliminary experiments to set the patch dimension and use $l = 384$ pixels for the training patches extracted from the MLNTO data set. The extracted patches are categorized into different categories based on the pixel classes they consist of (detail of patch categorization is given in Section 4.3). The systematic approach for the patch extraction and categorization is illustrated in Figure 3.3. We developed this systematic approach for extracting and categorizing patches from WSIs as an alternative to the usual random sampling [Wang et al., 2016a; Liu et al., 2017; Bejnordi et al., 2017] since we empirically found that it performs better than with random extraction. Moreover, our approach facilitates the coverage of all desire areas of the WSI without repetition of any particular area. The repetition of a particular area -which might happen in random sampling- during training may cause biases of the model toward that repeated area. The patch categorization facilitates creating an expected class distribution during training.

For the patch-based *binary class data set* (i.e., PCam), the described preprocessing step is not required since the extraction and categorization of patches have been performed by its creators [Veeling et al., 2018] (see Section 2.4.2). We can therefore use the PCam data set as it is.

Training. In the training step, we select training patches from the extracted ones of the previous step by maintaining a desire distribution of classes. The patches from a particular category are selected randomly from all available patches for that category. The random patch selection is followed by shuffling the whole training set to prevent having all the patches in a mini-batch from the same category. This makes the convergence faster during training and provides greater accuracy [Koller et al., 2015]. The generated training set is used to train an FCNN model for the segmentation task (i.e., pixel classification) or a CNN model for the classification task (i.e., patch classification). The network architectures and hyper-parameter settings are described in Section 3.2.2.

Inference. During inference, the trained model is employed to predict either pixels (in the segmentation task) or patches (in the classification task) for unseen test WSIs. We extract same-sized overlapping patches from the WSIs. When the non-ROIs are annotated in the data set (i.e. for the multi-class data set), we predict the patches from whole WSIs by applying the same process of extracting the training patches we described. For the binary-class data set, the patches from regions with histological structures are considered for prediction. In this case, we separate the regions without histological

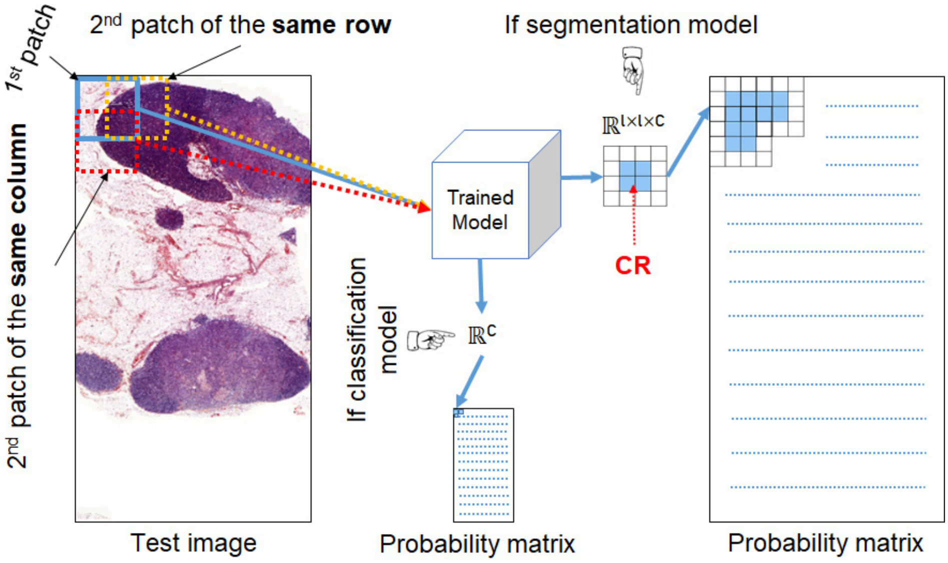


Figure 3.4: **Inferring probability matrix for a test WSI by a trained model** If the model is for segmentation, it produces probability for each class for each pixel of a patch ($\mathbb{R}^{<\times<\times\mathbb{C}}$); stitching the probabilities of the center regions in their corresponding places makes the probability matrix for the full WSI. If the model is for classification, it produces probability for each class for each patch ($\mathbb{R}^{\mathbb{C}}$); stitching the probabilities for all patches to their corresponding places produces the probability matrix for the full WSI. Here, CR means center region.

structures, i.e., background, using Otsu thresholding [Otsu, 1979]; then we extract the patches from the remaining regions following the same systematic approach described for the multi-class data set. From the extracted patches, we focus on a particular part in the middle, the central region, and the remaining part as a border for each patch. Similar to the PCam data set, the ground-truth of the central region is considered as the ground-truth of the whole patch. While the patches overlap, the central regions do not overlap. When the patches from test WSIs are fit to the trained model one by one for prediction, the trained-model produces the probability ($\in \mathbb{R}^{l\times l\times C}$ or $\mathbb{R}^{C}$, where C is the number of classes) of each class either for each pixel (for segmentation model) or for each patch (for classification model) (see Figure 3.4). For the segmentation task, stitching the probabilities of the center regions in their corresponding places makes the probability matrix for the full WSI. For the classification task, we stitch the probabilities of patches to get the probability matrix. The matrix for the classification task is downsampled by a factor of patch size, while it is the same size as the input WSI for the segmentation task. As a default choice, taking the argmax decision on that matrix produces the predicted segmented mask. It is also possible to generate the heatmap of a class from the probability matrix by considering the probabilities of each pixel/patch for that class. If the patches are already extracted in the test set like PCam, it is not possible to generate segmented masks or heatmaps. For all cases (the test set with full WSIs or already extracted patches), the probability of each patch can be processed for evaluation by considering each patch as a sample.

Evaluation. We consider both the single threshold-based (e.g., recall, precision) and multi-threshold-based (e.g., ROC, PR curves) metrics for the evaluation. For the single threshold-based metrics, we consider the argmax decision as the threshold of predicted probabilities. Those metrics are computed in two different ways for the segmentation task that we described in Section 3.3. For the threshold-free/multi-threshold-based metrics, we utilize the predicted probabilities.

3.2.2 Neural architecture and hyper-parameters

In this section we describe the hyper-parameter settings for the segmentation and classification tasks along with the selected network architectures.

3.2.2.1 Segmentation task

We selected U-net [Ronneberger et al., 2015] as our CNN architecture (see Figure 3.5) for the segmentation task as it allows to propagate large context information without losing localization accuracy. Moreover, it has been proven to be effective, even when using a limited number of training images, and only requires a moderate amount of time to train [Ronneberger et al., 2015]. We implemented the architecture using Keras [Chollet et al., 2015] on the TensorFlow backend. We used 8 GPUs Nvidia Geforce GTX 1080 TI (3584 Cuda cores each).

We used MLNTO to test our hypotheses in the segmentation setting. In all the experiments, we randomly selected 80% of the training data to train the model and the remaining 20% for validation. All data were normalized by

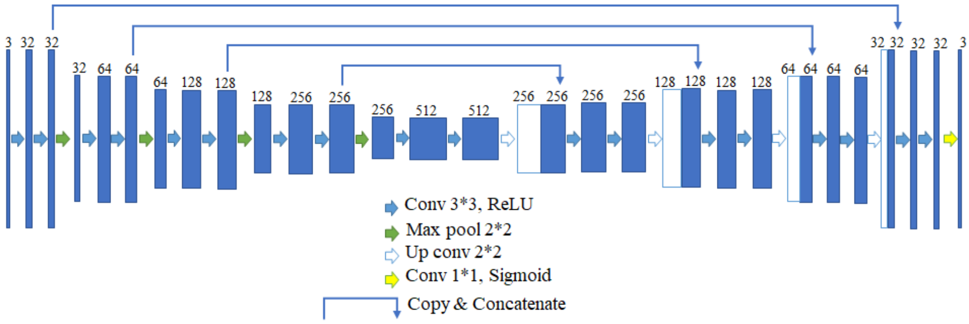


Figure 3.5: The U-net architecture: the first and last layers represent the input and the output of the model respectively, while the layers that come after an arrow indicate the output of the operation denoted by the arrow. The number on top of each layer indicates the number of feature channels.

scaling the pixel value from [0, 255] to [0.0, 1.0] dividing each value by 255. It makes the convergence of training faster [LeCun et al., 2012].

Each model was trained from scratch, i.e. without using transfer learning. After a preliminary empirical evaluation, we set the number of epochs (i.e. the number of times the algorithm passes through the entire training data) to 35 and the mini-batch size (i.e., the number of training examples the algorithm sees in one pass) to 5. We opted for the categorical cross-entropy as the loss function, similarly to the original U-net. To optimize the objective function, we used the Adam optimizer [Kingma and Ba, 2014]. After empirical preliminary evaluation, we set the learning rate -the amount of change of weights in the direction of the gradient for a mini-batch- of Adam as $1e^{-5}$. The initial weights were drawn randomly from the zero-mean Gaussian distribution as recommended in [Krizhevsky et al., 2012a]. We used a standard deviation of 0.05, which is the default setting in Keras.

3.2.2.2 Classification task

The CNNs are inherently translation equivariant, i.e. they can learn the same features from any particular location in an image. In WSIs, besides translation, the histological structures can be in any orientation. In other words, the rotation and reflection are common features in WSIs [Graham et al., 2020]. Thus, we selected the group equivariant convolutional networks (G-CNN) architecture [Cohen and Welling, 2016], a rotation and reflection equivariant architecture which was tuned for PCam data set by Veeling et al.

[2018]. The equivariant nature of G-CNNs is effective, and leads to state-of-the-art performance levels for WSI data [Veeling et al., 2018]. For the G-CNN architecture, we kept the default settings of all the hyper-parameters to reproduce the results (when we use full training set) achieved by Veeling et al. [2018], the only exception being the reduced mini-batch size. With our hardware configuration, the maximum possible mini-batch size while training a G-CNN is 8.

3.3 Patch-based evaluation method for segmentation task

A model evaluation is done by considering assigned label(s) to each sample by that model. In an image classification problem, a whole image is considered as a sample. The evaluation for the classification task is thus by default patch/image-based. In an image segmentation problem, by contrast, the sample is a pixel. Each pixel is thus considered during the evaluation of a segmentation-based model. A common measure used to evaluate segmentation models is the percentage of pixels correctly labeled [Everingham et al., 2010].

In our work, we are aiming at helping pathologists in detecting the region(s) with unhealthy tissue and filtering out the remaining (i.e., healthy ones that do not show any signs of cancer) in WSIs. In this context, it is not harmful if some pixels around a detected region are miss-classified due to the existing gaps between the actual ground truth and human annotations. Notably, having annotation gaps in the ground truth is acceptable for images whose size can be measured in gigapixels since it is impractical to consider every pixel during manual annotation by a human. Consequently, it becomes less relevant to consider the usual pixel-based evaluation for the segmentation task. Hence, we consider the patch-based evaluation for the segmentation task similar to the classification one. The approach is similar to the pathologist's way of checking a WSI.

In this section, we describe how we conduct patch-based evaluation for the segmentation task. We consider two different methods described as follows:

First method. We utilize the predicted mask for the full WSI. We divide a predicted mask and its corresponding ground-truth mask into regions that we call *evaluation patches*. The patches are not overlapping (unlike the training

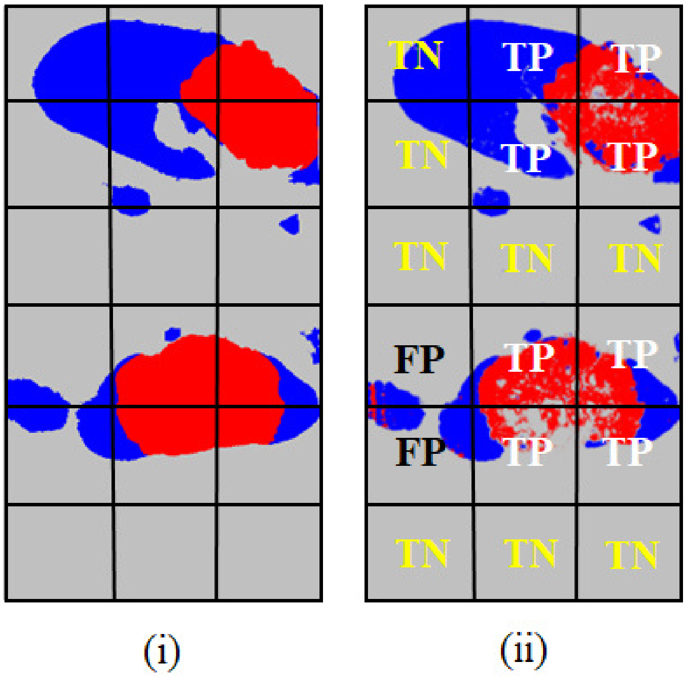


Figure 3.6: Illustration of the patch-based evaluation process when evaluating, for example, the class ℂ (red). Each image patch (i.e., evaluation patch) from the predicted output (ii) is compared to the ground truth (i). True-positive is when the model detects cancer (the class ℂ) in the patch, and the patch indeed contains cancer in the ground truth.

patches) and could be of any dimension from 1×1 pixel to the full image. When the patch dimension is 1x1, the evaluation measure is equivalent to a pixel-based evaluation.

In our case, the analogy is, when we evaluate the class ℂ, a patch in the predicted mask (see Figure 3.6 (ii)) is defined as TP if it contains class ℂ pixels (at least one pixel) and the corresponding patch in the ground truth mask (see Figure 3.6 (i)) also has class ℂ pixels, otherwise that patch is defined as FP. On the other hand, a patch in the predicted mask is defined as TN if it does not contain class ℂ pixels and the corresponding patch in the ground truth mask has no class ℂ pixel, otherwise that patch is defined as FN.

Figure 3.6 illustrates the way we generate the confusion matrix for the class ℂ (red) with this method, where (i) is a ground truth mask divided into patches, and (ii) is a predicted mask divided into the same corresponding patches of ground truth and annotated as TP, FP, TN. Since the predicted mask in (ii) has no false-negative, there is no annotation as FN. With the generated confusion matrix, we compute the evaluation metrics presented in the previous chapter (Section 2.5).

The size or dimension of a patch is a system hyper-parameter that have an impact on the results. We analyze the impact of the evaluation patch size in the next chapter. The pathologists' interest is on the in-between dimension; we thus set the default evaluation patch dimension as 500×500 in our result with this evaluation method.

Second method. We process the pixels-based predicted probabilities of each patch ($\mathbb{R}^{l \times l \times C}$) just after the prediction, instead of forming the full predicted mask then dividing it into evaluation patches. We convert the pixels-based predicted probabilities of each patch to patch-based probabilities ($\mathbb{R}^{C}$) thus making it resemble the prediction of a classification model. We take the class-wise maximum probability of the central region and assign it to the whole patch. In this way, the patch gets a probability for each class similar to the output of a classification model.

The first method is applicable to compute the single threshold-based metrics only since it does not consider the predicted probabilities. In contrast, the second method is applicable to compute both single threshold and multi-threshold based metrics. The argmax decision on the computed patch-based probabilities is considered as the predicted class label of that patch for computing the single threshold-based metrics, and the computed patch-based probabilities themselves are further considered to compute the multi-threshold-based metrics.

3.4 Conclusion

By reviewing different studies on cancer research from the literature, we figure out that the topic of hyper-parameters tuning is under-studied, especially, there is no dedicated work on class distribution analysis for WSI data. To fulfill the gap, we consider this topic in this thesis. For this purpose, we design a framework suitable for this task, where patch extraction and categorization are considered to facilitate the generation of different distributions for training. Additionally, we also propose two patch-based evaluation approaches, which are practical and similar to pathologists' observation of a WSI. This basic framework and evaluation approaches are utilized in the rest of the manuscript.

Chapter 4

Analyzing Class Distributions in WSI for Deep Learning

Contents

Abstract.

The class distribution of a training data set is an important factor which influences the performance of a deep learning-based system. In this chapter, we tackle the problem of finding the optimal class distribution of a training set to train an optimal model that detects cancer in WSIs. At first, we conduct a preliminary study. Based on the findings from that study, we formulate several hypotheses, which are then tested. The experiments have been designed to account for both segmentation and classification frameworks with various class distributions in the training set, including natural, balanced, over-represented cancer, and over-represented non-cancer.

Our experimental results show that the natural class distribution is better than the artificially generated balanced distribution. Moreover, we found that the over-representation of non-cancer classes compared to the cancer class reduces the number of samples which are falsely predicted as cancer (false-positive), although the result depends on some other hyper-parameters. Furthermore, we found that the least expensive to annotate non-ROI (non-Region-of-Interest) data can be useful in compensating for the performance loss in the system due to a shortage of expensive to annotate ROI data. The multi-label examples are more useful than the single-label ones to train a segmentation model. When the classification model is tuned with a balanced validation set, it is less affected than the segmentation model by the class distribution of the training set.

4.1 Introduction

The huge success of deep learning approaches, such as CNNs, in visual recognition [Krizhevsky et al., 2012a; LeCun et al., 2015; Rawat and Wang, 2017] has encouraged researchers to explore their use in various domains, including cancer detection from histological images or WSIs [Wang et al., 2016a; Bejnordi et al., 2017; Liu et al., 2017].

When working on a patient's case, the manual analysis of WSIs demands a high level of concentration and is time-consuming for pathologists. In this context, an automatic system can help by filtering out the healthy regions of the images and indicating possible (otherwise potentially overlooked) cancer regions. During the last few decades, many automatic systems based on machine learning techniques [Gurcan et al., 2009; Bejnordi et al., 2017; Litjens et al., 2017; Hu et al., 2018; Komura and Ishikawa, 2018; Bera et al., 2019; Srinidhi et al., 2020; Zhou et al., 2020; Li et al., 2021a] have been put forward for cancer detection. Some of them are reviewed in the previous chapter (Section 3.1). Among them, methods based on deep learning have recently become the most popular because of their impressive performance levels in vision tasks. These methods however are mainly focused on end-to-end pipeline developments for cancer detection, while the success of such systems depends on several hyper-parameters [Hinz et al., 2018; Bacanin et al., 2020].

We hypothesized that one of the important hyper-parameters is the class distribution of the training set, as the training set provides the supervision for all learning-based systems [Cracknell and Reading, 2014; Crawford, 2016; Deisenroth et al., 2020; Sarker, 2021]. Moreover, several studies have focused on class distribution analysis, in particular, the comparison between balanced and imbalanced distributions for different tasks, which illustrates its importance [Weiss and Provost, 2003; Batista et al., 2004; Prati et al., 2015; Buda et al., 2018; Thabtah et al., 2020]. Nonetheless, the results of different studies are inconsistent, depending on the tasks, data sets and the machine learning techniques employed [Prati et al., 2015]. This inconsistency casts doubt on their generalization for WSI data in the context of cancer detection, which is our topic of interest.

Generally, in machine learning, an imbalanced data distribution has been shown to lead to inferior models compared to a balanced distribution [Chawla et al., 2002; Khan et al., 2017], and hence, a lot of efforts have been put in to develop methods that overcome data imbalance by making the data artificially balanced. Some studies [Buda et al., 2018; Thabtah et al., 2020] have reviewed the popular methods, such as oversampling, undersampling, thresholding, cost sensitive learning, one-class classification, and various hybrids. These studies may indeed lay the path for balanced distribution to become the default choice as a deep learning state-of-the-art method [Bejnordi et al., 2017; Liu et al., 2017; Halicek et al., 2019], although Prati et al. [2015], for example, have shown that it is not optimal in all cases. Unfortunately, very few analytical studies on the performance impact of different distributions exist in the literature [Weiss and Provost, 2001, 2003; Prati et al., 2015; Thabtah et al., 2020]. Moreover, the available studies have been mainly conducted on toy data sets, even though real data sets may be very different and more complex. There is no evidence from the conclusions of these studies that they would be appropriate for cancer WSIs thus.

Furthermore, the outcomes of available studies are contradictory: some support an imbalanced distribution [Weiss and Provost, 2001, 2003; Thabtah et al., 2020] while others support a balanced distribution [Prati et al., 2015]. For this reason, it is not straightforward to decide on a specific class distribution for all types of tasks. Consequently, being a special kind of image type, domain-specific studies are required to answer the research questions: *Is the most adopted balanced distribution optimal for the WSI data for the cancer detection task? If not, which class the training set should be biased toward?*

State-of-the-art methods for cancer detection [Bejnordi et al., 2017; Liu et al., 2017; Halicek et al., 2019] utilize different types of distribution (usually balanced) and obtain very convincing results for a large-scale breast cancer data set using different models. In particular, the existing systems achieve very high sensitivity/recall [Wang et al., 2016a; Liu et al., 2017], while false-positives remain an ongoing issue [Pham et al., 2019]. However, to our knowledge, no analysis indicates if the commonly-adopted balanced distribution is the most appropriate distribution for cancer detection in WSIs nor which class should be over-represented. Moreover, it is also worth knowing which distribution produces fewer false-positives with high sensitivity and why. It would help in choosing training examples and their ratios for building robust training data sets.

In this chapter, we present an analysis that determines the performance impact of different class distributions on training data. In our analysis, at first, we answer to the research questions about balanced distribution through a preliminary study. Based on the outcome of that study, we derive several hypotheses concerning WSIs used for cancer detection and test them with two commonly used target applications: image segmentation and image classification. We choose a FCNN and a CNN architecture to test the hypotheses using segmentation and classification settings, respectively (see Section 3.2.2). We employ two data sets for training, MLNTO -which is a multi-class data set (see Section 2.4.1) and is applicable to the segmentation setting- and PCam -which is a binary-class data set (see Section 2.4.2) and is applicable to the classification setting. We conduct a series of experiments and analyze the results in detail to be able to provide comprehensive conclusions.

While the main focus of this chapter concerns training class distribution, we also discuss the case of test data distribution. We use the test set of CAMELYON16 data set (see Section 2.4.2) along with the other data sets.

The rest of the chapter is structured as follows. We first review the literature on the class imbalance problem and class distribution analysis in Section 4.2. In Section 4.3, we describe the patch categories in WSIs that we utilize in our analysis. We present a preliminary study on class distribution analysis in Section 4.4, while a deeper analysis is presented and discussed in Section 4.5 which addresses the limitations of the preliminary study. In Section 4.6 we draw our conclusions and discuss future directions.

4.2 Related work

The natural imbalanced distribution of data is generally considered as a problem/obstacle in machine learning. For example, Crawford [2016] identified an unfair AI in predictive policing due to data imbalance. Researchers put a lot of effort into balancing data. In contrast, comparative analytical studies of different distributions are few in number. In this section we investigate several studies related to class imbalance problems and class distribution analysis.

4.2.1 Studies on class imbalance problems

One of the common problems in machine learning is dealing with class-biased or imbalanced data. In the real world, the availability of some classes makes them an over-represented majority, while the scarcity of other classes makes them an under-represented minority. This imbalance usually makes the classification task challenging for a classifier. There are many studies [Kubat et al., 1997; Chawla et al., 2002, 2003; Sun et al., 2007; Masko and Hensman, 2015; Levi and Hassner, 2015; Wang et al., 2016b; Khan et al., 2017; Jaccard et al., 2017; Buda et al., 2018; Yuan et al., 2018; Afzal et al., 2019; Wu et al., 2020; Zhang et al., 2020; Abou Elassad et al., 2020] showing that imbalanced training data leads to a loss in performance, so various methods have to be adopted to make the training data balanced. Prati et al. [2015] and Thabtah et al. [2020] listed some of the most popular methods. Most studies were conducted on classical machine learning methods, and only a few discuss the deep learning perspective [Buda et al., 2018; Johnson and Khoshgoftaar, 2019]. Among the studies referring to this perspective, some suggested data-level modifications (e.g., oversampling of the minority class, undersampling of the majority class) [Levi and Hassner, 2015; Jaccard et al., 2017; Afzal et al., 2019; Hamad et al., 2020; Zhang et al., 2020], some preferred tweaking different hyperparameters of the network or making algorithm-level modifications (e.g. thresholding, cost sensitive learning) [Wang et al., 2016b; Khan et al., 2017; Baloch et al., 2019], while others suggested combining data and algorithm-level modifications [Chawla et al., 2003; Havaei et al., 2017; Rendón et al., 2020]. Buda et al. [2018] performed a comparative study of different methods to address the class-biased problem. According to the authors, imbalanced data had an adversarial effect on the classification accuracy of the CNNs, similarly to classical machine learning techniques. The most recommended solution was to oversample the minority class [Masko and Hensman, 2015; Buda et al., 2018]. Johnson and Khoshgoftaar [2019] conducted another detailed survey on recent techniques used to deal with the imbalanced data problem in deep learning. They concluded that not enough evidence existed to suggest that a particular technique was superior in dealing with class imbalance through the use of deep learning.

4.2.2 Studies on class distribution analysis

All the articles from the previous subsection described the adversarial effects of imbalanced data, and suggested methods for making the class distribution of the training data balanced. However, these methods did not provide the answer to the important question of *whether the balanced distribution is optimal for all types of learning techniques and data sets.* To answer this question, comparative study on different class distributions of training data is required. Such studies are few in number [Thabtah et al., 2020]. Among the available comparative studies, Weiss and Provost [2001, 2003] showed that neither a naturally occurring class distribution nor a balanced distribution was the best for learning, and often a substantially better performance could be obtained by using a different class distribution. For their analysis they employed 26 data sets from the University of California Irvine (UCI) repository [Dheeru and Karra Taniskidou, 2017]. According to the analysis by Prati et al. [2015] on 20 data sets from UCI [Frank and Asuncion, 2010] and a small number of private data sets, the best distribution for seven different learning algorithms, including neural networks, was a balanced distribution. They concluded that only SVMs were less affected by class imbalance. However, the study was carried out on toy data sets, and hence there is no evidence that the commonly prescribed balanced distribution is a generalizable solution for more complex real data sets. Consequently, a separate analysis is required for each special kind of data. This conclusion is clearly connected to the No Free Lunch Theorem [Wolpert, 1996], which states that there is no single model that works best for every task. For example, Zhu et al. [2016] conducted an elaborate domain specific study to optimize the training data distribution for land cover data in a bid to detect change. According to the authors, a class distribution proportional to the naturally occurring distribution was superior to a balanced distribution for land cover data when using random forests as the learning algorithm. Thabtah et al. [2020] did an analysis of class distribution on the autism spectrum disorder (ASD) diagnosis problem with the probabilistic Naive Bayes as the base classifier. Using five data sets from the UCI repository, they showed that for the majority of data sets, the evaluation metrics are at their minimum values when the data sets are balanced (50%:50%). The highest evaluation metric values are derived when the data set is imbalanced, specifically, 10%:90% or 90%:10%. To our knowledge, there is no such comparative study for cancer detection in WSIs except ours one presented in [Reshma et al., 2021]. In our study, we discussed the impact of non-ROIs and natural distribution where the conclusion was in favor of non-ROIs biased data set (see Section 4.5.3.1.1).

4.3 WSI patch categories

The main material of this analysis is WSI (see Section 2.1). Due to typical memory limitations, rather than using the whole images as training examples, it is common practice to use patches extracted from WSIs. The patch extraction process is described in our methodological framework (see Section 3.2.1). For our analysis, the extracted patches need to be placed in different categories to create different distributions in training sets. We consider different categories of patches based on the annotation of the WSIs in the data set. This section describes how we categorize the patches in different categories, both for multi-class and binary-class data sets.

For a *multi-class WSI data set*, MLNTO (presented in Section 2.4.1), we consider four categories of patches as defined in Table 4.1. Patches that contain more than 99.999% class $\mathbb{O}$ pixels are labeled as the *other* ($\mathbb{O}$) category. The

Table 4.1: **Definition** of Patch Categories

Data set type	Patch category	Definition	Label type	Comment
Multi-class	*Other* ($\mathbb{O}$)	Containing $> 99.999\%$ $\mathbb{O}$ pixels	Single-label	Non-ROI/non-lymph node as the negative non-ROI class
	Cancer ($\mathbb{C}$)	Containing $> 0.001\%$ $\mathbb{C}$ pixels and no $\neg\mathbb{C}$ pixels.	Single-label	Metastasis as the positive ROI class
	Non-cancer ($\neg\mathbb{C}$)	Containing $> 0.001\%$ $\neg\mathbb{C}$ pixels and no $\mathbb{C}$ pixels.	Single-label	Lymph node without metastasis as the negative ROI class
.	*Mixed* ($\mathbb{C}\&\neg\mathbb{C}$)	Containing $> 0.001\%$ $\mathbb{C}$ pixels and $> 0.001\%$ $\neg\mathbb{C}$ pixels.	Multi-label	Belongs to both $\mathbb{C}$ and $\neg\mathbb{C}$ class at the same time
Binary class	*Cancer* ($\mathbb{C}$)	Containing $\mathbb{C}$ pixels at a particular center region of the patch.	Single-label	Metastasis as the positive class
	Non-cancer ($\neg\mathbb{C}$)	Containing $\neg\mathbb{C}$ or $\mathbb{O}$ pixels and no $\mathbb{C}$ pixels at the central region.	Single-label	Lymph node or non-lymph node (unlike multi-class data set case) without metastasis as the negative class

remaining patches belong to the other categories: *cancer* ($\mathbb{C}$), *non-cancer* ($\neg\mathbb{C}$), or *mixed* ($\mathbb{C}\&\neg\mathbb{C}$) with an optional presence of class $\mathbb{O}$ pixels ($< 99.999\%$). The category $\mathbb{C}\&\neg\mathbb{C}$ is multi-labeled (since it contains two ROI classes at the same time), and the others are single-labeled. We use the same notations for both patch categories and pixel classes.

When considering the MLNTO data set and 384 square-sized overlapping patch extraction described in pre-processing step of our methodological framework (Section 3.2.1), a total of 127,898 patches were extracted from the training set. They were indeed pairs, the patch from WSI and its corresponding annotated mask from the ground truth. On average, about 3,761 patches were extracted from each training WSI.

Table 4.2: The number of patches belonging to each category in the MLNTO training set when WSIs are downsampled by a factor of 8 and the stride is $l/2$ (i.e., 192).

patch category	#patches
Background/other ($\mathbb{O}$)	$90,374$
Metastasis/cancer ($\mathbb{C}$)	$15,328$
Lymph node/non-cancer ($\neg\mathbb{C}$)	$17,274$
Mixed ($\mathbb{C}\&\neg\mathbb{C}$)	$4,922$

The number of patches in each category is given in Table 4.2 where we can see that the category $\mathbb{O}$ (*other*) patches are much higher than other categories.

For the *binary class WSI data set*, PCam, the patches are already categorized by its creator. For the CAMELYON16 data set, we follow the same approach of categorization as for PCam: the patches containing $\mathbb{C}$ pixels at a pre-defined center regions are labeled as $\mathbb{C}$, otherwise as $\neg\mathbb{C}$ category. The numbers of patches from binary data sets are given in the corresponding data set description (see Section 2.4.2).

We use the categories described above to design several experiments with the aim of answering the research questions (Section 4.4) and testing various hypotheses concerning the class distribution of the training set (Section 4.5).

4.4 Preliminary studies

We design and conduct several experiments with different distributions, including balanced, natural, $\mathbb{C}$-biased, and $\neg\mathbb{C}$-biased distributions and perform a data-driven comparative study. All the experiments are design based on the MLNTO data set and segmentation task. Moreover, our preliminary study shows the impact of different evaluation patch dimensions.

Our design techniques of balanced and class-biased training are described in the following section.

4.4.1 Class distributions for training sets

To analyze the impact of class-biased (both naturally and artificially obtained) and balanced (artificially obtained) training on the cancer detection task with the MLNTO data set, we design several experiments. In a given experiment, we first select the patches categories presented in Table 4.2 as defined in the corresponding experiments and then use the patches for training.

In the baseline experiment (*All*), we use all the extracted patches from the training set (Table 4.2), its class distribution thus is "natural" and class $\mathbb{O}$-biased. The second experiment ($\mathbb{C}\&\neg\mathbb{C}$) focuses on patches from the category that mixes $\mathbb{C}$ and $\neg\mathbb{C}$, while some class $\mathbb{O}$ pixels can also be present in the corresponding patches. The class distribution of the second experiment could be considered as a "balanced" distribution since here all the training examples contain all three classes. In the next experiment ($\mathbb{C}, \mathbb{C}\&\neg\mathbb{C}$), we increased the number of examples from the class $\mathbb{C}$, thus creating class "$\mathbb{C}$-biased" training set artificially. We hypothesize that the model resulting from this training will produce higher recall for class $\mathbb{C}$, and at the same time, over-representation of this class will produce more false-positive, i.e., lower precision for the $\mathbb{C}$ class. Reversely, it prevents the false-positive generation of the class $\neg\mathbb{C}$ thus provides higher precision for the class $\neg\mathbb{C}$. The next experiment ($\neg\mathbb{C}, \mathbb{C}\&\neg\mathbb{C}$) is the twin case of the previous one, where now we select patches from categories $\neg\mathbb{C}$ thus creating class "$\neg\mathbb{C}$-biased" training set artificially. We hypothesize that the model will produce twin results compared to the previous model. Finally, experiment ($\mathbb{C}$, $\neg\mathbb{C}$, $\mathbb{C}\&\neg\mathbb{C}$) aims at studying another "balanced" case, where all the patches that contain class $\mathbb{C}$ and $\neg\mathbb{C}$ are included. The description of the experiments are as follows and summarized in Table 4.3:

- *(All)*: it includes all possible patches, and the distribution of the training set is natural since we did not modify the original data distribution here.

- ($\mathbb{C}\&\neg\mathbb{C}$): this experiment is done with patches from the $\mathbb{C}\&\neg\mathbb{C}$ category. The three classes are balanced in terms of pixels however, the number of training examples is small (4,922), we thus named it *balanced_S*.

- ($\mathbb{C}, \mathbb{C}\&\neg\mathbb{C}$): patches are from the $\mathbb{C}$ and $\mathbb{C}\&\neg\mathbb{C}$ categories. The majority of the pixels are class $\mathbb{C}$. By excluding the $\neg\mathbb{C}$ category, here, we limit the presence of class $\neg\mathbb{C}$ and the training set is $\mathbb{C}$-biased.

- ($\neg\mathbb{C}, \mathbb{C}\&\neg\mathbb{C}$): patches are from the $\neg\mathbb{C}$ and $\mathbb{C}\&\neg\mathbb{C}$ categories. This is the twin case of ($\mathbb{C}, \mathbb{C}\&\neg\mathbb{C}$) for class $\neg\mathbb{C}$: the training set is $\neg\mathbb{C}$-biased.
- ($\mathbb{C}, \neg\mathbb{C}, \mathbb{C}\&\neg\mathbb{C}$): all the patches from the categories $\mathbb{C}$, $\neg\mathbb{C}$, and $\mathbb{C}\&\neg\mathbb{C}$ are used. Here, the class $\mathbb{C}$ and $\neg\mathbb{C}$ pixels are almost balanced (see Figure 2.9 (i) and Table 4.2). Class $\mathbb{O}$ pixels however are downsampled in comparison to the experiment (*All*) to make all three class pixels almost balanced. Note that this experiment differs from the experiment (*All*) by the absence of patches belonging to category $\mathbb{O}$. Considering the large number of training examples, we named it *balanced_L*.

Table 4.3: Experiments with the corresponding class distribution and size of the training set (the numbers of 3^{rd} column are from Table 4.2)

Experiment	Class distribution	Size of the training set ($\mathbb{O} + \mathbb{C} + \neg\mathbb{C} + \mathbb{C}\&\neg\mathbb{C}$)
($\mathbb{C}\&\neg\mathbb{C}$)	Balanced_S	$0 + 0 + 0 + 4,922$
($\mathbb{C}$, $\mathbb{C}\&\neg\mathbb{C}$)	$\mathbb{C}$-biased	$0 + 15,328 + 0 + 4,922$
($\neg\mathbb{C}$, $\mathbb{C}\&\neg\mathbb{C}$)	$\neg\mathbb{C}$-biased	$0 + 0 + 17,274 + 4,922$
($\mathbb{C}$, $\neg\mathbb{C}$, $\mathbb{C}\&\neg\mathbb{C}$)	balanced_L	$0 + 15,328 + 17,274 + 4,922$
All	Natural	$90,374 + 15,328 + 17,274 + 4,922$

4.4.2 Findings and discussion

Pathologists are interested in high recall in class $\mathbb{C}$ (they do not want to miss any cancer) and high precision in class $\neg\mathbb{C}$ (they want to filter out only the non-cancer regions but do not want to filter out an image that contains cancer by mistake). For this reason, we evaluated our model for both the classes $\mathbb{C}$ and $\neg\mathbb{C}$ separately. The results we reported were computed on our first method of patch-based evaluation approaches and then averaged over the 27 test WSIs of the MLNTO data set. The evaluation patch dimension is 500×500 pixels as an in-between dimension of pixel-based to the full image.

4.4.2.1 Balanced distribution is not generalizable to WSI data

In Figure 4.1, we report the results for both class $\mathbb{C}$ (top row) and class $\neg\mathbb{C}$ (bottom row). The x-axis of the figure reports different distributions presented

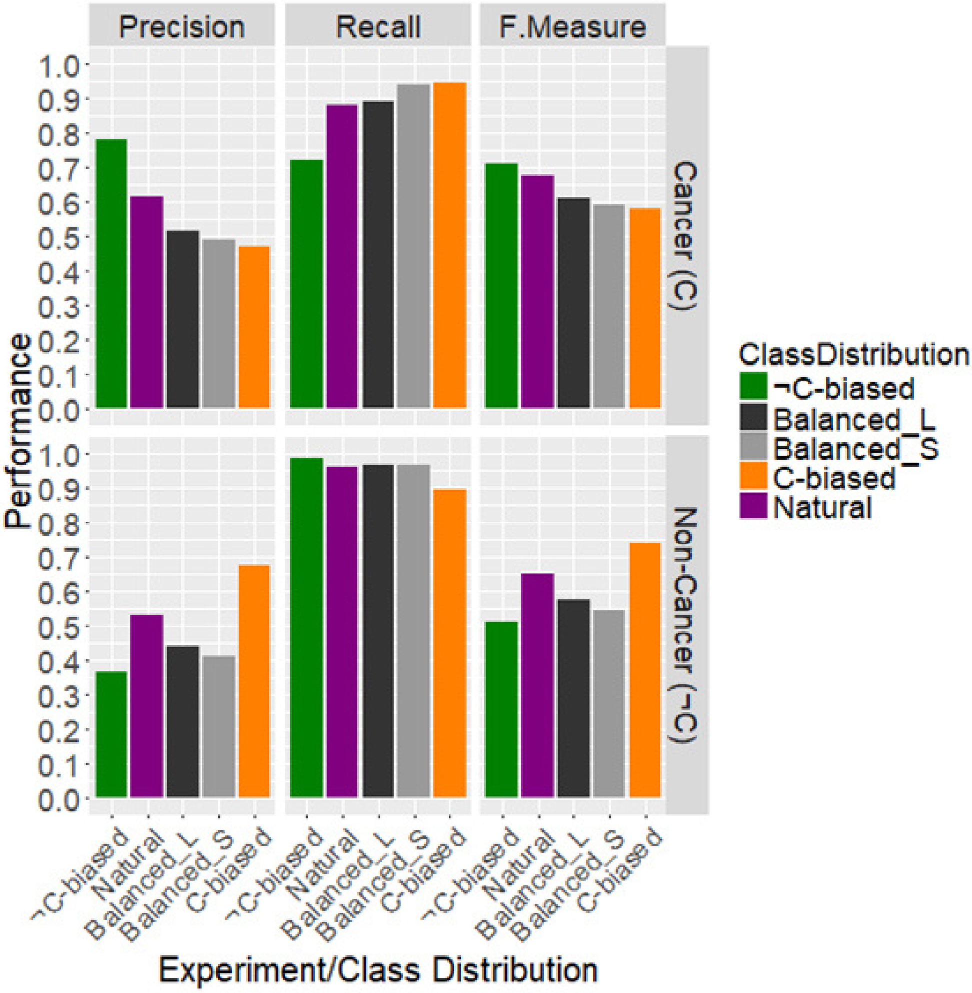


Figure 4.1: ℂ**(resp. ¬**ℂ**)-biased training produced the highest recall (resp. precision) for class** ℂ**.** Model performance for class ℂ (top row) and ¬ℂ (bottom row) when using different combinations of the patch categories as a training set, i.e., different distributions of the classes in the training set. Here, the x-axis presents class distributions used in different experiments (see Table 4.3) and the y-axis presents the performance of those experiments based on precision, recall, and F-measure. The experiments are arranged according to the ascending order of the recall on class ℂ.

in Table 4.3, and the y-axis is either precision, recall, or F-measure. The height of the bar thus represents the performance of the corresponding experiment with a particular class distribution. Experiments are ordered according to the ascending order of recall on class ℂ. From Figure 4.1, we can see that recall is higher than precision for both classes, which implies that in this domain, most of the errors come from false-positives rather than false-negatives.

Is the most adopted balanced distribution optimal for the WSI data for the cancer detection task?
Unlike it has been reported in [Prati et al., 2015] for 20 different data sets from UCI machine learning repository [Frank and Asuncion, 2010], Statlog [Michie et al., 1995], and some private data sets, here, balanced distribution (Figure 4.1: black and gray bars), i.e., experiments ($\mathbb{C}, \neg\mathbb{C}, \mathbb{C}\&\neg\mathbb{C}$) and ($\mathbb{C}\&\neg\mathbb{C}$) does not produce the best result; thus balanced distribution is not optimal for the WSI data for the cancer detection task.

If not, which class the training set should be biased toward?
The natural distribution (biased to class $\mathbb{O}$), is the best distribution (Figure 4.1: purple); it produces reasonable recall and precision for both classes $\mathbb{C}$ and $\neg\mathbb{C}$ at the same time. Since, our main objective is to help pathologists in all cancer locations detection with the less false-positives generation as possible, the distribution producing the best results on cancer class $\mathbb{C}$ is the most desirable.

For class $\mathbb{C}$, the best precision is for $\neg\mathbb{C}$-biased training set (Figure 4.1: top row, green), i.e., the distribution produces the least false-positives, while the best recall is for $\mathbb{C}$-biased training set (Figure 4.1: top row, orange), i.e., the distribution produces the least false-negatives. While considering both recall and precision at the same time, i.e., F-measure, however, $\neg\mathbb{C}$-biased distribution is the best distribution, and $\mathbb{C}$-biased distribution is the worst distribution for class $\mathbb{C}$. According to this result, we can hypothesize that some $\neg\mathbb{C}$ regions look like class $\mathbb{C}$ regions, i.e., there are some inter-class similar regions. That could be the reason why the absence of enough $\neg\mathbb{C}$ examples compared to class $\mathbb{C}$ examples in the training set (i.e., experiment ($\mathbb{C}, \mathbb{C}\&\neg\mathbb{C}$)) causes false-positive for class $\mathbb{C}$ during the test.

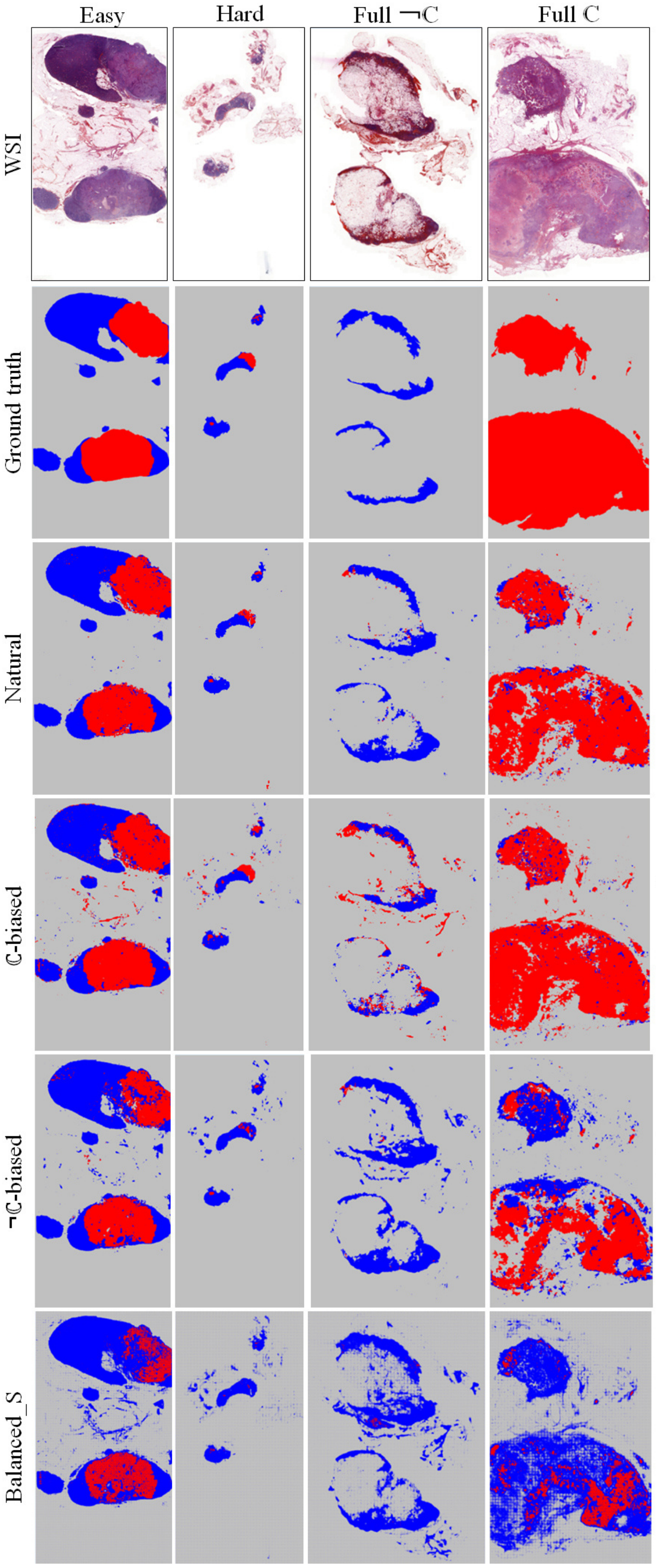
When considering the class $\neg\mathbb{C}$, this is the other way around: the best precision is for $\mathbb{C}$-biased training set (Figure 4.1: bottom row, orange), while the best recall is for $\neg\mathbb{C}$-biased training set (Figure 4.1: bottom row, green). While considering both recall and precision at the same time, i.e., F-measure, however, $\mathbb{C}$-biased distribution is the best distribution for class $\neg\mathbb{C}$.

In a nutshell, for cancer class $\mathbb{C}$, $\mathbb{C}$(resp. $\neg\mathbb{C}$)-biased training makes recall (resp. precision) higher. For non-cancer class $\neg\mathbb{C}$, the result is the opposite. The balanced training causes an average result for both classes and the natural distribution, i.e., training with the original distribution of the training set leads to the best trade-off in recall and precision for both classes at the same time. The class $\mathbb{O}$ is predicted well whatever the experiment is. Detailed results are presented in Table 4.4.

Table 4.4: **Average results computed from the results of 27 test WSIs for the different experiments.** Here, R, P, F means the recall, precision, and F-measure, respectively.

Experiment	Distribution	Class $\mathbb{C}$	Class $\neg\mathbb{C}$	Class $\mathbb{O}$	Comment
(*All*)	Natural	R: .88 P: .61 F: .68	R: .96 P: .53 F: .65	R: 1.0 P: .93 F: .96	Best trade-off
($\mathbb{C}$, $\mathbb{C}\&\neg\mathbb{C}$)	$\mathbb{C}$-biased	R: **.943** P: .47 F: .58	R: .89 P: **.68** F: **.74**	R: 1.0 P: .94 F: .97	Best R for $\mathbb{C}$
($\neg\mathbb{C}$, $\mathbb{C}\&\neg\mathbb{C}$)	$\neg\mathbb{C}$-biased	R: .72 P: **.78** F: **.71**	R: **.98** P: .36 F: .51	R: 1.0 P: .93 F: .96	Best P and F for $\mathbb{C}$
($\mathbb{C}\&\neg\mathbb{C}$)	Balanced_S	R: **.939** P: .49 F: .59	R: .96 P: .41 F: .55	R: 1.0 P: .95 F: .97	Average for $\mathbb{C}$
($\mathbb{C}$, $\neg\mathbb{C}$, $\mathbb{C}\&\neg\mathbb{C}$)	Balanced_L	R: .89 P: .52 F: .61	R: .97 P: .44 F: .57	R: 1.0 P: .93 F: .96	Average for $\mathbb{C}$

In Figure 4.2, we present some example predictions from four cases, including easy, hard, full $\neg\mathbb{C}$, and full $\mathbb{C}$ by the models trained with four differently distributed data. The remarks on each distribution are given on the right side of the same row on the figure. According to these examples, again the experiment with natural distribution gives the best result because of its highest number of examples from all three classes. The experiment with the balanced_S distribution gives the worst result because of its lowest number of examples from all three classes. On the other hand, the $\mathbb{C}$-biased and $\neg\mathbb{C}$-biased experiments cause miss-classification of the class $\mathbb{O}$ pixels due to the downsampling of the class $\mathbb{O}$ pixels from the training set. From these example predictions, it is clear that all the experiments can correctly locate cancer (although not all pixels are identified and it is not important if all the



Identifies both $\mathbb{C}$ and $\neg\mathbb{C}$ with less FP

Over predicts $\mathbb{C}$ with the most FP when it should predict $\mathbb{O}$ pixels but predict $\neg\mathbb{C}$ with the least FP

Identifies all $\mathbb{C}$ locations with less FP but over predicts $\neg\mathbb{C}$ with some FP when it should predict $\mathbb{O}$ pixels

Under predict $\mathbb{C}$ with less FP and over predicts $\neg\mathbb{C}$ with the most FP when it should predict either $\mathbb{C}$ or $\mathbb{O}$ pixels

Figure 4.2: **Natural distribution maintains the best trade-off in predicting class $\mathbb{C}$ and $\neg\mathbb{C}$.** Predictions considering four examples (Easy, Hard, Full $\neg\mathbb{C}$ (healthy), and Full $\mathbb{C}$ (cancer)). Remarks on each distribution are given on the right side of the same row.

cancer locations are correctly identified) however, produce a lot of FP at the same time. The FP reduction is thus the major challenge in this task.

For cancer class $\mathbb{C}$, an interesting result is found from the experiment with the smallest training set, i.e., from the experiment with the balanced_S distribution (see Table 4.3). In Figure 4.1 top row, while comparing the gray and orange bars, both bars are comparable. Especially the recall are almost the same for both experiments, and they are the best. Since the smallest training set provides one of the best recall, while it is not the case for precision, it can be claimed that getting a high recall is cheaper than getting high precision. In other words, reducing false-positives is more difficult than reducing the false-negatives.

4.4.2.2 Impact of the evaluation patch size

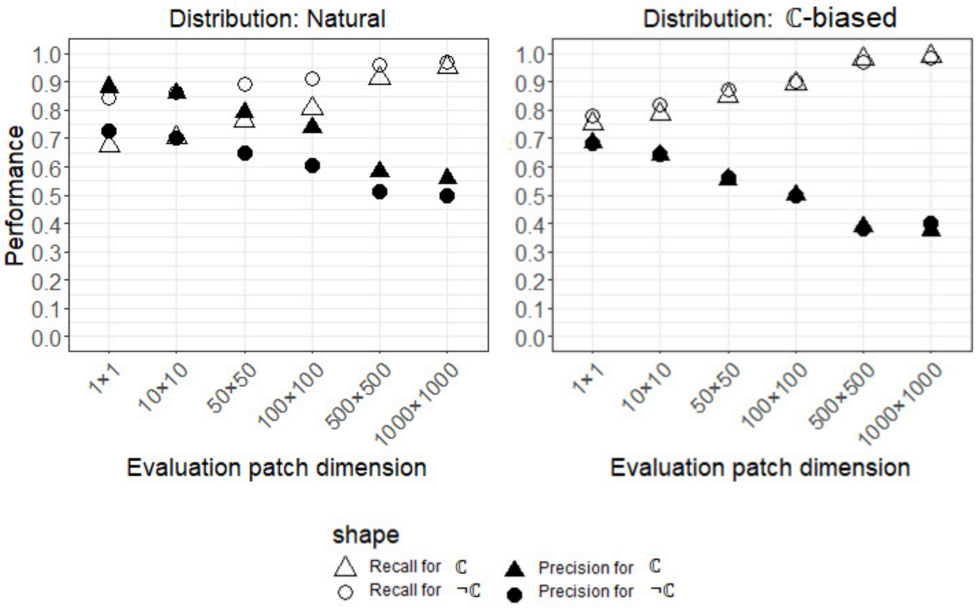


Figure 4.3: **Performance of the model at different *evaluation patch* dimension.** The result is presented for two experiments, the experiment with natural (left) and $\mathbb{C}$-biased (right) distributions.

We also measure the impact of different dimensions of the evaluation patch (Figure 4.3). We considered experiments with natural (Figure 4.3: left) and $\mathbb{C}$-biased (Figure 4.3: right) distributions. At the lowest level of dimension (1×1 pixel), this is the pixel level, which means that the model should determine for each pixel if the pixel is class $\mathbb{C}$, $\neg\mathbb{C}$ or $\mathbb{O}$. At the highest level of dimension, this is the image level (full image), and the model should decide whether or not the entire WSI contains class $\mathbb{C}$. For a pathologist, the most interesting level is in-between. That is why, we decided to consider several dimensions of patches as follows: 1×1, 10×10, 50×50, 100×100, 500×500, and 1000×1000 pixels. We did not consider the full-image case since the MLNTO data set has only one WSI without class $\mathbb{C}$, and three WSIs without class $\neg\mathbb{C}$. There is thus very limited scope to produce FP -which cases very high precision- for both classes in the full-image level of evaluation.

For the experiment with natural distribution (Figure 4.3: left), we can see that for both $\mathbb{C}$ and $\neg\mathbb{C}$ classes, recall increases when the patch dimension increases, while at the same time the precision decreases.

For the experiment with $\mathbb{C}$-biased distribution, (Figure 4.3: right), we can see that the results for both $\mathbb{C}$ and $\neg\mathbb{C}$ classes are very similar. The same trend as in the experiment with the natural distribution can be observed (increase of recall and decrease of precision when the patch dimension increases).

In summary, at the low dimension of the evaluation patch, the performance is average in terms of both recall and precision, while at the higher dimensions of the evaluation patch, the performance gets higher in terms of recall and lower in terms of precision. Both cases of dimension do not help end-users to get a proper view of model performances. The in-between dimension is thus preferable, which is also consistent with pathologists' preference.

4.4.3 Limitations

Although this analysis gives a preliminary flavor of the behavior of the model towards the different distribution of classes in the training set, it demands deeper analysis. Specifically, here, the numbers of training examples were not the same for all experiments; we tested the class distribution for a fixed set of network parameters; the evaluation was done on single threshold-based metrics only. The next section (Section 4.5) reports further analysis that solves the above-mentioned issues.

4.5 Analyzing class distributions: hypothesis-driven approach

From our preliminary experiments, we found that the balanced set does not provide the best performance, and FP reduction is the major challenge in cancer detection from WSIs. The results encouraged us to draw certain hypotheses and investigate the results at length. These hypotheses led to the design of a new generic framework that can be applied to any new task to provide an optimal choice of class distribution in the training set for a machine learning model.

4.5.1 Methodological framework

Figure 4.4 illustrates the methodological framework for this analytical study. There are two major steps in this framework: 1) identifying the research questions/hypotheses and 2) testing those hypotheses. For each hypothesis, to answer the general question of what the optimal class distribution of the training set should be, we created d training sets of d different distributions with n patches in each set. We then trained d models with the created training sets. The models were tested and evaluated on the same unseen test set. We repeated the process 10 times for each hypothesis and calculated the mean to be able to make the final decision.

The details about hypotheses are described in Section 4.5.1.1 and the pipeline to test those hypotheses is presented in Section 4.5.1.2.

4.5.1.1 Hypotheses and relevant class distributions

We make several hypotheses and design several experiments with the relevant class distributions to be able to test the proposed hypotheses.

While testing one hypothesis, the total number of patches in the training set (n) of each experiment is kept the same to ensure fair comparison. The experiments within a group of experiments differ from one another according to the patch categories that make up the training set, and according to their ratio, i.e. the distribution of the classes in the training set. The hypotheses are presented below.

H1: Natural distribution is not optimal for training

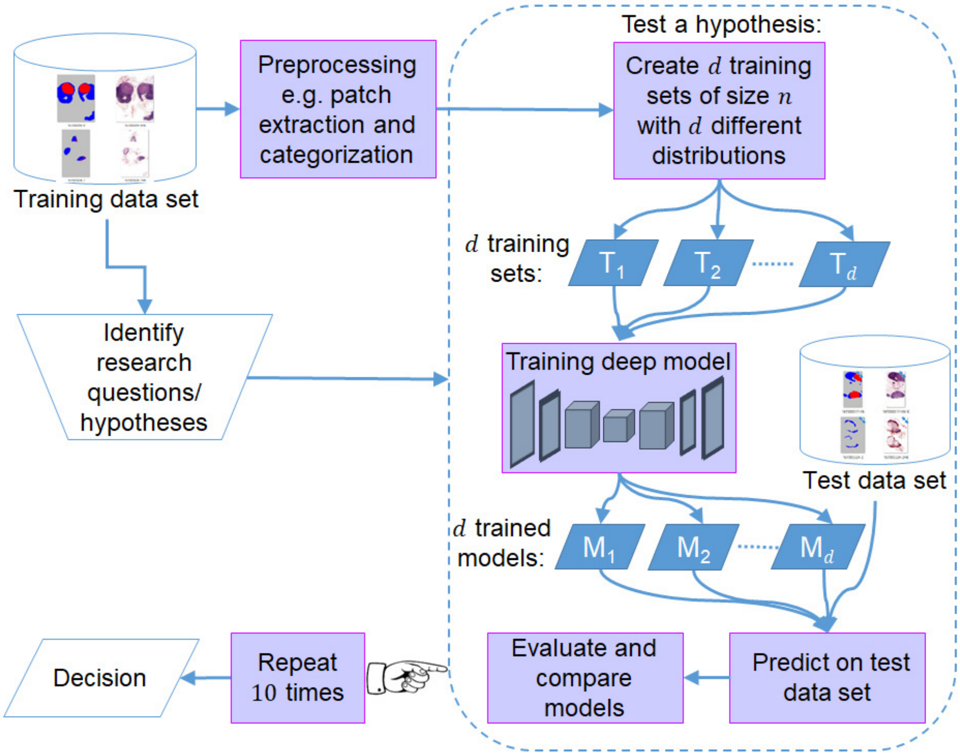

Training data set
Preprocessing e.g. patch extraction and categorization
Test a hypothesis:
Create d training sets of size n with d different distributions
d training sets:
T_1
T_2
T_d
Identify research questions/ hypotheses
Training deep model
Test data set
d trained models:
M_1
M_2
M_d
Decision
Repeat 10 times
Evaluate and compare models
Predict on test data set

Figure 4.4: Our methodological framework. To answer the general question of what the optimal class distribution of the training set should be, we created d training sets of d different distributions with n patches in each set. We then trained d models with the created training sets. The models were tested and evaluated on the same unseen test set. We repeated the process 10 times for each hypothesis and calculated the mean to be able to make the final decision.

Table 4.5: **Experiment settings E1**: E1 settings are designed to test H1 (natural distribution is not optimal) with a total of 9 units ($\mathbb{U}$) of patches in the training set of each experiment.

Experiment ID	Distribution	Patch ratio ($\mathbb{O} : \mathbb{C} : \neg\mathbb{C}$)
E1.a	Balanced	$3:3:3$
E1.b	Over-represented $\mathbb{O}$ (natural)	$7:1:1$

The WSI data are naturally biased towards the non-ROI class, $\mathbb{O}$. Class $\mathbb{O}$ is not a region of interest for pathologists. It is a common practice to filter out the excessive examples of class $\mathbb{O}$ and make a balanced training set out of the remaining classes.

We hypothesize that the trained model with over-represented class $\mathbb{O}$ (i.e. natural distribution) will be effective at detecting the regions that are not of interest (non-ROIs) to pathologists, since it will be trained using a large number of different non-ROI cases. However, we also hypothesize that this distribution will be less effective for the less frequently occurring ROI cases, although these cases are much more interesting for pathologists.

To test H1, we designed two experiments: E1.a and E1.b (i.e., $d = 2$). In E1.a, we consider the same number of patches in each of the three classes, whereas in E1.b the training examples are highly biased (7 times) towards class $\mathbb{O}$ similar to the natural distribution (see Table 4.5). We hereby introduce $\mathbb{U}$ to denote a unit, a fixed amount of patches. The size of $\mathbb{U}$ is determined by the expected patch ratio and the total number of extracted patches in the smallest patch category for a particular data set so that we can consider both the under and over-representation of that smallest category. We consider only single-label patch categories. For example, let us suppose that $\mathbb{C}$ is the smallest category with size N (total number of extracted patches) and one of the expected patch ratios for $\mathbb{C}$ and $\neg\mathbb{C}$ is $3 : 1$, then $\mathbb{U} = N/3$. To test H1, a total of $9\mathbb{U}$ of patches is used to create a natural, balanced distribution (i.e. $n = 9\mathbb{U}$).

This hypothesis can be tested on the multi-class data set only. Since class $\mathbb{O}$ is not annotated separately in binary class data sets, it is not possible to accurately separate class $\mathbb{O}$ from class $\neg\mathbb{C}$, especially if histological structures are available in $\mathbb{O}$ regions.

H2: Over-representing the $\neg\mathbb{C}$ class in the training set reduces false-positives during cancer detection.

We observed that in WSIs the regions for class $\neg\mathbb{C}$ are more heterogeneous than the regions for class $\mathbb{C}$. Indeed, class $\neg\mathbb{C}$ regions may contain germinal centers, macrophage, blood vessels or artifacts (e.g. blur areas), etc. (see Figure 4.5). Some of these are visually similar to class $\mathbb{C}$ regions. To ensure the coverage of different variations and to reduce any confusion with class $\mathbb{C}$ regions, the over-representation of class $\neg\mathbb{C}$ could be useful. Thus, over-representing class $\neg\mathbb{C}$ compared to class $\mathbb{C}$ should reduce the false-positive rate for class $\mathbb{C}$.

To test this hypothesis, we designed three experiments (i.e. $d = 3$), as presented in Table 4.6. In E2.a, we consider the balanced case between $\mathbb{C}$ and

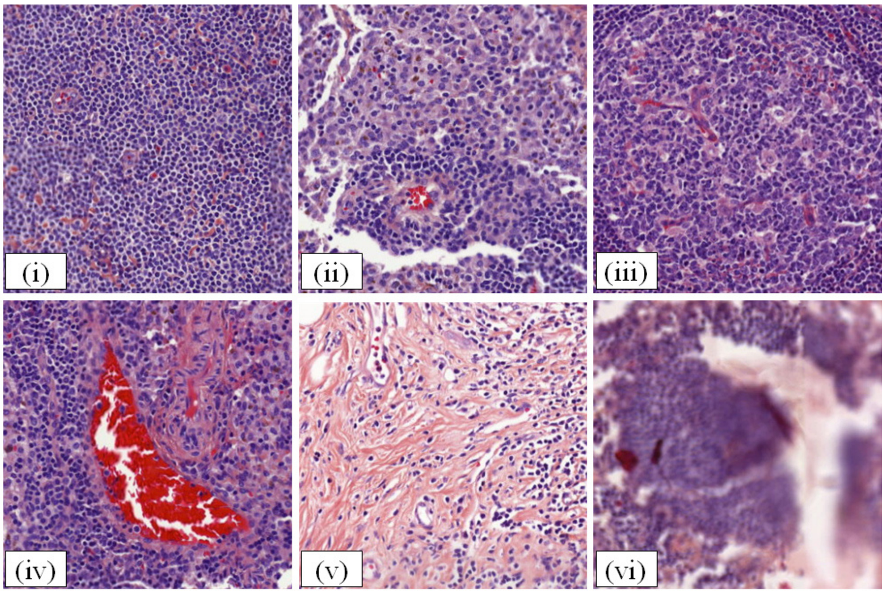


Figure 4.5: Samples of heterogeneous patches of class ¬ℂ (lymph node): (i) usual lymphocytes in lymph nodes, (ii) sinusal macrophages in lymph nodes, (iii) germinal center, (iv) blood vessel, (v) fibrosis with lymphocyte, and (vi) artifact (blur area). Here, (ii) and (iii) have some common visual features with class ℂ; on the other hand, (iv), (v) and (vi) are non-specific structures which might also appear in the other two classes, ℂ and 𝕆.

Table 4.6: **Experiment settings E2**: E2 settings are designed to test H2 (i.e. ¬ℂ-biased training produces less false-positives) with single-label patches. There is a total of 4𝕌 patches in the training set of each experiment.

Experiment ID	Distribution	Patch ratio (ℂ : ¬ℂ)
E2.a	Balanced	2 : 2
E2.b	Over-represented ¬ℂ	1 : 3
E2.c	Over-represented ℂ	3 : 1

¬ℂ, while E2.b (resp. E2.c) over-represents ¬ℂ (resp. ℂ). In the literature, the ratio of under-representation to over-representation is around 1:3 [Bejnordi et al., 2017; Liu et al., 2019], so we follow the same ratio in our experimental design. The total number of patches in the experiments used to test hypothesis H2 is lower than that of H1, mainly because of the expected ratio (1:3) and the total number of extracted patches for the less frequently occurring ROI classes ($\mathbb{C}$ and $\neg\mathbb{C}$). Thus, the results of experiments E1 and E2 are not fully comparable.

The hypothesis H2 can be tested on both multi-class and binary-class data sets. For the multi-class data set, we do not consider patches from category $\mathbb{O}$ –there are many $\mathbb{O}$ pixels in ROI patches– to be able to focus on the ROI cases, similarly to the common trends in the literature of filtering out non-ROIs.

H3: Multi-label examples are more useful than single-label examples as training data.

This hypothesis states that the patches containing both ROI classes $\mathbb{C}$ and $\neg\mathbb{C}$ (i.e. category $\mathbb{C}\&\neg\mathbb{C}$) add more valuable information during training than the patches containing a single ROI class (i.e. categories $\mathbb{C}$ or $\neg\mathbb{C}$). Indeed, by having both classes of pixels at the same time, multi-label patches include boundary information for the two classes, and hence there is more contextual information that could be helpful during training, especially for segmentation models, which can localize multiple classes within the same patch. In other words, we consider having multiple ROI classes in the same patch as advantageous.

Table 4.7: **Experiment settings E3**: E3 settings are designed to re-test H2 when multi-label patches are used and to test H3 (i.e. multi-label patches are more useful than single-label patches).

Experiment ID	Distribution	Patch ratio ($\mathbb{C} : \neg\mathbb{C} : \mathbb{C}\&\neg\mathbb{C}$)
E3.a	Balanced	1.5 : 1.5 : 1
E3.b	Over-represented $\neg\mathbb{C}$	0 : 3 : 1
E3.c	Over-represented $\mathbb{C}$	3 : 0 : 1

To test this hypothesis, we designed a trio of experiments analogous to E2 (which are designed with single-label patches). To construct class-biased distributions, we replaced the under-represented class examples (category $\mathbb{C}$ or $\neg\mathbb{C}$) in E2.b and E2.c with examples of multi-label patches (category $\mathbb{C}\&\neg\mathbb{C}$). At the same time, to re-evaluate H2 in the current case, we designed the corresponding E2.a experiment as well. The detailed design of the experiments is given in Table 4.7. First, in E3.a, we considered a balanced case between $\mathbb{C}$ and $\neg\mathbb{C}$. Then, similarly to E2, in E3.b and E3.c we considered over-represented

$\neg\mathbb{C}$ and over-represented $\mathbb{C}$ cases. The total number of patches in E3.* is the same as in E2.*, namely 4$\mathbb{U}$. Hence, the result of an E2.* experiment is comparable with the result of the corresponding E3.* experiment, where * can be replaced by a, b or c.

In short, we re-tested H2 with the current E3 experiment setting and tested H3 by comparing the corresponding pairs (E3.*, E2.*).

H4: Non-ROI data are useful for training.

Training CNNs requires very large data sets; sometimes data augmentation is applied to make a data set artificially larger [Walach and Wolf, 2016]. Although data augmentation is useful in increasing the generalization power of CNN models, it can cause overfitting in the case of small data sets [Shorten and Khoshgoftaar, 2019]. Obtaining a non-artificially augmented large data set of WSIs is generally expensive, since annotating ROI data requires domain experts. However, in our particular case, annotating non-ROIs is less expensive. Furthermore, inside the ROIs there are some common non-ROI regions. Non-ROI examples are required to learn how to classify these common non-ROIs inside the ROIs. By considering all these reasons, we hypothesize that non-ROI data could be useful for training a CNN, especially when the annotated ROI data set is small.

To test this hypothesis, we designed a trio of experiments analogous to E3. Specifically, we replaced some examples of class $\mathbb{C}$ and $\neg\mathbb{C}$ from E3.* with class $\mathbb{O}$ examples (see Table 4.8) to simulate the shortage of ROI data. We kept the balanced, $\neg\mathbb{C}$-biased and $\mathbb{C}$-biased distribution in the corresponding experiment for ROI classes. This represents another setting for testing H2 in which, along with class $\neg\mathbb{C}$, we consider another negative class (non-ROI class, $\mathbb{O}$). We are thus revisiting H2 as *over-representing negative classes (both* $\neg\mathbb{C}$ *and* $\mathbb{O}$*) compared to the positive class (*$\mathbb{C}$*) reduces the false-positives.*

Table 4.8: **Experiment settings E4**: E4 settings are designed to re-test H2 where non-ROI patches are used, and to test H4 (i.e. non-ROI data are useful).

Experiment ID	Distribution	Patch ratio ($\mathbb{O} : \mathbb{C} : \neg\mathbb{C} : \mathbb{C}\&\neg\mathbb{C}$)
E4.a	Balanced	1 : 1 : 1 : 1
E4.b	Over-represented $\neg\mathbb{C}$	1 : 0 : 2 : 1
E4.c	Over-represented $\mathbb{C}$	1 : 2 : 0 : 1

We designed three experiments denoted as E4.*, as shown in Table 4.8, analogous to the E3.* experiments. Similarly to the E3 experiment settings, E4 serves two purposes. The first purpose is to test H4 by comparing E4 with E3; the second is to re-test H2 with the current E4 settings. Here, we designed E4.a by replacing the same number of patches for categories $\mathbb{C}$ and $\neg\mathbb{C}$ with patches for category $\mathbb{O}$ from E3.a, thereby adding some extra non-ROI information while keeping the balance between ROI classes. We designed the following two experiments, E4.b and E4.c, as the corresponding pair for E3.b and E3.c by replacing $1\mathbb{U}$ patches from the over-represented ROI class with patches of category $\mathbb{O}$. As we kept the total number of training examples $4\mathbb{U}$, the E4.* experiments are comparable with the previous E2.* and E3.* experiments.

In summary, the E1 experiment settings test H1 on the impact of a natural, balanced distribution, while the E2, E3, and E4 experiment settings test H2 on the impact of balanced, class-biased distributions in three different cases. Moreover, the comparison between E2 and E3 tests the H3 on the impact of multi-label patches. On the other hand, the comparison between E3 and E4 tests H4 on the usability of non-ROI patches.

Note that H2 is the only hypothesis that can be tested on both multi-class and binary-class data sets: the other three are based on multi-class data sets only.

4.5.1.2 Pipeline for testing the hypotheses

After formulating the hypotheses, the next step is to test the hypotheses (Figure 4.4). The pipeline for testing a hypothesis consist of 4 steps, including pre-processing, training, inference, and evaluation. All these steps are the ones in our general processing framework described in Section 3.2.1. Here, we reformulate the steps, specifically the tanning, inference, and evaluation with mathematical instances for both the segmentation and classification formats.

Training. At the training step, we generate different class distributions as described in Section 4.5.1.1. Let $T_d = \{(I_k, g_k) : 1 \leq k \leq n\}$ be a training set of size n for a particular distribution, where $I_k \in \mathbb{R}^{l \times l \times 3}$ is a patch of dimension $l \times l$ with its corresponding ground-truth g_k. Here, $g_k \in \mathbb{R}^{l \times l \times C}$ for the segmentation task and $g_k \in \mathbb{R}^C$ for the classification task, and C is the number of classes in the ground-truth annotation (not to be confused with the cancer class $\mathbb{C}$). The training is a process of finding a classifier function $M_d : \mathbb{R}^{l \times l \times 3} \rightarrow \mathbb{R}^{l \times l \times C}$ (for segmentation) or $M_d : \mathbb{R}^{l \times l \times 3} \rightarrow \mathbb{R}^{C}$ (for classification) by minimizing a loss function $\mathcal{L}(g_k, M_d(\Theta, I_k))$, where Θ is the set of parameters of the classifier.

Inference. During inference, the trained model is employed to predict either pixels (in the segmentation task) or patches (in the classification task) for unseen test WSIs. Let W_i be the i^{th} WSI from the test set, and I_{ij} the j^{th} patch in W_i. The trained model (M_d) predicts the probability ($\hat{g}_{ij} \in \mathbb{R}^{l \times l \times C}$ or $\mathbb{R}^{C}$) of each class for each pixel in I_{ij} for segmentation, or for the whole patch for classification. In the case of segmentation, the predicted probability of the central region of a patch is taken into account during evaluation, i.e., the predicted probability $s_{ij} \in \mathbb{R}^{r \times r \times C}$ is considered, where $r \times r$ is the dimension of the central region and s_{ij} is the centre crop of $\hat{g_{ij}}$.

Evaluation. During the evaluation, we consider class $\mathbb{C}$ as the positive class and the other class(es) as the negative class for both the binary and multi-class data sets. Moreover, only the evaluation result for class $\mathbb{C}$ is presented, hence focusing on cancer detection. We also consider the patch-based evaluation for both the segmentation and classification tasks (see Section 3.3). We convert the pixel-based predicted probabilities of the segmentation task to a patch-based probability since we require the predicted probability of class $\mathbb{C}$ ($s_{ij}^{\mathbb{C}} \in \mathbb{R}$) for each patch rather than for each pixel, which we obtain by taking the class-wise max over $s_{ij} \in \mathbb{R}^{r \times r \times C}$. The predicted probabilities ($s_{ij}^{\mathbb{C}}$) are computed for all the patches from the test set.

Since one random fold is not enough to validate a hypothesis, we perform 10 trials (runs) for each experiment. The final result for an experiment is given by the mean computed over the 10 runs. We also compute the standard deviation to reflect how accurately the mean represents the 10 runs [Lee et al., 2015]. For our 11 experiments presented in Tables 4.5, 4.6, 4.7 and 4.8, we therefore have a total of $11 \times 10 = 110$ runs to test all the hypotheses for a particular hyper-parameter setting in relation to a particular data set.

4.5.2 Experimental setup

4.5.2.1 Data sets and pre-processing

The descriptions of data sets are given in Section 2.4. In this section, we mentioned the task-specific settings and pre-processing for the considered data sets.

Data set for segmentation task. For the segmentation task, we utilized the MLNTO data set (see Section 2.4.1). In the pre-processing step of our methodological framework, a total of 127,898 overlapping patches of size 384×384 pixels were extracted from the training set (for detail, see Table 4.2). According to the number of extracted patches, the value of $\mathbb{U}$ in the different experiment settings presented in Table 4.5 to 4.8 for MLNTO is 5,000. By applying the same extraction process of training patches to the test set, we obtain 101,262 patches, of which 17,351 belong to class $\mathbb{C}$.

Data set for classification task. We used the PCam data set (see Section 2.4.2) for the classification task. Since the ground-truth annotations of the PCam are class labels instead of class masks, we can only use it for the classification task. Besides, since PCam is a binary-class data set, we are only able to test H2 on this data set. To create an imbalanced distribution of a 1:3 ratio in the training set, we set $\mathbb{U} = 43,690$. Note that, we created an artificially imbalanced distribution for the training set only: the validation and test sets are still balanced. As an example of imbalanced test set, we utilized the test set of CAMELYON16 (see Section 2.4.2).

4.5.2.2 Hyper-parameter settings

For the network architectures and other hyper-parameter settings of this analysis, we refer to Section 3.2.2. Along with the settings presented in that section for the segmentation task, we ran the same experiments for one more learning rate (10^{-4}) and different random seeds; results were consistent across the various configurations.

4.5.2.3 Evaluation measures

In our hypotheses, we emphasize false-positives (FPs), and consider the precision and the false-positive rate (FPR) as representative metrics. Along with the FPs, measuring the false-negatives (FNs) is also important. We consider recall (also known as sensitivity or TPR) as a metric which is responsive to FNs.

Rather than measuring these values separately, we preferred to resort to curve-based metrics that show the trade-off between two measures, i.e., considered multi-threshold-based metrics. Specifically, we considered ROC and PR curves. Note that, in these cases the predicted probability threshold is automatically set to obtain a certain value of, let us say, recall. This means that in a ROC or PR space, a particular point is computed for different models at different thresholds of predicted probability. Thus, a fair comparison has to be prevented to enable hypothesis testing. To resolve this challenge, we computed precision and FPR for a constant list of predefined predicted probability thresholds for all the models, and presented them as precision and FPR curves. The constant list of thresholds includes 101 points from 0.0 to 1.0 by maintaining a regular stride. Moreover, we used the Area Under Curve (AUC) to compare curves overall.

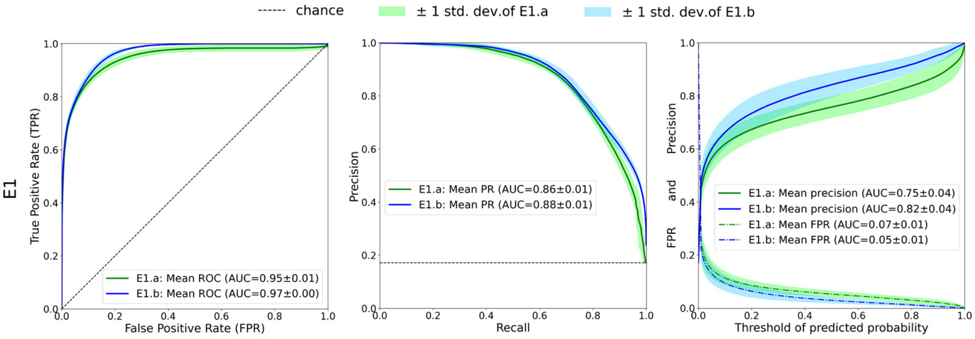
To calculate the mean curve of the 10 runs, we normalized the predicted probabilities for each run by applying min-max normalization, then computed the curve-based metrics for each run. To calculate the mean ROC and PR curves, we selected 1,001 points from both the ROC and PR spaces for each run by using linear interpolation. We then calculated the mean over 10 runs. We considered 1,001 points, as we have a huge number of data points (i.e. test patches) ranging from 32k to 10.5M, depending on the data set.

For FPR and precision curves, no interpolation is required to compute mean curves since the predicted probability threshold is constant for all runs.

Although we emphasized the multi-threshold-based metrics, we also computed the result on single threshold-based metrics. We, however, presented only the statistical test on the precision in that case since the conclusion of the results based on these metrics was consistent with the multi-threshold-based ones.

4.5.3 Results and discussion

In this section we present the results which allow us to decide whether a hypothesis is true or false. First, we experiment on the multi-class data set (i.e. MLNTO) while taking the segmentation task into account. Later, we discuss the results for the binary class data sets while taking the classification task into account. In both cases, we make a decision about a hypothesis based on the numeric results before discussing the reasons for the validity or invalidity of a hypothesis.



(i) ROC curve (ii) PR curve (iii) Precision and FPR curves

Figure 4.6: **Natural distribution (E1.b) is better than balanced distribution (E1.a) (¬H1).** Mean curves for (i) the ROC curve, (ii) the PR curve, and (iii) the precision and FPR curves. Sub-figures are obtained for 10 runs for balanced distribution (E1.a, lines in green) and for natural over-representation of $\mathbb{O}$ distribution (E1.b, lines in blue). The standard deviation is represented by green/blue colored shading. MLNTO data have been used.

4.5.3.1 Results for segmentation task

4.5.3.1.1 Natural imbalanced distribution is better (¬H1)

If we consider the experiment settings E1 as summarized in Table 4.5, we can observe that the results do not confirm hypothesis H1 and we can conclude that *the naturally imbalanced distribution for class $\mathbb{O}$ (E1.b) is better than the artificial balanced distribution (E1.a)* for WSIs in the segmentation setting.

Figure 4.6 shows the main results for H1. While considering the trade-off between two metrics, in Figure 4.6(i), which represents the ROC curve, we can observe that E1.b has a higher true-positive rate (TPR) than E1.a at almost every false-positive rate (FPR) point. Models trained under the natural distribution give higher recall while also producing fewer false-positives (FPs) than models trained under the balanced distribution.

According to the precision-recall (PR) curve presented in Figure 4.6(ii), the precision of E1.b is also higher at almost every recall point than that of E1.a. Even at the recall point 1.0, the precision of E1.b is better than that of the random chance baseline, while the precision of E1.a is as low as the random chance.

With regard to the precision and FPR curves in Figure 4.6(iii), the conclusion in favor of E1.b also holds: E1.b has higher precision and a lower FPR

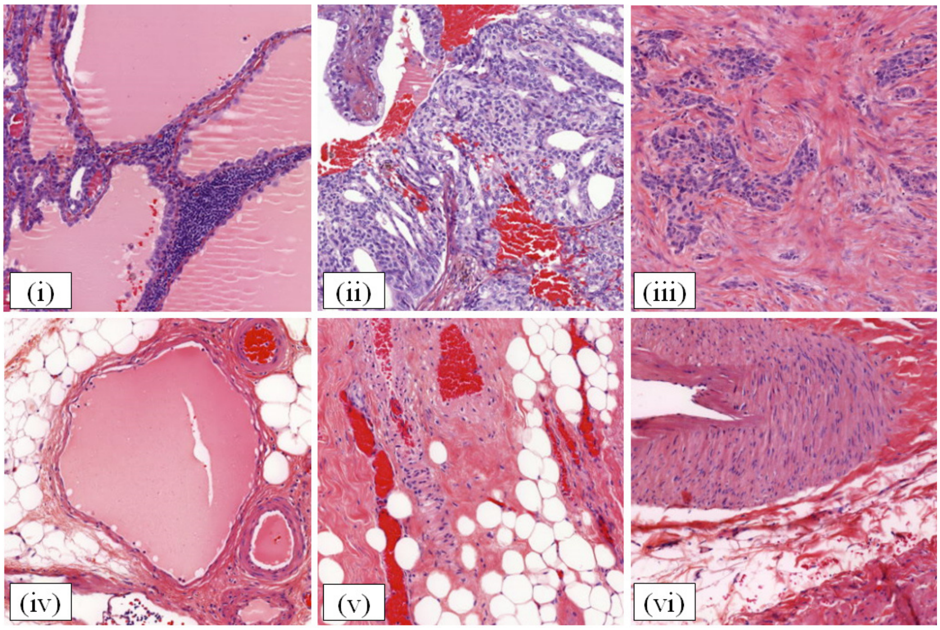


Figure 4.7: **Some examples of confusion between *other* ($\mathbb{O}$) and *cancer* ($\mathbb{C}$)** classes showing inter-class similar regions with common histological structures. On the first row, (i), (ii) and (iii) are patches of category $\mathbb{C}$ with fluid, blood and fibrosis with lymphocytes, respectively, which are taken from the training set. On the second row, (iv), (v) and (vi) are patches of category $\mathbb{O}$ from the test set with the same histological structures. During training, if there are not enough examples of class $\mathbb{O}$ containing the aforementioned structures compared to $\mathbb{C}$, the common histological structures in the test patches are predicted as $\mathbb{C}$ (false-positive).

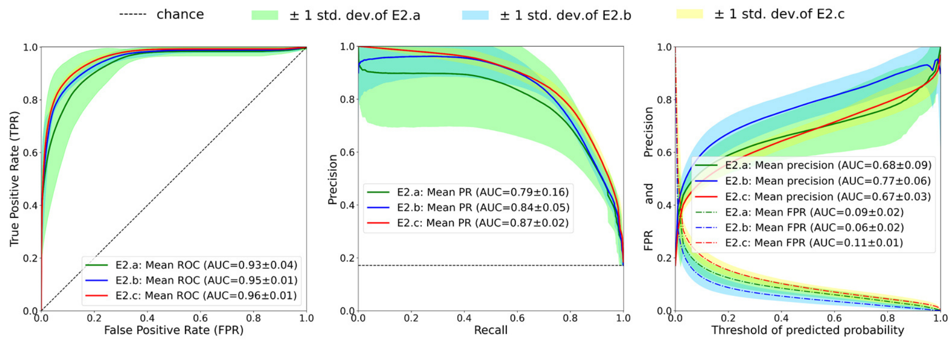
than E1.a for all predefined thresholds.

We also applied statistical tests. When precision is calculated using the default argmax decision of the model, precision is about 7% higher in E1.b than E1.a, and the difference is statistically significant according to the t-test (p-value < 0.002).

We further investigated why the distribution helps the models, if it is imbalanced towards $\mathbb{O}$, by manually looking at the predicted masks of the models. Pathologists confirmed that the miss-classification by E1.a (the balanced distribution) is certainly due to inter-class similar regions: the regions containing common histological structures, e.g. blood, fibrous tissue, etc. These histological structures are common in both ROI ($\mathbb{C}$ and $\neg\mathbb{C}$) and non-ROI ($\mathbb{O}$) regions (see Figure 4.7). When an ROI contains such a histological structure in a small area, the human annotator overlooks the small area, annotating it as a corresponding ROI (ℂ or ¬ℂ), although the annotation is actually wrong. During training, we therefore need enough examples of these inter-class similar regions (i.e. the over-representation of class 𝕆) with their actual annotation to compensate for the unavoidable pitfalls arising from the annotation, or else false-positives may be caused during prediction.

4.5.3.1.2 Over-representing ¬ℂ class reduces false-positives (H2)

For testing hypothesis H2, *"over-representing ¬ℂ class reduces false-positives"*, we first used the E2 settings, where every example belongs to a single-label category (see Table 4.6).



(i) ROC curve (ii) PR curve (iii) Precision and FPR curves

Figure 4.8: **Over-representation of negative classes ¬ℂ reduces false-positives (hypothesis H2).** Mean curves with standard deviation for experiment settings E2.a (for balanced distribution in the training set, green), E2.b (for over-represented ¬ℂ, blue) and E2.c (for over-represented ℂ, red). Experiments test H2 in the case of E2, and every example belongs to a single-label category. Same notations as in Figure 4.6.

Figure 4.8 presents the main comparative results among the experiments, E2.a (for balanced distribution), E2.b (for over-represented ¬ℂ) and E2.c (for over-represented ℂ) for H2. In Figure 4.8, we can first observe that the balanced distribution (E2.a) is not as robust as the other distributions (high standard deviation in green). We suspect that the random selection of the under-represented class ℂ in the training set could not cover all the cancer types, thereby inducing more uncertainty in the results and causing this deviation (we further discuss this in Section 4.5.3.1.3).

Nonetheless, if we consider the mean results (curves) over 10 runs, we can see that the balanced distribution (green curve) is at its lowest for the ROC curve (see Figure 4.8(i)). The results are closer for the over-representation of $\mathbb{C}$ (red curve) and $\neg\mathbb{C}$ (blue curve).

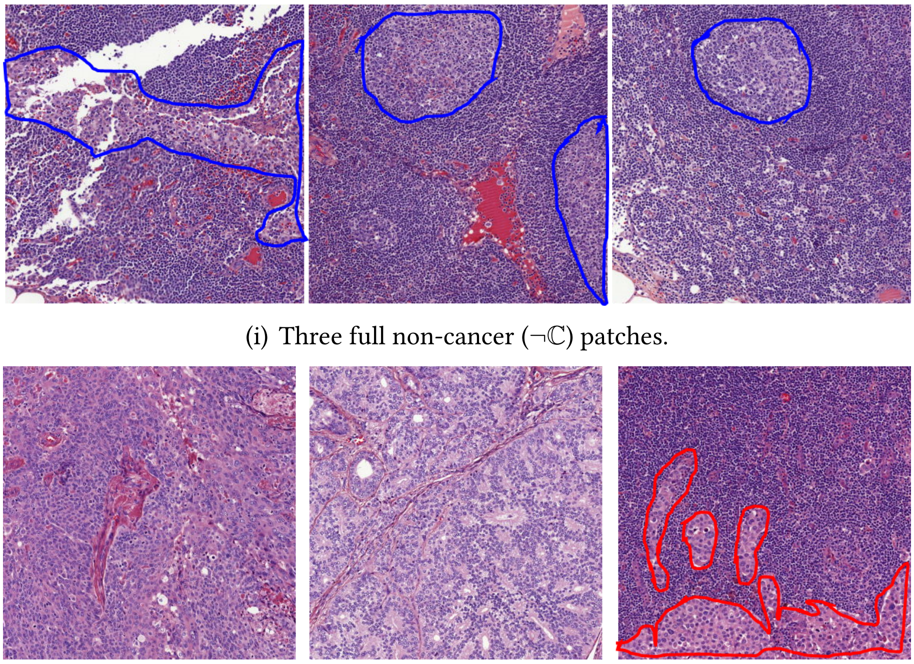
The same holds for precision-recall (PR) curves (Figure 4.8(ii)): while the balanced distribution is the lowest on average and has a high standard deviation, the two other settings are close to each other and more robust, with a very small superiority in favor of the over-representation of $\mathbb{C}$.

While the previous two curves (ROC and PR) are computed for automatic non-fixed thresholds and present the trade-off between two metrics, the precision and FPR curves are computed for predefined constant thresholds for all models and present threshold by threshold fair comparison (Figure 4.8(iii)). When looking at the curves on Figure 4.8(iii), we can see that the precision is the highest and the false-positive rate (FPR) the lowest in the case of the over-representation of $\neg\mathbb{C}$ (blue curve). We can also examine the area between the curve and the x-axis (i.e., AUC) to compare the curves. The AUC for the over-representation of $\neg\mathbb{C}$ (E2.b) in terms of precision is 0.77 compared to 0.67 and 0.68 for the over-representation of $\mathbb{C}$ (E2.c) and balanced (E2.a) distribution. Moreover, according to the t-test on the precision calculated from the default argmax decision of the models, we found that the precision of E2.b is statistically significantly higher than E2.a and E2.c with p-values < 0.0274 and 0.0013. These results show that our hypothesis is true: the over-representation of the negative class $\neg\mathbb{C}$ reduces the false-positives, and the balanced distribution may be less robust than the others we tested.

We took a closer look at the data in order to gain a better insight into the numeric results. While examining several heatmap images for the balanced distribution (E2.a) and for the over-represented $\mathbb{C}$ distribution (E2.c), and comparing them to the ground-truth image, we observed that the errors are caused by the inter-class similar regions between classes $\mathbb{C}$ and $\neg\mathbb{C}$ along with the case of common histological structures we described earlier.

The inter-class similar regions are the regions which share common visual characteristics between class $\mathbb{C}$ and class $\neg\mathbb{C}$. Specifically, there are some $\neg\mathbb{C}$ regions containing a germinal center, macrophages and a blur area (a kind of artifact) (see Figures 4.5 and 4.9) that have some visual characteristics in common with class $\mathbb{C}$ regions.

In Figure 4.9, we illustrate an example of inter-class similarity and intra-class difference. In Figure 4.9 (i), the regions inside the blue contours are visually different from the usual $\neg\mathbb{C}$ region (i.e. the region outside of the



(i) Three full non-cancer ($\neg\mathbb{C}$) patches.

(ii) First two patches are cancer ($\mathbb{C}$) patches with 100% class $\mathbb{C}$ pixels, while the last one is a mixed ($\mathbb{C}$&$\neg\mathbb{C}$) patch (cancer inside the red contour and non-cancer outside).

Figure 4.9: **Examples of intra-class difference in $\neg\mathbb{C}$ and inter-class similar regions between cancer $\mathbb{C}$ and non-cancer $\neg\mathbb{C}$ classes**. (i) The regions inside the blue contours are examples of the intra-class different regions for the non-cancer $\neg\mathbb{C}$ class, which are visually different from the usual non-cancer $\neg\mathbb{C}$ region (i.e. the region outside the blue contour); these regions are visually more similar to the cancer $\mathbb{C}$ regions in (ii) than the non-cancer $\neg\mathbb{C}$ regions, and are thus examples of inter-class similar regions between cancer $\mathbb{C}$ and non-cancer $\neg\mathbb{C}$ classes.

blue contour); hence, they are the intra-class difference regions of $\neg\mathbb{C}$. These $\neg\mathbb{C}$ regions (inside blue contours) are visually more similar to the $\mathbb{C}$ regions in Figure 4.9 (ii) than the usual $\neg\mathbb{C}$ regions (outside the blue contours) and hence they are examples of inter-class similar regions between $\mathbb{C}$ and $\neg\mathbb{C}$ classes. For instance, both are lighter in color than the other parts of the lymph node, the nuclei in both regions are farther from each other than the usual parts of the lymph node, and the nuclei in both regions are larger than the usual size of a nucleus. Hence, during training, if there are not enough examples of $\neg\mathbb{C}$ containing these *specific* inter-class similar regions compared to class $\mathbb{C}$ examples, class $\mathbb{C}$ will dominate. As a result, it is more likely that

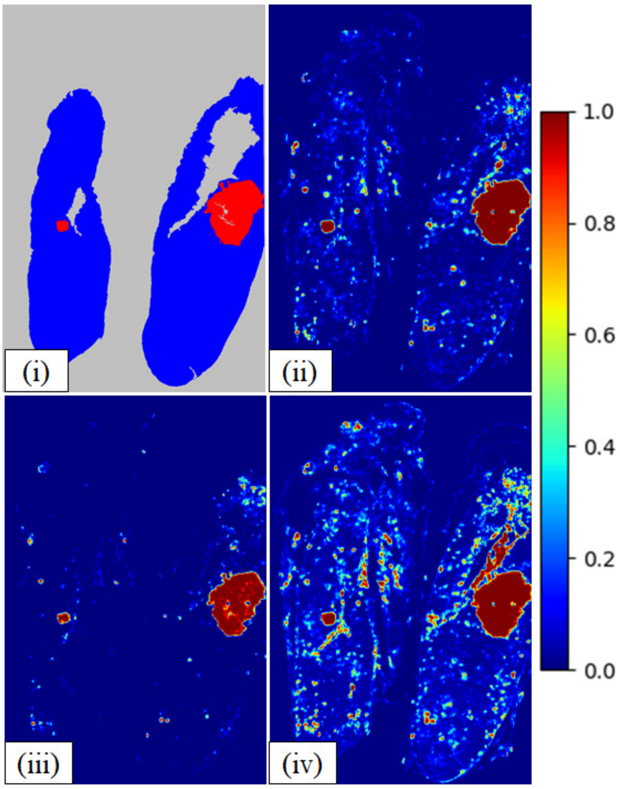
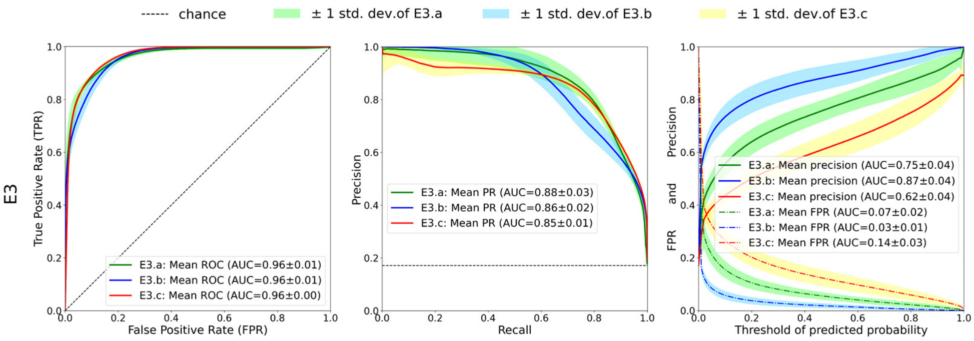


Figure 4.10: **Illustrative example E3.b is the best**. Heatmaps of class ℂ for the WSI given in Figure 2.3 (one of the WSIs with the most false-positives) from the test set of MLNTO. Here, (i) is the ground-truth, (ii) is the heatmap generated by the E3.a, (iii) by E3.b, and (iv) by E3.c.

these regions will be predicted as ℂ in any of the test WSIs, although many of them will be false-positives. Indeed, in practice, the number of these *specific* regions in the lymph node are few compared to the whole lymph node. Thus, it is better to choose a class-biased distribution toward ¬ℂ so that enough examples containing the aforementioned *specific* regions are used during training. The result of experiments E2 in Figure 4.8 indicates this conclusion empirically. We further test the H2 in two other settings, E3 and E4, in the following subsections, along with two other hypotheses H3 and H4.

Figure 4.10 shows some examples of heatmaps of class ℂ generated by the experiments E3.a, E3.b and E3.c for a WSI. From Figure 4.10, we can see that the prediction by E3.b is better than the two other predictions regarding the number of false-positives in the prediction.

4.5.3.1.3 Multi-label examples give extra advantages over single-label examples (H3)



(i) ROC curve (ii) PR curve (iii) Precision and FPR curves

Figure 4.11: **Multi-label examples give extra advantages over single-label examples (H3 hypothesis).** Mean curves with standard deviation using E3 settings (Table 4.7) where there are multi-label examples: E3.a (for balanced distribution in the training set, green), E3.b (for over-represented $\neg\mathbb{C}$, blue) and E3.c (for over-represented $\mathbb{C}$, red) and using the same notations as in Figure 4.8.

Unlike in the E2 settings, where every example belongs to a single-label category, in the E3 settings presented in Table 4.7, some of the examples belonging to a single-label category are replaced by multi-label examples (i.e. some pixels are $\mathbb{C}$ and some are $\neg\mathbb{C}$ in the same patch example). E3 setting serves two purposes; one is to test H3 (multi-label examples are more useful than single-label examples) by comparing with the experiment setting E2, and another is to revisit hypothesis H2 in a different setting. Figure 4.11 presents the mean curves and the standard deviation of 10 runs for E3.

By comparing single-label (Figure 4.8) with multi-label (Figure 4.11), we can see that the results in Figure 4.11 have improved when considering both the curves and AUC values. The ROC curves in Figure 4.11(i) are more stable (i.e. have less deviation) and have higher AUC than those in Figure 4.8(i) except for E3.c ($\mathbb{C}$-biased). Same improvements are observed in the precision-recall (PR) curves, precision curves, and FPR curves.

According to the t-test on the precision calculated from argmax decision, we found that the precision of E3.a and E3.b are significantly higher (p-value $<$ 0.002 and 0.0002) than those of E2.a and E2.b, respectively, while the precision of E3.c is significantly lower (p-value $<$ 0.04) than that of E2.c. The findings are consistent in all evaluation metrics. Hence, the H3 is true for E3.a and E3.b while not for E3.c.

We investigate the reason for having lower result for E3.c than for E2.c. According to our observation, the reason is the ratio of positive and negative classes in the training set of E3.c. Including the patch category $\mathbb{C}\&\neg\mathbb{C}$ in E3.c changes the ratio of classes (negative:positive) from 1:3 (see Table 4.6) to 1:4 (see Table 4.7), i.e., it decreases the presence of negative class since $\mathbb{C}\&\neg\mathbb{C}$ category contains class $\mathbb{C}$ (positive class) pixels along with class $\neg\mathbb{C}$ (negative class) pixels. Consequently, increasing the over-representation of the positive class decreases the result of E3.c. Thus, according to our general hypothesis H2, having a lower result in E3.c than E2.c is expected. At the same time, the better result of E3.a and E3.b than the corresponding E2.* empirically proves that H3 is true.

The multi-label patch category $\mathbb{C}\&\neg\mathbb{C}$ gives two advantages. 1) It gives an opportunity to the FCNN models to learn about two classes from the same patches thus gives a chance to learn boundary line between those classes. 2) In the case of micro-metastasis, it is not always possible to extract single-label patches for category $\mathbb{C}$ thus there is a chance of missing examples from a cancer type or WSI. It is, however, always possible to extract patches of category $\mathbb{C}\&\neg\mathbb{C}$ from both micro and macro metastases; thus it reduces the chance of missing any cancer type and example from a WSI which reduces the uncertainty and produces a less deviated result.

We re-tested the H2 in the current setting E3 by comparing the results of E3.a, E3.b, and E3.c as described in the following paragraphs.

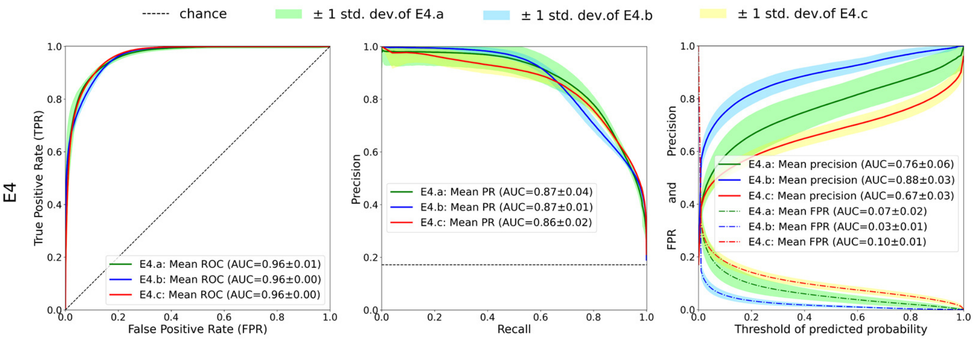
While considering the trade-off between FPR and TPR in ROC curves (see Figure 4.11 (i)), the results are comparable for all experiments (all ROC AUCs are 96), i.e., the ROC curve is insensitive to the different distributions in training set for this case.

According to the PR curves, and depending on the trade-off under consideration, the settings have a different impact. For high precision, the over-represented $\neg\mathbb{C}$ (in blue) is both robust (low standard deviation) and has a higher impact than the other settings. From these results, it is also clear that along with the distribution of the training set, the cutoff or threshold selection is also important if we are to obtain a desirable performance from the system.

The differences are much greater when we consider precision and false-positive rates (FPR) curves (Figure 4.11 (iii)). If we take a closer look at the blue curves on the right side of Figure 4.11, the over-representation of ¬ℂ (E3.b) has the highest precision (AUC 0.87 ±0.04) and the lowest FPR (AUC 0.03±0.01), while the balanced distribution reaches an AUC of 0.75 ± 0.04 for precision and 0.07 ±0.02 for FPR. These results are consistent with the results from Figure 4.8(iii) in terms of the interest of over-representing ¬ℂ, i.e., hypothesis H2.

Additionally, we performed statistical tests for precision when it was calculated using the default argmax function. We found that precision is higher (statistically significant) for E3.b (for the over-represented ¬ℂ distribution) than for both E3.a (for balanced distribution) and E3.c (for over-represented ℂ distribution). In other words, H2 is true in setting E3 as well.

4.5.3.1.4 Non-ROI data are useful (H4)



(i) ROC curve (ii) PR curve (iii) Precision and FPR curves

Figure 4.12: **Non-ROI data are useful (H4 hypothesis).** Mean curves with standard deviation using E4 settings (Table 4.8) where 𝕆 examples replace some ℂ examples and some ¬ℂ examples: E4.a (for balanced distribution in the training set, green), E4.b (for over-represented ¬ℂ, blue) and E4.c (for over-represented ℂ, red) and using the same notation as in previous figures.

If we compare the result of E4.* (with non-ROI patches (Figure 4.12) with the result of E3.* (without non-ROI patches) (Figure 4.11), we can see that the results for E4.* are slightly better than, or comparable to the corresponding result for E3.*.

ROC curves in Figure 4.12(i) are very similar to each other and also very similar to the curves obtained in E3 (AUC=0.96) (Figure 4.11(i)). Precision-recall (PR) curves in Figure 4.12(ii) have similar shapes and are ordered the same way as in E3 (Figure 4.11(ii)). AUC values are also quite similar to each other when compared to E3.

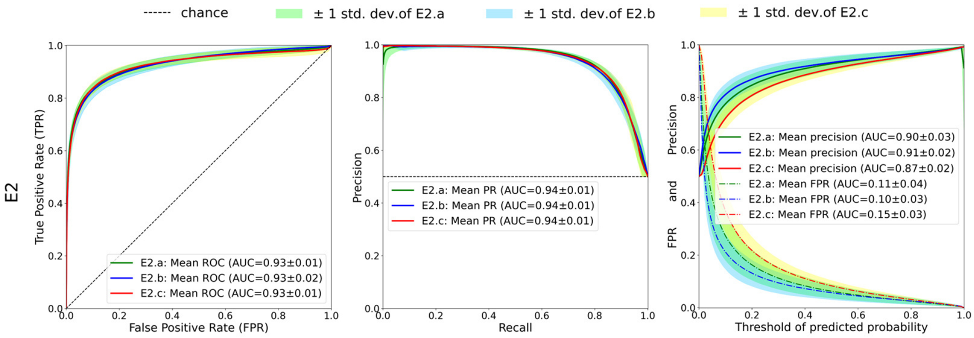
According to Figure 4.12(iii), the results for E4 have slightly improved compared to E3 (AUC increases slightly for precision and decreases slightly for the false-positive rate (FPR)) in Figure 4.11(iii). The standard deviation for the 10 runs also slightly decreases, except for the balanced distribution, which still has the highest standard deviation, regardless of the measure being considered. By using the t-test on precision with the default argmax decision on predicted probabilities, we found that the higher precision for E4.b compared to E3.b is not statistically different (p-value 0.22), while it is for the E*.c case (p-value 0.0034).

Nonetheless, the better results for E4.* compared to E3.* indicate that a considerable number of errors come from inter-class similar regions or common histological structure between ROIs and non-ROIs. However, miss-classification due to common histological structure (e.g. blood) between ROIs and non-ROIs is easy to identify, even with the naked eye, while miss-classification due to inter-class similarity between the ROI classes, $\mathbb{C}$ and $\neg\mathbb{C}$, is more difficult to identify. This explains why the experiment setting for E3.b is more appropriate than the setting for E4.b. Nevertheless, as H4 is a true hypothesis, inexpensive to annotate non-ROI examples will be useful for CNN training. Therefore, if there is a shortage of expensive to annotate ROI examples while training a CNN, this can be compensated by adding relatively easy-to-annotate non-ROI examples. This result is also consistent with our results from hypothesis H1.

We also re-test H2 in the E4 setting, and the findings are consistent with the E3 case. In other words, H2 is true in the E4 setting as well.

4.5.3.2 Results for the classification task

In this section, we further discuss the results of hypothesis H2 ($\neg\mathbb{C}$-bias produces fewer false-positives) for the binary class data sets PCam and CAMELYON16 for the classification task. Here, all the models were trained on the PCam data set. The trained models were then tested on both PCam (balanced) and CAMELYON16 (highly imbalanced) test sets.



(i) ROC curve

(ii) PR curve

(iii) Precision and FPR curves

Figure 4.13: **Over-representing the negative class ¬ℂ reduces false-positives - on PCam.** Mean curves with standard deviation (std. dev.) of experiments for balanced distribution (E2.a), for over-represented ¬ℂ (E2.b), and for over-represented ℂ) (E2.c) on PCam.

.

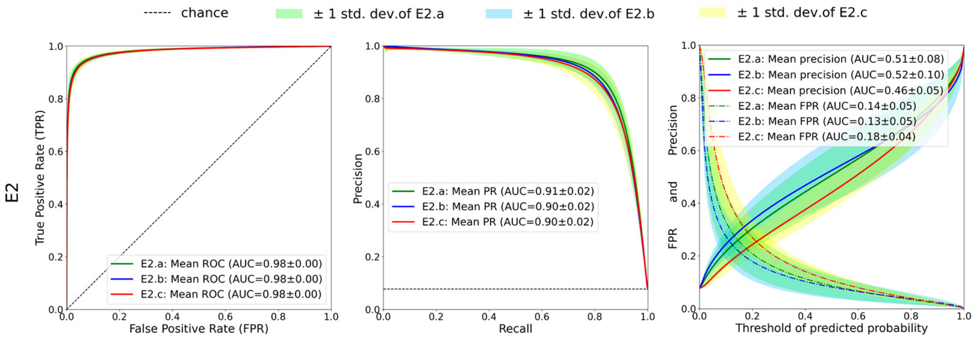
4.5.3.2.1 Balanced test set: over-representing the negative class reduces false-positives (H2)

According to the ROC and PR curves in Figure 4.13 (i) and (ii), all the distributions produce comparable results (also observed when zooming in). However, we can see that E2.b has a higher level of performance than the other two experiments at low thresholds.

According to the precision and FPR curves in Figure 4.13 (iii), E2.b has the highest level of performance, although for the threshold of predicted probability greater than 0.8, the results are comparable. According to the evaluation based on the argmax decision on the model, the precision of E2.b is higher than that of E2.c (p-value < 0.019) and not statistically different from E2.a.

In other words, for the classification task with balanced test and validation sets, the ¬ℂ-biased distribution produced fewer FPs than the other two distributions, which is consistent with previous results, and confirms the H2 hypothesis. However, this is not always significantly true for this classification task with balanced test and validation sets. We must assume that the balanced distribution of the validation set, which is used for parameter tuning during training, might be the reason.

4.5.3.2.2 Imbalanced test set: over-representing the negative class reduces false-positives (H2)



(i) ROC curve (ii) PR curve (iii) Precision and FPR curves

Figure 4.14: **Over-representing the negative class ¬ℂ reduces false-positives - on CAMELYON16.** Mean curves with standard deviation (std. dev.) of experiments E2.a (for balanced distribution), E2.b (for over-represented ¬ℂ) and E2.c (for over-represented ℂ) using the same keys as in the other figures. In all the experiments, the models were trained on PCam and tested on CAMELYON16.

Figure 4.14 illustrates the test results for H2 on CAMELYON16, whereas the models were trained on PCam (a subset of CAMELYON16). According to this figure, the results on CAMELYON16 are consistent with the results on PCam, i.e. H2 is true, although not always significantly true for both balanced and imbalanced test sets, while the validation set has a balanced distribution.

Figure 4.14 (i) shows that the ROC curves are insensitive to the different distributions in the training set. It may be because of the highly optimistic nature of ROC curves with regard to a highly imbalanced test set [Davis and Goadrich, 2006].

On the other hand, in Figure 4.14(ii), it is clear that the PR curves are also insensitive to the different distributions in the training set for a highly imbalanced test set, except for a certain range of recall where we can observe some slight differences in favor of the balanced distribution for the recall range between 0.75 and 0.85, along with a high standard deviation.

On the contrary, in Figure 4.14(iii), the precision of E2.b is higher than the two others for predicted probability threshold less than 0.7. Moreover, E2.b has the lowest FPR. These results are consistent with the results we obtained for the PCam test set. According to the t-test on the precision calculated from the default argmax decision of the models, the finding is consistent with the PCam case as well. Specifically, we found that the precision of E2.b is not significantly higher (p-value ≈ 0.37) than that of E2.a, i.e., both are comparable, while it is significantly higher (p-value < 0.04) than that of E2.c. In short, for the classification task with the mentioned setting, H2 is true, however, not always with a significant difference.

Note that the CAMELYON16 test set is highly imbalanced towards class $\neg\mathbb{C}$ (see Figure 2.10(ii)). Thus, there is a strong likelihood of producing FPs, which is what the models do. Hence, the AUC of the precision curve in Figure 4.14(iii) is lower than the other two data set cases presented in Figures 4.8(iii), 4.11(iii), 4.12(iii), and 4.13(iii). However, the high performance levels of the models in terms of the ROC, PR and FPR curves shown in Figure 4.14 indicate that they fail to reflect our observation, while the precision curve succeeds. Consequently, the precision curve is more robust than the ROC, PR and FPR curves in comparing the performances of the different models.

We also observed the predictions of different experiments for the CAMELYON16 test set, and we found that, like the MLNTO data set, the false-positives are the major problem caused by inter-class similarity and intra-class difference. Figure 4.15 shows some examples of heatmaps predicted by different experiments. According to this figure, the E2.b (for over-represented $\neg\mathbb{C}$) produces fewer false-positives than the other two experiments, which confirms that H2 is true.

4.6 Conclusion

In this chapter, we performed a data-level analysis to determine the optimal distribution of the classes in the training set for WSIs when using deep learning. According to our preliminary study, we found that the default choice - the balanced distribution- is not optimal for cancer detection from WSIs. Rather class-biased distributions perform better, and FP is the major cause of error in this task. Based on these findings, we derived several hypotheses and performed a deeper analysis. In this analysis, we considered the case of FCNN for segmentation and CNN for patch classification.

To the best of our knowledge, our analysis is pioneering in the case of class distribution analysis of WSI data for deep learning models; previous research has mainly focused on end-to-end pipeline development for cancer

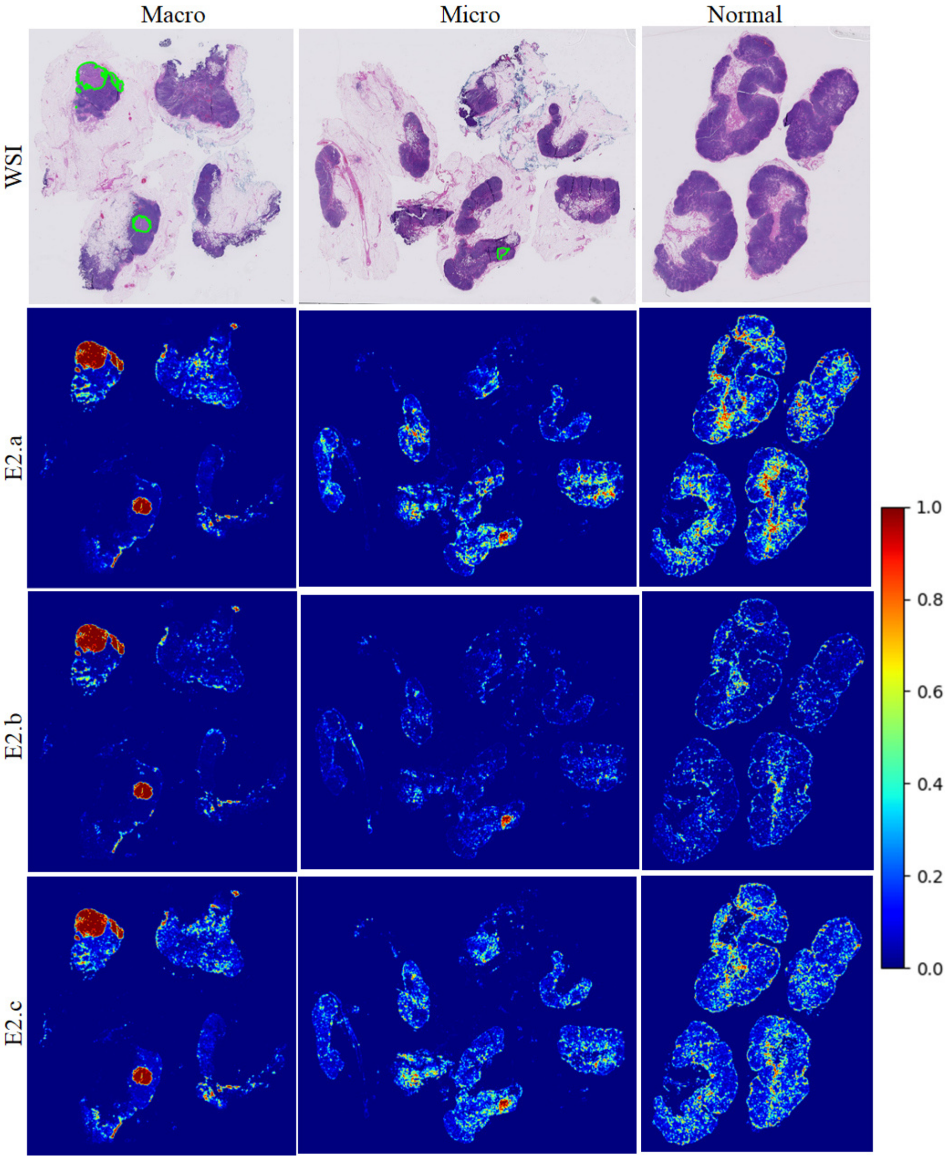


Figure 4.15: **Illustrative examples for ¬ℂ-biased training (E2.b) produces the smallest number of FPs (H2).** Example heatmaps of class ℂ generated by the experiments E2.a (for balanced distribution), E2.b (for over-represented ¬ℂ) and E2.c (for over-represented ℂ) for WSIs with macro-metastasis, micro-metastasis and normal tissue from the CAMELYON16 test set. Here, in the WSIs, the regions inside the green contours are ground-truth annotations for class ℂ.

detection.

We first compared the natural distribution (biased towards non-ROI pixels/patches) with the more commonly used balanced distribution (H1, experimental setting E1). We found that the natural distribution of the WSI data is superior to the artificially balanced distribution. In natural distribution, the data is highly biased towards the non-ROIs, while the distribution of the two ROI classes is variable, depending on the WSIs which have been included in the data set. Since non-ROI examples are usually much easier to annotate, and annotation of ROI is costly in domains where only experts can provide the labels, this result is of huge importance.

We then focused on the distribution of ROI classes. The ROI class distribution can be balanced, as in the MLNTO training set, or highly imbalanced, as in the CAMELYON16 test set. Since the natural distribution of the ROI classes in a WSI data set is variable, choosing an optimal distribution for the ROI class while building a training set is an issue. In H2, and with experimental setting E2, we show that the generally recommended balanced distribution is not the best. Instead, the non-cancer-biased training set produces the best performance, bearing in mind precision and the number of false-positives.

In the literature, multi-label patches are considered problematic [Halicek et al., 2019]. According to the test result of hypothesis H3, we found that multi-label patches give extra advantages over single-label patches. Thus, when building a training data set, they can be of huge importance. In other words, it is better to have multi-label patches than additional positive examples.

We carried out an in-depth analysis of the results from the first hypothesis that non-ROIs can still be of use. They are indeed useful as a replacement of ROI data in a case where the ROI data are limited or small. Moreover, they are easier to annotate than cancer/non-cancer (i.e., ROI data). This finding is very important because non-ROIs could be the choice for obtaining a large enough data set at a low cost for training a deep model.

In addition to observing the results of deep learning models, we also had a close look at the data to be able to form medically-oriented conclusions. While manually observing the predicted mask, we learned the importance of class heterogeneity and inter-class similarity. When building a new data set of histological images, more examples should be added from a heterogeneous class, which can compensate for the confusion caused by inter-class similarities.

When it comes to the two different tasks, segmentation and classification, we found that classification task is less sensitive to the different distributions in the training set than the segmentation task.

While we mainly focused on the training set distribution and, to a lesser extent, on the test set distribution for which we obtained consistent results, we did not study the distribution of the validation set. This set can also have an impact, since it is used for tuning the models during training. For example, for the classification task and when analyzing the impact of the test set distribution, we kept the original (balanced) distribution of the PCam validation set. The balanced distribution of the validation set may have an impact on the relative insensitivity of the test distribution with regards to the classification task. We aim to address this challenge in future work.

In summary, our study is representative, although not exhaustive. The conclusions could be further tested in other domains and some other machine learning algorithms to test the generalizability of the proposed hypotheses. Although here we considered the class ratio as 1:3 to create class-biased data distribution as recommended in [Bejnordi et al., 2017; Liu et al., 2019], other class ratios could also be explored to find the optimal class ratio. Nevertheless, we believe that the outcomes of the analysis will be helpful for researchers who are building a training data set of WSIs. Similar kind of analyses on any domain could serve for deciding which examples should be first added in the training set when they are costly to add, and which class distribution should be followed while utilizing existing training sets in CNN training. Especially, such analyses could help in the real-world problems where data have a complex history, as discussed by Crawford [Crawford, 2016] regarding the importance of building a training set with the proper distribution.

This contribution on the preliminary study has been published in [Reshma et al., 2019]. A part of the contribution on deeper analysis has been published in [Reshma et al., 2021], and the full contribution has been submitted to the Journal of Digital Imaging (JDI).

Chapter 5

PR-T Evaluation Metric for Binary Classification

Contents

Abstract.

Binary classification efficiency is usually evaluated with the Receiver Operating Characteristics (ROC) curve, which considers the trade-off between the incorrectly predicted negative example rate and correctly predicted positive example rate. The ROC curve is too optimistic in the case of imbalanced class distribution of the examples, which is common in real-life use cases. The Precision-Recall (PR) curve does not have this drawback. Computing the mean PR curve, however, is not straightforward since it requires non-linear interpolation or approximation and has to address start, end, intermediate points with tied values cases. There is no precise method that correctly computes the mean PR curve by handling all these special cases. In this chapter, we show that the PR curve can also be inefficient because of presenting precision as a function of equally spaced recall and using interpolation. We present the precision and recall as functions of continuous threshold thus develop precision and recall (PR-T) curves as an alternative to existing measures. It is a multi-threshold-based evaluation metric. We show that it is more informative than the existing measures and not over-optimistic to the imbalanced data sets. Moreover, it does not require interpolation and does not need to address any special cases with the tied values. It can be applied to any binary classification problem, either with balanced or imbalanced data sets. Additionally, we propose a method to compute the mean PR curve correctly to ensure a fair comparison with mean PR-T curves.

5.1 Introduction

Binary classification aims at classifying data examples into two classes. It has many applications, such as determining whether a person suffers from a disease or whether a connection to a system is a fraud. The usual practice is to consider the class of interest, for example the disease class, as the positive class and the other class as the negative one. In automatic binary classification, a model is trained from positive and negative examples based on their features and the target class. The trained model can then predict a confidence score (also known as classification score or predicted score) for each of the classes for any new example based on its features. The confidence score is the probability for the example to belong to a class. A decision boundary or *threshold* is used to determine the predicted class from the predicted score. When the predicted score for an example is greater or equal to a given threshold, it is a predicted positive.

The performance of a classification model is evaluated on a data set -the test set- with different metrics. The Receiver Operating Characteristics (ROC) curve is one of the most popular evaluation metrics for binary classification problem [Fawcett, 2006; Saito and Rehmsmeier, 2015]. It is also validated by our survey in Section 2.5 which shows that it is the third most popular metrics among 24 metrics, including both single and multi-threshold-based, and the most popular among the multi-threshold-based metrics. The ROC curve graphically represents the trade-off between the false-positive rate (FPR) and the true-positive rate (TPR) for different thresholds. It has many advantages [Cook and Ramadas, 2020] including a visual representation of TPR and FPR for all possible thresholds, easy computation of the curve, the applicability of linear interpolation [Davis and Goadrich, 2006]. Along with the curve, the area under the ROC curve (ROC-AUC) is also popularly used to have a summarized evaluation of the model without the graphical representation [Boyd et al., 2012].

However, the ROC curve is not recommended for an imbalanced test set where the negative class usually outnumbers the positive class. Because the model has many negative examples to learn from, it may predict well the negative examples but poorly the positive ones. In that case, TPR will be high for a very small change in FPR considering the ratio of correctly and incorrectly predicted examples. In other words, the easy to predict negative examples will influence getting high curve values. Consequently, the ROC curve (and ROC-AUC) will be inflated: the ROC is then overly optimistic to the model [Fu et al., 2017; Saito and Rehmsmeier, 2017; Sofaer et al., 2019; Cook and Ramadas, 2020] and will not be able to distinguish appropriately two models that differ in their performance on the minority class.

Unlike the ROC curve, the Precision-Recall (PR) curve, which represents the trade-off between TPR and the proportion of True-Positive (TP) to the total predicted positive examples, is less sensitive to the over-representation of negative data [Sofaer et al., 2019] because by definition it excludes True-Negative (TN) examples from the computation. It is thus a popular alternative metrics to the ROC curve [Brodersen et al., 2010; Keilwagen et al., 2014; Ozenne et al., 2015; Fu et al., 2017; Sofaer et al., 2019]. Nevertheless, if the number of positive examples is high enough thus get high TP, the PR curve can be amplified by the easy to predict positive examples since it is easy to obtain high precision (say 96 percent) in domains where the positive class is over-represented (say 95 percent) [Prati et al., 2011]. Hence, PR does not fully solve the problem of imbalanced class and distinction of model performances.

The first challenge in binary classification evaluation is thus to define a measure that is not falsely inflated and cannot mislead about the actual performance of the model due to data imbalance. More importantly, the measure should also be able to separate models that differ from each other.

The second challenge regarding an evaluation metric is whether it allows the model designer to analyze properly the model errors. Some applications need to minimize False-Negative (FN) (e.g., medical diagnosis), while others need to minimize False-Positive (FP) (e.g., fraud detection). For example, misclassification of a patient with a certain disease as a negative (i.e., FN) might delay the treatment, which might cost a life. In contrast, misclassification of a non-fraud as a fraud (i.e., FP) might be the reason for punishment to an innocent. Hence, presenting the reflection of false predictions is beneficial to measure the application-specific reliability of the classifier model. From ROC and PR curves, especially from inflated ones, it is difficult to know the type of false prediction the model produces the most.

PR curves also face the difficult problem of interpolation, which is however important. When evaluating a model, it is common that different trials of the model are run, e.g., with different parameter values or different train/test splits. In these cases, the mean values for the PR curve cannot be computed directly since not all trial-models have common recall (x-axis) values. Interpolation is then required to compute the mean curve of several trial-models and get the corresponding y-axis values for common predefined x-axis values or to compute the intermediate points when two adjacent points on the curve are distantly separated [Saito and Rehmsmeier, 2017]. While it is straightforward for the ROC because FPR and TPR are linearly correlated, it is not for the PR curve since precision does not monotonically change with recall. Therefore, it requires non-linear interpolation. Existing methods to non-linearly interpolate a point on the PR space [Davis and Goadrich, 2006; Boyd et al., 2013; Keilwagen et al., 2014] are not enough to compute the mean PR curve (and PR-AUC) since the start, intermediate, and endpoints with the tied recall (ties in x-axis values) need to be treated as well[1].

With the aim to target the three challenges above-presented, we present in this chapter a novel metric, named the PR-T curves, which present precision and recall in two separate curves as functions of the threshold used on the predicted scores. Both theoretically and experimentally, we show that PR-T curves eliminate the limitations of ROC and PR curves. Moreover, by considering the difficulty in mean curve computation of the PR curve, we propose a new method of mean PR curve computation by handling all the special cases.

We consider the imbalanced data sets and the related problem of appropriately distinguishing models. Later we test it on the balanced data set as well. We also consider the mean of several models to illustrate the mean computation process and corresponding effects. On the experimental part, we use a real data set of cancer detection, the CAMELYON16 as an example of a large imbalanced test set and the PCam as an example of the ready-made training set and balanced test set, as well as a toy data set of natural images (CIFAR10 [Krizhevsky et al., 2009]).

The rest of the chapter is organized as follows: Section 5.2 presents the related work; Section 5.3 details the limitations of state-of-the-art metrics, ROC and PR curves. In Section 5.4, we explain our proposed evaluation metric, PR-T curves, along with its properties. We describe the experimental setup to train our desire models in Section 5.5. Before assessing our trained models with evaluation metrics, we present a pre-assessment of those models with the predicted score statistics in Section 5.6. In Section 5.7, we present and discuss comparative studies between our proposed metric and state-of-the-art ones. In Section 5.8, we re-evaluate some representative results from our previous contribution with our proposed PR-T curves. In Section 5.9, we propose methods to compute the mean PR curve and PR-AUC. Finally, in Section 5.10, we draw some conclusions.

[1] Note that it is not the ties in predicted scores that can be solved by randomized order.

5.2 Related work

5.2.1 State-of-the-art metrics

Many papers analyzing, reviewing, and proposing different evaluation metrics are available in the literature. In this section, at first, we review the survey papers, then papers on the ROC, alternative to ROC, PR, alternative to PR, and extension on PR curves.

Prati et al. [2011] surveyed different graphical evaluation metrics including ROC, PR, and cost curves, and suggested when to use which metric. According to them, for example, the PR curve was suggested when the data is biased to the negative class and the positive class is more interesting to detect; the ROC curve is suggested if primary interest in the discriminability of the predictive model rather than the predicted score. Padilla et al. [2020] did another survey on the evaluation metrics used in object detection and localization task. According to them, the most popular metric in object detection domain is Average Precision (AP). However, there are several methods (e.g., 11-point interpolation, all-point interpolation) to compute AP, and different method might give different AP. Therefore, comparing two models' performances with AP might be incorrect if the APs for both models are not computed with the same method [Padilla et al., 2020]. To resolve the limitation, they proposed a standard implementation of AP with the programming language Python, which takes the bounding box description in two different formats and can produce the result in different variations of the AP metric including mean AP (mAP), AP@50, AP@75 and AP@50:5:95 using the 11-point or the all-point interpolations. However, the implementation was specific to the localization with the bounding box. Tharwat [2020] did a comprehensive review on different evaluation metrics, including single-valued scalar metrics (e.g., accuracy, precision, sensitivity, specificity, F-measure, geometric mean, Youden's index) and multidimensional graphical metrics (ROC, PR, and Detection Error Trade-off curves), used to evaluate binary and multi-class classification models. In their review, they detailed overviews of different metrics including the definitions, the relations among them, their way of calculation, the robustness of each metric against imbalanced data (e.g., PR curve is sensitive to the class imbalance and can be applicable to that kind of data, while ROC is not), and explanations of different curves in a step-by-step approach. They specifically detailed the methodology of computing ROC and PR curves comprehensively with the illustration of tied predicted score. With the ties in predicted scores, both ROC and PR curves can be optimistic, pessimistic, or expected.

According to the literature analysis done by Saito and Rehmsmeier [2015], the most popular metric in bioinformatics for binary classification including the classification on imbalance data is ROC. On ROC, Fawcett [2006] presented a detail tutorial by describing its basics, computing methodology, and AUC. McClish [1989] proposed a method to compute AUC at any portion of the ROC curve -which is called partial AUC- and to compare two such areas. According to them, most of the ROC curves from continuous or binormal[2] data are indistinguishable; one curve might be superior over the other for a particular range of FPR, while for the remaining range of FPR, it is not. Consequently, the AUC might be the same and indistinguishable. In the case of large test sets, however, the ROC curves of the two models can be indistinguishable for all ranges of FPR. In that case, the idea of partial AUC will not work. Apart form the McClish [1989]'s one, some other methods were proposed to compute partial AUC [Jiang et al., 1996; Wu et al., 2008; Bradley, 2014; Yang et al., 2019; Carrington et al., 2020]. The latest one was proposed by Carrington et al. [2020]. They proposed a new concordant partial AUC, which is the sum of the vertical and horizontal partial ROC-AUC, while previous methods only considered vertical partial. By defining concordant partial AUC, they also define partial c-statistics for ROC data. The c-statistic for a classifier is the proportion of times when the classification score for the actual positive is greater than the score for the actual negative, which is actually the AUC, i.e., AUC = c-statistics. According to them, the AUC has mathematical relationships to concordance/c-statistics, average TPR, and average TNR, however, none of the state-of-the-art partial AUCs including the McClish [1989]'s one, have the same three mathematical relationships that the AUC has. By including both vertical and horizontal parts of c-statistics into their concordant partial AUC, they maintained the same relationship with the three elements an actual AUC has. Although they demonstrated their concordant partial AUC worked on a small test set, which is biased to negative class, there is no guarantee that it will work on a large test set. Performance inseparability of multiple models on a large test set thus remains unsolved.

[2] Binormal distributions are the joint distributions over two independent variables which are normally distributed [Macskassy and Provost, 2004].

Yu [2012] proposed an alternative metric maned ROC surface (ROCS) to evaluate results on imbalance data, specifically for the data biased to negative class. The ROCS combines the original ROC and one of the ROC variants depicting FDR (False Discovery Rate)-TPR. It is a three-dimensional surface-based metric that expresses the relationship among TPR, FPR, and the True Discovery Rate (TDR)/1-FDR/precision. They also defined FDA-controlled AUC (FCAUC) for a pre-determined FDA as the area under the traditional ROC –which is actually a partial AUC defined by McClish [1989]- and Volume Under the Surface (VUS) to summarize their ROCS. Authors showed that their metric is efficient for gauging classifier performance on the class-skewed test set, specifically biased to negative class. The metric values of two models, however, can be inseparable when the test set is biased to positive class or being large due to the same reason we explain for the PR curve. Moreover, it is not straightforward to compare two models with this metric because of being a three-dimensional surface-based one.

Saito and Rehmsmeier [2015] and Cook and Ramadas [2020] rather suggested to use PR curve as an alternative to ROC. Considering the efficiency of the PR curve in handling imbalance data, some studies utilized it in solving different problems in different domains. For example, Fu et al. [2017] utilized PR curve in model selection by coupling it with the sparse regularized variable selection algorithm for handling imbalance data case. Sofaer et al. [2019] discussed the efficiency of the PR curve over ROC curve in accessing the rare species distribution models in biogeographical domain. According to them, PR-curve is useful in that domain because PR-AUC does not increase with the inclusion of many highly unsuitable locations within the study area, and secondly, PR-AUC reflects the ability of a model to guide surveys for new populations.

In contrast, some studies described the limitations of PR curve and proposed an alternative evaluation metric to PR curve. Among them, Flach and Kull [2015] suggested the Precision-Recall-Gain (PRG) curve by relating the enclose area of PR curve to the expected F-measure on a harmonic scale. According to them, the arithmetic mean that is used to compute the PR-AUC in different methods is methodologically wrong and instead of it, they proposed to use harmonic scale since the standard way to combine precision and recall into a single performance measure is through the F-measure, which is a harmonic mean of recall and precision. They assumed that the classifier predicting always positive as baseline, hence the minimum and maximum limit of recall and precision should be $[\pi, 1]$, where π is the skewness -the ratio of the number of the positive examples to the total number of examples- of the test set. Assuming this, they re-scaled the precision and recall using min-max normalization in harmonic scale thus compute precision- and recall-gain for their PRG curve. However, a model can be worse than the baseline thus limit should be [0, 1]. Oksuz et al. [2018] proposed a new metric specifically for object detection and localization named Localization recall precision (LRP) error which is composed of three elements related to localization, FPR, and FNR. The metric considered the inability of the AP, the PR-AUC, to distinguish very different PR curves, and the lack of directly measuring bounding box localization accuracy.

Some of the articles considered computing the PR-AUC and interpolating a point on the PR curve were challenging because of the non-linear relationship of the recall and precision; thus proposed a method to interpolate a point on the PR-curve. Among them, Davis and Goadrich [2006] are the pioneer. According to them, for a fixed data set, a point can be translated between ROC and PR spaces and linear interpolation is appropriate for ROC curve. Using this theory, they proposed a very first method to correctly interpolate a point on the PR space. In their method, they non-linearly interpolated a point on the PR space for discrete change in TP between two adjacent points (say A and B) in PR space. They computed the local skew, $skew = \frac{FP_B - FP_A}{TP_B - TP_A}$, i.e. computed the number of false-positive to have one true-positive, where (TP_A, FP_A) and (TP_B, FP_B) were the TP and FP pairs of the known points A and B, respectively. They created the TPs of new points as $TP_A + x$ for all discrete values of x such that $1 \leq x \leq (TP_B - TP_A)$. For all new TPs, they interpolated the corresponding FPs (FP_xs) by the Equation 5.1.

$$FP_x = FP_A + skew.x \tag{5.1}$$

Form the interpolated TPs and FPs, they computed the corresponding recall and precision on the PR curve. Once interpolation is done, they suggested to use trapezoidal rule (Eq. 2.10) to compute the PR-AUC. In [Keilwagen et al., 2014], authors reintroduced the method of interpolating a point on the PR space non-linearly using the linearity in ROC space. Instead of discrete change in TP, however, they applied continuous change. They however did not consider the tied values on the x-axis. Boyd et al. [2013], conducted an empirical analysis on different methods of computing PR-AUC and their confidence intervals, specifically in handling the tied scores of the x-axis (i.e., recall) of PR curve. According to them, the best method to compute the PR curve and its AUC is not readily apparent. They recommended the lower trapezoid, average precision, and interpolated median for estimating point of the PR curve and the binomial and logit methods for constructing interval estimates.

5.2.2 Mean curve computation for ROC and PR

Fawcett [2006] described two methods, the *vertical averaging* and the *threshold averaging*, to compute the mean ROC curve of a list of ROCs that belong to a list of trial-models.

In the *vertical averaging*, each ROC curve is treated as a function, ROC_i, such that $tpr = ROC_i(fpr)$, where tpr and fpr are the instances/values of TPR and FPR, respectively. Indeed, for a predefined list of equally spaced x-axis values (fpr), the corresponding maximum y-axis values (tpr) of each trial-model are considered. If the corresponding y-axis value is not available for any trial-model, the value is interpolated. The linear interpolation is applicable in this case since the x- and y-axes components for the ROC curve are linearly correlated [Davis and Goadrich, 2006]. The mean curve is then calculated as $\overline{ROC(fpr)} = mean[ROC_i(fpr)]$. It is also possible to compute the standard deviations of the y-axes values in vertical manner in order to draw confidence bars [Fawcett, 2006]. In the *threshold averaging*, a predefined set of thresholds is considered. For each of these thresholds, the ROC points (i.e., FPR and TPR) of all trial-model are searched and averaged vertically (y-axis value of ROC) and horizontally (x-axis value of ROC). The standard deviation can also be computed in both vertically and horizontally. The vertical averaging requires only single-dimensional (on y-axis values) computation and is usually utilized in the literature, while the other method requires two-dimensional (on both x- and y-axes values) computation. Among the two methods, the threshold averaging gives the information about the threshold, however still getting the threshold information for the interpolated points is not possible since they are not computed for any particular threshold.

The mean computation of the PR curves of several models, by contrast is tricky. To our knowledge, there is no precise method to compute the mean PR curve. Since the computation methods of the ROC and PR for the single model are the same [Saito and Rehmsmeier, 2015; Tharwat, 2020], the mean computation methods described for the ROC curve can be customized for the PR curve. The interpolation and some special cases, such as, start, end, and intermediate points with tied x-axis values, however, need to be considered.

Saito and Rehmsmeier [2017] proposed a tool to compute the mean PR curve with the programming language R. They did not mention however how to handle the tied recall at x-axis including at the endpoint (recall = 1.0). Thus, the endpoint of the PR-curve is not always correctly computed. Cook and Ramadas [2020] introduced a Stata module *prtab* to plot PR curves where they handled the tied predicted score of positive and negative examples. However, they did not handle the tied recall values and used linear interpolation, which is the wrong way to interpolate a point on the PR space, as shown in [Davis and Goadrich, 2006] and [Boyd et al., 2013]. Among the available tools to mean PR curve and PR-AUC computation, Saito and Rehmsmeier [2017] showed that theirs is the most accurate, however as we identified, the endpoint with the tied recall is not correctly computed.

After ordering the predicted score in descending order, the ground truth ranked list is $[1, 0, 1, 0, 0]$ for Model 1 and $[1, 0, 0, 0, 1]$ for Model 2. Model 1 is better than Model 2 since it achieves recall 1.0 without scanning all negative examples. Using Saito and Rehmsmeier [2017]' formula, both models will have the same precision, $2/5$ at the endpoint of the curve. In case of ties, it leads to misinformation at recall 1.0 when computing the mean curve. Indeed, that calculation requires to interpolate one precision per recall for tied recall cases (Model 1 has three precision values at recall 1.0). Even considering the recommended median of the precision values for tied recall 1.0, it will under-estimate Model 1.

Example 1: the problem of ties in [Saito and Rehmsmeier, 2017]

5.3 Limitations of ROC and PR curves

We describe the ROC and PR curves with their component metrics earlier (Section 2.5.2). This section describes their limitations that lead us to propose a new metric, PR-T.

5.3.1 Not always possible to retrieve threshold information

From the ROC (resp. PR) curve of a single model, it is possible to retrieve the threshold t_i for any point i on the curve. When different trials of the model are run, what we call *multiple trial-models* however, there is no guarantee of having the same thresholds for all trial-models for a given FPR. $fpr_1(t_1) = fpr_2(t_2)$, where $fpr_1(t_1)$ and $fpr_2(t_2)$ are the FPRs from two trial-models, for example, does not guarantee that threshold, $t_1 = t_2$.

While it is possible to calculate a mean point for each FPR, there is not a single threshold associated with that mean point which does not help in selecting the best threshold for an intended purpose (e.g. low false-negatives). This problem can be partially solved if the mean is calculated by the threshold averaging method without applying any interpolation. It is impossible to know the threshold for an interpolated point since the point is an approximation and is not computed for a particular threshold.

5.3.2 Can be optimistic or pessimistic

Saito and Rehmsmeier [2015] and Tharwat [2020] described the method to compute the ROC and PR curves. In those methods, the test examples are ranked in descending order of their predicted score. Every predicted score is then considered as a threshold for which a confusion matrix can be created, and a point on the curve space can be computed.

Those papers show that due to the tied predicted scores for negative and positive examples, both ROC and PR curves can be optimistic, pessimistic, or expected for the same model and test set. During the computation of the curve, if all the positive examples with the predicted score s are ordered before all the negative examples with the same score, the curve will be optimistic. Likewise, if the examples are ordered the other way around, the curve will be pessimistic, and if they are ordered randomly, the curve will be as expected [Tharwat, 2020]. Both ROC and PR curves thus are strongly impacted by the order of presentation of examples with tied predicted scores.

5.3.3 Artificial inflation or deflation

In the ROC and PR curves, if two adjacent points are distantly separated, interpolation is required to approximate intermediate points on the curve and to obtain small enough intervals on the x-axis for AUC computation [Davis and Goadrich, 2006; Saito and Rehmsmeier, 2017] (see Section 2.5.3). As a consequence, the interpolation might give an artificially high or low indication of performance when the real points from where the interpolated points are derived are high or low [Carrington et al., 2020].

Along with the AUC computation, interpolation can be required to compute the mean curve of several curves from trial-models.

5.3.4 Performance inseparability of models

Table 5.1: **Example of confusion matrices of two models, m1 and m2, for a large and a small sized test sets.**

Size of test set	Model	Confusion matrix	
Large	m1	TP=80,000	FN=400
		FP=4,000	TN=10,000,000
	m2	TP=80,200	FN=200
		FP=2,000	TN=10,002,000
Small	m1	TP=8,000	FN=400
		FP=4,000	TN=100,000
	m2	TP=8,200	FN=200
		FP=2,000	TN=102,000

In the real world, if it is possible to collect a large number of data examples of a particular class, it can be assumed that most of the collected examples have some common features. Because of these common features, most of the examples can be easily classified. For example, in a metastatic lymph-node image data for cancer detection task, most of the image patches that belong to the non-region-of-interest (non-ROI) or non-lymph-node class are very similar to each other; thus, they are easy to classify. Only the patches with inter-class similar regions are difficult to classify and might cause false predictions according to our findings in [Reshma et al., 2021]. In this kind of situation, the number of true predictions (TP or TN) is much higher than the number of false predictions (FN or FP).

According to the mathematical formula (Eq. 2.1, 2.2, 2.4) of the TPR (or recall), precision and FPR, if FN<<TP, FP<<TP, and FP<<TN, even with the

Table 5.2: **Example of performance inseparability of two models, m1 and m2, in terms of TPR and FPR for a large sized test set.** We considered two different sized test sets, large and small. Here, $\triangle^*$ means difference of the performances of two models in terms of a metric and indicates if the performances are separable or not, where * can be replaced by TPR, Prec, or FPR; the TPR, Prec, and FPR are computed by placing the value of TP, FN, FP, and TN from Table 5.1 at Equation 2.1, 2.2, and 2.4.

Size	Model	TPR	$\triangle$TPR	FPR	$\triangle$FPR	Prec	$\triangle$Prec
Large	m1	$\frac{80{,}000}{400+80{,}000}$	.003	$\frac{4{,}000}{4{,}000+10{,}000{,}000}$	.000	$\frac{80{,}000}{4{,}000+80{,}000}$	.024
	m2	$\frac{80{,}200}{200+80{,}200}$		$\frac{2{,}000}{2{,}000+10{,}002{,}000}$		$\frac{80{,}200}{2{,}000+80{,}200}$	
Small	m1	$\frac{8{,}000}{400+8{,}000}$	.024	$\frac{4{,}000}{4{,}000+100{,}000}$	.019	$\frac{8{,}000}{4{,}000+8{,}000}$	.005
	m2	$\frac{8{,}200}{200+8{,}200}$		$\frac{2{,}000}{2{,}000+102{,}000}$		$\frac{8{,}200}{2{,}000+8{,}200}$	

significant difference in false predictions (i.e., FN or FP), performances of two models might be indistinguishable in terms of these metrics. We present such a situation in the row for the large test set of Table 5.1 (the 1^{st} row), where the number of positive examples (80,400) and negative examples (10,004,000) both can be considered as high[3]. In such case, if the number of false predictions (FN, FP) is much lower than the true predictions (TP, TN) similar to the row for Large in Table 5.1, performance inseparability of models can happen (see Table 5.2: row for Large). Here, the model m_1 and m_2 are inseparable (i.e., $\triangle^*$ is too small) in terms of TPR and FPR thus will be inseparable in terms of ROC, while they are more separable in terms of precision thus will be more separable in terms of PR. For example, if the number of negative examples is $10,004,000$, FP generated by models, $m1$ and $m2$ are 2000 and 4000, respectively, the performance of the models will be indistinguishable according to FPR (too small $\triangle$FPR) since $2000/10,004,000 \approx 4000/10,004,000$, although there is a significant difference in FP ($FP(m2) - FP(m1) = 2000$) for the two models. If the test set is small, for the same number of false predictions, however, performances of models can be separable (see Table 5.2: row for Small).

The precision also will be inseparable if we flip the number of examples between classes, i.e., if the test set is biased to positive class instead of negative

[3] The way of calculating the number of positive and negative examples from a confusion matrix is given in Table 2.4.

class, thus the FP is comparatively much smaller than the TP for the same test set.

5.4 A new multi-threshold-based evaluation metric

By tuning the threshold of predicted scores, it is possible to increase or decrease the false predictions; thus, presenting the evaluation metric as a function of the continuous threshold should be beneficial to facilitate application-dependent optimal threshold selection [Freeman and Moisen, 2008]. PR-T curves follow this intuition.

5.4.1 The PR-T curves

The PR-T metric we develop presents the trade-off between recall and precision as a function of the continuous threshold.

In PR-T, the precision and recall are presented in two separate curves (y-axis) while thresholds correspond to the (x-axis). In that way, the performance on the positive class (by recall) and error from the negative class (by precision) are clearly expressed; unlike the ROC and PR curves, the performance on the under-represented class is perceivable in an imbalance test set, thus it is not misleading the analyst. Because of this representation also, PR-T does not require interpolation, thus, artificial inflation or deflation of curves is impossible and PR-T mean curve and AUCs can be easily calculated without error. PR-T is easy to read and interpret as well. Algorithm 1 shows the method of computing the lists of recall ($RECALL$[4]) and precision ($PREC$[4]) points on the PR-T curves. In this method, we scale up the predicted score of a model within 0.0 to 1.0 by applying min-max normalization. The normalization makes the predicted scores of different models commensurate. Then, $\forall t_i \in T$ (T is a list of predefined thresholds), we compute a confusion matrix by considering all the examples having predicted score $\geq t_i$ as predicted positive otherwise predicted negative. For the computed confusion matrix, we find the precision, $prec(t_i)$ and recall, $recall(t_i)$ and store in the lists $RECALL$ and $PREC$, respectively.

[4] Note that, here, recall, precision: metric names; $recall$, $prec$: single values; $RECALL$, $PRECISION$: lists of values

Once the $RECALL$ and $PREC$ are computed with Algorithm 1, plotting t_i at the x-axis and corresponding $prec(t_i) \in PREC$ at the y-axis provides the precision curve. Likewise, the recall curve for $RECALL$ can be drawn.

Unlike for ROC and PR curves computations, there is no sorting step of the test examples. The order of examples thus does not affect the results, and there is no chance of being optimistic, pessimistic. The curves will be always the same as expected for the same model and test set.

5.4.2 Area under the PR-T curve

The general trapezoidal rule as presented in Equation 2.9 can be applied to each interval of the x-axis to compute the AUC of that interval. Summing up AUCs of all intervals results in the final AUC of a curve. Specifically, the trapezoidal rule is applied to each interval of the x-axis for precision and recall curves separately to compute the precision-AUC and recall-AUC in a similar manner presented in Equation 2.10. Algorithm 2 shows the procedure of computing those.

The higher the size of T, $|T|$ ($N = |T|$ in Eq. 2.10), the closer the computed AUC to the actual AUC; $|T|$=101 could be a good trade-off between computing time and accuracy of the obtained AUC.

Unlike the ROC and PR AUCs, the PR-T AUCs are neither affected by the ties of the predicted score nor it has ties in x-axis values. Due to considering all the test examples at once, the order of positive and negative examples with tied predicted score does not affect the PR-T AUCs. Since the x-axis of the PR-T is threshold, and for each threshold only one confusion matrix can be computed, hence there is no tie in the x-axis values. Since no interpolation is required to compute the PR-T AUCs, the AUCs are real and reliable than the ROC-AUC and PR-AUC.

5.4.3 Solution of Pareto front for PR-T curves

Finding the optimal points on the PR-T curves is a two-objectives (recall and precision) optimization problem like the PR curve. Any application prefers to maximize both recall and precision, although which objective is preferable to the other depends on the application type. Pareto fronts are the set of probable optimal points. There may be an infinite number of Pareto optimal points constituting the Pareto optimal set or Pareto fronts for any given

Algorithm 1: Method for generating PR-T points

```
Input: $L$ be the list of ground truth consists of $0$ (if negative) and $1$
       (if positive); $pos$ and $neg$ be the number of positive and
       negative examples, respectively in $L$;
       $T = \{t_i : 0.0 \leq t_i \leq 1.0\}$ be the set of equally spaced
       thresholds, where $i = 1, 2, \cdots, 101$ (or 1001) when stride,
       $\triangle t = 0.01$ (or 0.001); $S = \{s_j : s_j \in \mathbb{R}_0^+\}$ be the set of
       predicted scores by the classifier, that is to say the probability
       that the example $j$ is a positive example, where
       $j = 1, 2, \cdots, |L|$ and $|L|$ is the size of $L$.
Output: $RECALL$, a list of recall points; $PREC$, a list of precision
        point
Require: $pos > 0$, $neg > 0$
$S^n \leftarrow S$ normalized by min-max normalization
for $i = 1$ to $|T|$ do
    $L_i^p = [\,]$    // an empty list to store predicted class
    for $j = 1$ to $|L|$ do
        if $s_j^n \in S^n \geq t_i$ then
            $L_i^p[j] \leftarrow 1$
        else
            $L_i^p[j] \leftarrow 0$
    $TP_i, FN_i, FP_i, \_ \leftarrow$ CONFUSION_MATRIX($L, L_i^p$)    // Get
     confusion matrix for $t_i$
    $PREC[i] \leftarrow \frac{TP_i}{FP_i+TP_i}$, $RECALL[i] \leftarrow \frac{TP_i}{FN_i+TP_i}$

Function CONFUSION_MATRIX($L$, $L^p$):
    $TP \leftarrow FN \leftarrow FP \leftarrow 0$    // Initialization to 0
    for $j = 1$ to $|L|$ do
        if $L[j] == 1$ then
            if $L^p[j] == 1$ then
                $TP \leftarrow TP + 1$
            else
                $FN \leftarrow FN + 1$
        else
            if $L^p[j] == 0$ then
                $TN \leftarrow TN + 1$
            else
                $FP \leftarrow FP + 1$
    return $TP, FN, FP, TN$
```

Algorithm 2: Method for computing PR-T AUCs

```
Input: $T = \{t_i : 0.0 \leq t_i \leq 1.0\}$ be the set of equally spaced
       thresholds, where $i = 1, 2, \cdots, 101$ (or 1001) when stride,
       $\triangle t = 0.01$ (or 0.001); $PREC$ and $RECALL$ be the list of
       precision and recall computed for $T$ by Algorithm 1.
Output: precision-AUC, the area under the precision curve;
        recall-AUC, the area under the recall curve.
Require: $|T| \geq 101$
precision-AUC ← recall-AUC ← $0$                    // Initialization
for $i = 1$ to $(|T| - 1)$ do
    precision-AUC ← precision-AUC + TRAPEZOID_AREA($prec(t_i)$,
      $prec(t_{i+1})$, $\triangle t$)
    recall-AUC ← recall-AUC + TRAPEZOID_AREA($recall(t_i)$,
      $recall(t_{i+1})$, $\triangle t$)

Function TRAPEZOID_AREA($base_1$, $base_2$, $h$):
    return $h\frac{base_1+base_2}{2}$
```

problem [Marler and Arora, 2004]. The problem of finding the optimal point of an evaluation metric, however, seeks a single final point usually. One of the methods to find the final optimal point could be converting the two-objective optimization problem to a single objective optimization problem [Giagkiozis and Fleming, 2014]. The conversion is done by assigning weight to each objective: $f = w_1 * recall + w_2 * prec$, where w_1 and w_2 are the preferred weights for recall and precision, respectively. The weight of each objective depends on the corresponding application and the analyst.

5.4.4 PR-T properties and experimental design overview

Imbalance Class Case and Model Separability with PR-T. An evaluation metric should be able to separate models when they differ. To measure this ability, we trained two models En and Ep considering two oppositely distributed training subsets so that the models will have different behaviors on the test set that the evaluation measures should reflect. Details are provided in Section 5.5.1 for the experimental design and in Section 5.7.1 for the results and discussions.

PR-T mean curves. In the case of PR-T curves, the mean curve calculation

Table 5.3: **Experiment design with two different distributions of classes in the training data**: experiments with negative-biased (*En*) and positive-biased (*Ep*) class distributions with a total of 4 units ($\mathbb{U}$) of examples in the training set of each experiment.

Experiment ID	Distribution	Class ratio (Negative:Positive)
En	Negative-biased	$3:1$
Ep	Positive-biased	$1:3$

is straightforward, as well as the mean AUC. It consists in repeating the Algorithm 1 and 2 for all trial-models and compute their threshold-wise average and standard deviation for the precision and recall. Because the same thresholds are used for all curves, the mean is directly applicable without interpolating any point. The same holds for their AUCs. An illustration of the power of PR-T in the case of multiple trials model is presented in Section 5.7.2, while the experimental design is presented in Section 5.5.2

Interpretability. The precision and recall are presented in separate curves in PR-T, hence class-wise performance or false predictions are easy to interpret. If the recall curve goes upper than the precision curve, that means the model produces higher FP than FN and the other way around (See Section 5.7.3).

5.5 Experimental setup

5.5.1 Imbalanced training set - En and Ep

To illustrate the efficiency of the evaluation metrics in assessing predictive models' performance, we trained two different types of models. The models were trained with two oppositely distributed training data set so that their prediction could be separable to each other. Specifically, we considered the negative-class-biased distribution with an over-represented negative class and the positive-class-biased distribution with an over-represented positive class in the training data. This opposition is beneficial to check if the evaluation metric can separate the model performances when they are opposite indeed. Table 5.3 illustrate the experiment design for the models' training. In *En*, we considered a $3:1$ ratio of negative and positive classes in the training data; it is thus negative-biased, whereas it is $1:3$ in *Ep*, thus positive-biased. This experimental design also illustrates the case of imbalanced class distributions.

Note that the experiment design of En and Ep are similar to that of the E2.b and E2.c, respectively presented in Table 4.6 in the previous chapter. The total number of examples in each experiment and the unit ($\mathbb{U}$) design methodology are the same as described in Section 4.5.1.1. Specifically, in each experiment, a total of (3+1=) 4 $\mathbb{U}$ of examples were considered. The size of $\mathbb{U}$ $= n/3$, where n is the size of the smallest class (usually, the positive class) in an original training set of a data set, and the 3 is its largest presence in the considered training subset's class ratios (Table 5.3).

Once the distribution (i.e., ratio) of the classes in training data was fixed, we randomly selected training examples from the original training set of the data set by maintaining that distribution as we did in the previous chapter. We then trained a Convolutional Neural Network (CNN) model.

5.5.2 Multi-trial models

In the literature, ROC and PR metrics are under-explored for the mean curve computation of several trial-models. It is, however a common practice to train several models for the same task and taking the mean result of all trials; this can show the robustness of the model. We consider the case of multiple trial models as follows: we trained 10 trial-models for each experiment of En and Ep. Each trial-model was created by training the CNN with a different subset of the training data. The subset was selected randomly from the original training set of the data set by maintaining the class ratio. We then calculated the mean and standard deviation of all trial-models for each experiment (En and Ep) for ROC, PR, and PR-T and associated AUCs.

5.5.3 Data Sets and pre-processing

To test the efficiency of the evaluation metrics on a very large and a moderate-sized data set, we utilized two different data sets for two real-world problems: 1) histological image data sets, PCam and CAMELYON16, for the cancer detection task, 2) natural image data set, CIFAR10 [Krizhevsky et al., 2009], for the automobile detection.

Cancer detection task. For the first task, cancer detection, we used PCam and CAMELYON16 data sets in a similar manner described for the classification task in Section 4.5.2.1.

Object detection task. For the object detection task, we utilized CIFAR10 [Krizhevsky et al., 2009], a popular toy data set of natural images. It is a multi-class (10 classes) data set with a training set of 50,000 images and a test set of 10,000 images of size 32×32 pixels. All 10 classes have the same number of images in the data set. To follow our purpose, we converted it to a binary class and imbalanced data set. We considered one of the 10 classes, the “automobile" class, as the positive class and the remaining classes as the negative class. The “automobile" class can be considered as difficult since there are three other classes (“airplane", “ship", and “truck") which share some visually similar features.

After converting the data set to a binary one, the ratio of the positive and negative classes is 1:9 in both training and test sets. In the training set, the positive class (automobile) consists in 5,000 images; we thus considered $1\mathbb{U}$ as (5,000/3=) 1,666. The training subset size was (1,666*4=) 6,664, from where 20% was used as validation. We selected the negative classes examples such that each negative class has an equal number of examples.

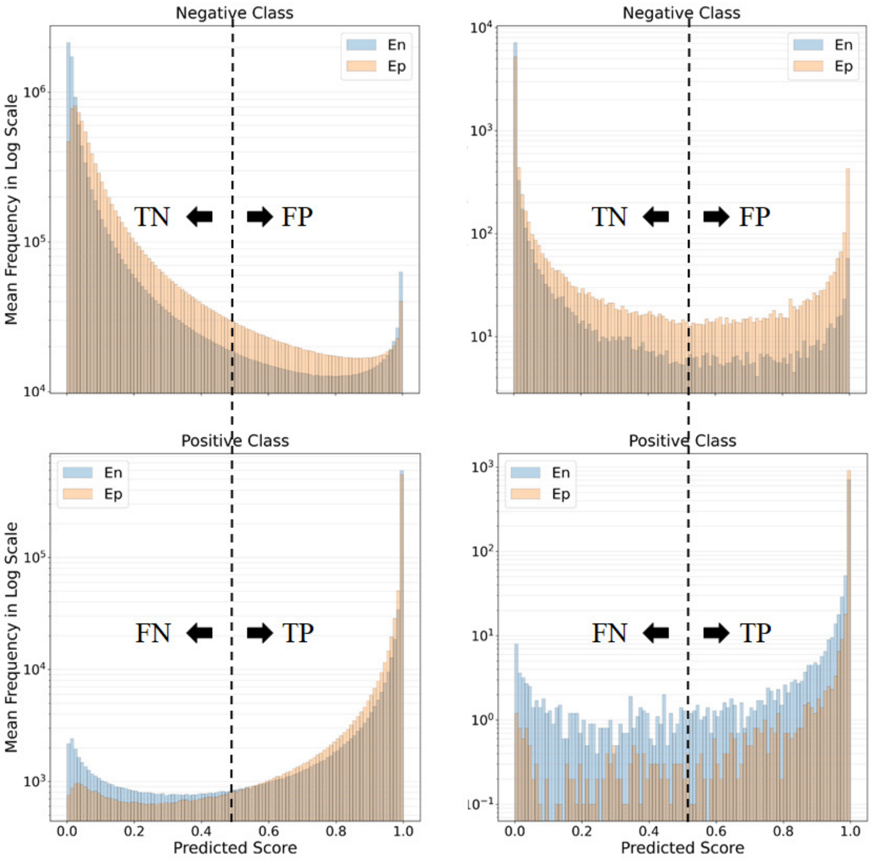
5.5.4 Hyper-parameter settings

We focused on the evaluation metric thus utilized off-the-shelf CNN architectures and hyper-parameter settings.

Cancer detection task. For the network architectures and other hyper-parameter settings of this task, we refer to Section 3.2.2.2.

Object detection task. For the object detection task, we used a pre-trained ResNet50 model from the Keras [Chollet et al., 2015], i.e., we utilized transfer learning since the training subset taken from CIFAR10 was small to train a model from scratch. In Keras, the model was pre-trained with the ImageNet [Deng et al., 2009] data set, which consists of natural images like CIFAR10. Moreover, using Keras's *ImageDataGenerator* function, we utilized applicable data augmentation techniques for natural images, such as random rotation (range 15), width, and height shift (range 5.0/32), and horizontal flip. The model was fine-tuned by applying the early stopping scheme to stop training before overfitting with mini-batch size 32. As an optimizer, we utilized the popular default, Adam, with a very small starting learning rate of $1e^{-4}$. As loss function, we utilized categorical cross-entropy. If the validation loss did not change for a maximum of 10 epochs, the learning rate was reduced by a factor of 0.3.

5.6 Pre-assessing the models with predicted score statistics



(i) train: PCam, test: CAMELYON16 (ii) train: CIFAR10, test: CIFAR10

Figure 5.1: **Illustration of the model's predicted scores (x-axis) in the linear scale and the corresponding mean frequencies of test examples in the log scale (y-axis).** (i) is for the cancer detection task, where models were trained on the subset of PCam, and tested on the CAMELYON16 test set, and (ii) is for the automobile detection task, where models were trained on the subset on the CIFAR10 and tested on the test set of same data set. Here, top row: for the actual negative class, bottom row: for the actual positive class, dashed black line: a representative threshold, left side of the black line: predicted negative class, and right side of the black line: predicted positive class. Each column resembles a confusion matrix.

To have an intuitive idea about the superiority of one model over the other, we assessed the models according to their predicted scores before assessing them with evaluation metrics in the next section. For this purpose, in Figure 5.1, we presented the predicted scores of models on the x-axis as 101 intervals in linear scale and the corresponding mean frequencies of examples on the y-axis in logarithmic scale. In a logarithmic scale, equal spaces on the plot indicate the same rate of change, unlike in a linear scale where equal spaces mean the same difference. For example, in the y-axis of the lower-left plot from the point 10^3 to 10^4, the difference in frequencies is (10,000-1,000=) 9,000, and the rate of change in frequencies is (10,000/1,000=) 10, where from the point 10^4 to 10^5, the difference is (100,000-10,000=) 90,000 and the rate of change is (100,000/10,000=) 10, i.e., the rates of change are the same while the differences are not.

The mean frequencies were computed from 10-trials of each experiment (En, Ep). Here, in the left column (i), the plots were for the cancer detection task; in the right column (ii), the plots were for the automobile detection task. In both tasks, the top row presents the histogram for the actual negative class, while the bottom row presents the actual positive class. The dashed black line indicates a representative threshold to perceive the true and false predictions by the models. The left side of a threshold line indicates the predicted negative class, while the right side indicates the predicted positive. The presentation of each column is analogous to the confusion matrix presented in Table 2.4.

While plotting such a histogram, an ideal model would have a plot where all the negative examples (top row) are on the left side of the threshold line (dashed black line) and all the positive examples (bottom row) are on the right without any overlap between two classes [Freeman and Moisen, 2008]. While looking at the histograms, however, it is not the case, i.e., the models have an overlap of positive (bottom row) and negative (top row) classes throughout the x-axis. In other words, the models produced false predictions. For both tasks, En (histograms in blue) produced higher FN (bottom row and left sides of the threshold lines), while Ep (histograms in orange) produced higher FP (top row and right sides of the threshold lines) at almost all positions of the threshold line (i.e., while moving the dashed black line at any position left or right), that is, the type of false prediction generated by each model is apparent. We will see in Section 5.7.3 that the same is reflected in the PR-T curves.

In Figure 5.1, one model (Ep) produced more FN, while another model

(En) produced more FP than the other, which implies that the models were indeed different from each other. If we consider two different threshold lines for two different models, however, the number of false predictions of both models could be the same. For example, if we move the threshold line for the Ep model to the right enough and keep the threshold line for the En model in the current place, it is possible to have the same FN with different FP by the two models. If the difference between FP is not high enough in comparison to the number of true predictions, however, the models' performance could be comparable. According to the logarithmic scale in the y-axis, the true predictions (TP and TN) were indeed much higher than the false predictions (FP and FN). Theoretically, thus, the difference in results -computed with the fraction of true and false prediction-based metrics- of two models will be very small, that is, the performance of models will be inseparable. Intuitively, this phenomenon usually happens in ROC and PR curves, where the metric values at the x-axis are kept common for all models.

Moreover, if we consider the logarithmic scale of the y-axis, the generated FP is much higher than the FN. Consequently, En is a better model than Ep while considering both types of false prediction at a particular threshold line because of producing fewer FP than Ep. We will see in Section 5.7 that if the evaluation metrics can reflect the phenomenon of one model's superiority over the other.

5.7 Result and discussion

5.7.1 PR-T curves appropriately distinguish predictive models

To compare the utility of PR-T curves with ROC and PR curves as evaluation metrics of classification model, we investigated whether they could distinguish between the performances of two different predictive models En and Ep. We considered the two imbalanced test sets CAMELYON16 (large size with 10.45M patches) and CIFAR10 (moderate size with 10K patches). The results of the Ep and En models are plotted as the ROC, PR, and PR-T curves (See Figure 5.2).

When the results were analyzed by the ROC curve, there was little apparent difference between the performance of the Ep and En models either

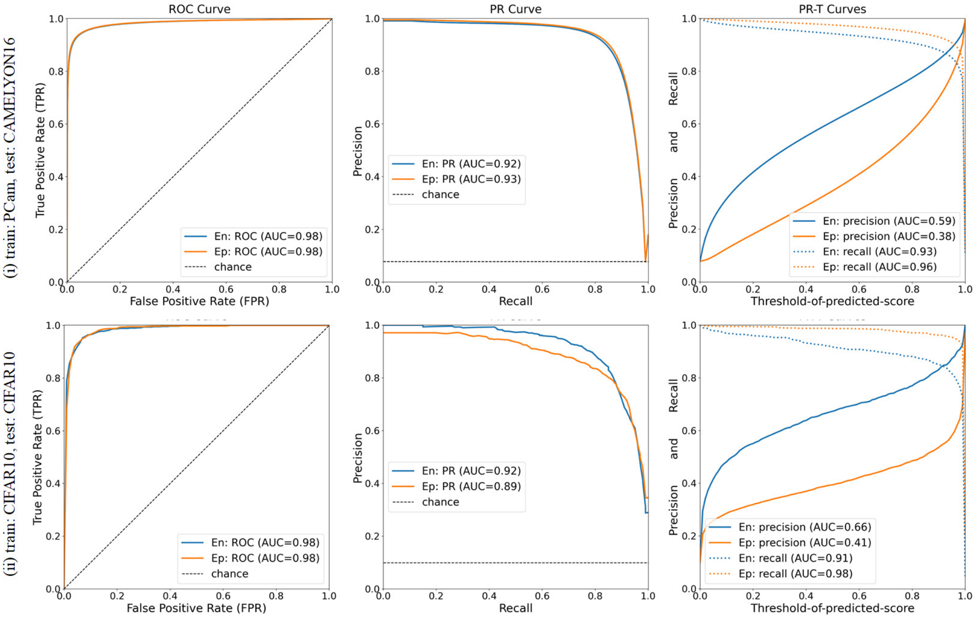


Figure 5.2: **PR-T curves show better than ROC and PR curves the different performances of two predictive models.** The models were trained either on the PCam data set and tested on the very large CAMELYON16 test set (top row), or were trained on the CIFAR10 training set and tested on the moderate-sized CIFAR10 test set (bottom row). The Ep models (orange) were trained on data subsets biased toward the positive class (e.g., cancer); the En models (blue) were trained on data subsets biased toward the negative class (e.g., non-cancer). The performances of the models were evaluated by ROC curve (left), PR curve (middle), or PR-T curves (right). Their corresponding AUCs are presented in parenthesis.

on the very large data set (Figure 5.2 (i)) or on the moderate-sized data set (Figure 5.2 (ii)). When the results were analyzed by the PR curve, the different performances of the Ep and En models could be distinguished to some extent on the very large data set and much better on the moderate-sized data set. When they were analyzed by PR-T curves, by contrast, the performances of the two models were very clearly different both for the very large and the moderate-sized data sets.

Considering recall (dotted lines) on the PR-T curves, the Ep model (orange) goes higher than the En model (blue); this indicates that the Ep model produced less FN than did En. Considering precision (solid lines), the En model (blue) goes higher than the Ep model (orange), which indicates that the En model produced less FP than did the Ep model. This is an expected results since En is trained with more negative examples, it is likely to be less effective on recall. Both ROC and PR curves however fail to properly show this result, while PR-T curves clearly show it.

On PR-T, we can also observe that the precision curve for En (blue) is much higher than the one for Ep (orange) while the recall curves are not very different. This indicates the En model is performing better than the Ep model when considering precision recall trade-off and this hold for both data sets. ROC is unable to show this difference. PR hardly shows this difference for the large test set (CAMELYON16), while a little for the moderate size test set (CIFAR10).

These results are consistent with the pre-assessment of models with the predicted score statistics in Section 5.6.

5.7.2 PR-T mean curves

To complete the previous result, we considered the case of 10 trial-models and the mean curves on the same two data sets. When the results were analyzed by the ROC curve, there is no apparent difference between the various trials of a given model since its standard deviation is very low. This holds for both data sets (See Figure 5.3 left part). We can observe that PR curves are better in that perspective where the difference between trials or the absence of difference slightly appear on the orange and blue shadings that represent standard deviation on the various trials. On the contrary, using PR-T, the differences between the trials -or here the standard deviation- are clear. In the case of CAMELYON16 the standard deviation considering the various trials is large while there is not much difference on CIFAR10. This was also an expected results since CAMELYON16 is very large but that ROC and PR hardly show.

As an additional information, we can also observe that the differences across trials on CAMELYON16 are larger for precision (solid line) than recall (dotted line) and larger for En (blue) than for Ep (orange). This analysis can help the designer in understanding the results s/he obtain.

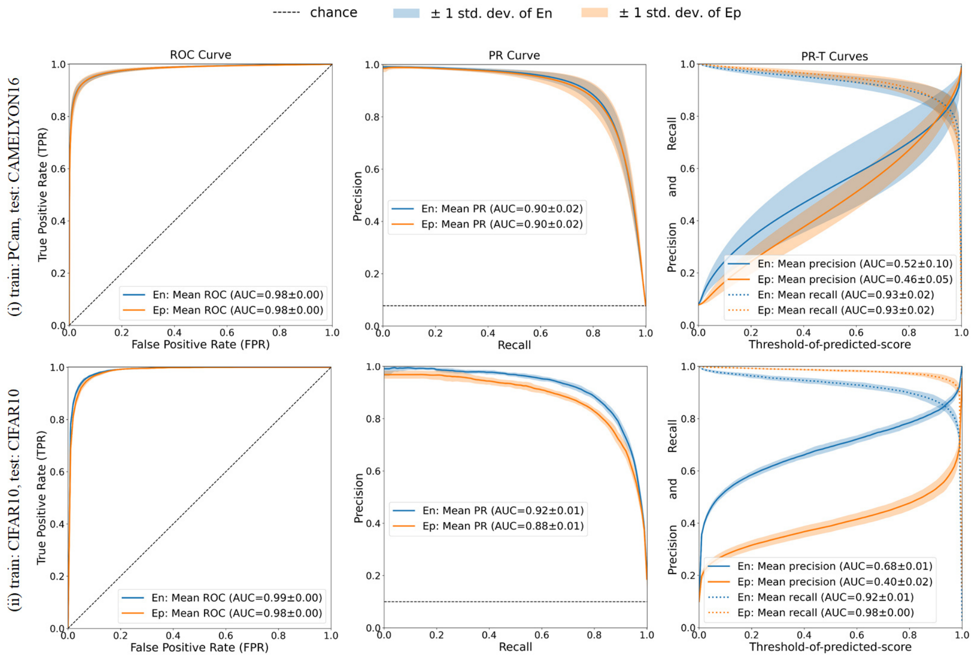


Figure 5.3: **Mean PR-T curves along with standard deviation appropriately summarize results and show better than mean ROC and PR curves the different performances of the trials.** The models, data sets, measures and notations are the same as in Figure 5.2 unless mentioned. 10-trials were used, where each trial differs by the training subset used ((See Section 5.5). The solid and the dotted lines indicate corresponding mean curves from 10 trial-models; the orange and blue shading indicates the standard deviation.

5.7.3 PR-T curves are more informative than the PR curve

PR curves should be preferred to ROC since it expresses the trade-off between the recall and precision in where by definition, the true predictions from the negative class (i.e., TN) are excluded [Sofaer et al., 2019]. Our results in Sections 5.7.1 and 5.7.2 also consistently indicate the superiority of PR on ROC.

The PR-T can also express the recall precision trade-off with other advantages. We illustrated those advantages by taking as an example, the PR and PR-T curves produced by using the En model trained on the negative class-biased subset of the CIFAR10 training data set and used to predict the CIFAR10 test set (Figure 5.4).

When considering recall precision trade-off, it is usual that the end-user

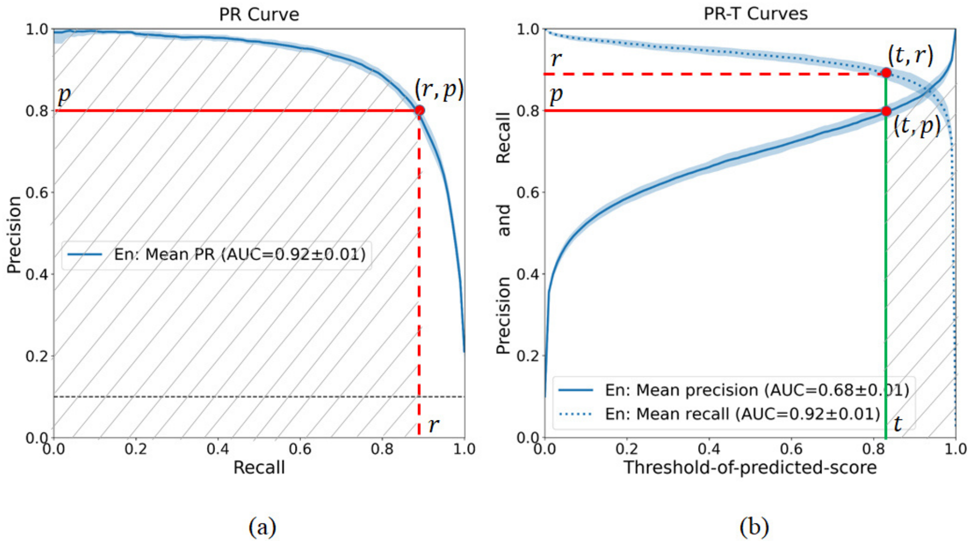


Figure 5.4: **PR-T curves are more informative than the PR curve**. The curves were generated using the same settings as in Figure 5.3; the notations are also the same. While there may be different thresholds used for single point (r,p) on PR mean curve, there is a single and identifiable threshold in the case of PR-T for that point (r,p). The areas under the PR curve in (a) and precision curve in (b) shaded with the gray line segments indicate the areas occupied by the precision for recall range 0.0 to 0.9.

needs to consider some specific point from the PR curve, often with the threshold information. Let us notate that point (r, p), where r is the recall and p the precision (See Figure 5.4 (a)). The threshold information of that point on the PR curve can only be obtained if the point is not an interpolated one. By contrast, the corresponding point is also quite easy to obtain on the PR-T curves from the points (t, r) and (t, p) along with the threshold information t (See Figure 5.4 (b)). Since PR-T does not require interpolation, and its computation (along with the mean) is threshold-oriented, it is always possible to obtain recall and precision with the corresponding threshold information from PR-T.

PR-T curves also provide information about the type of false prediction a model tends to generate. In the PR-T curves in Figure 5.4(b), when the precision curve (solid line) is below the recall curve (dotted line), by definition, it indicates that the model produces more false positives than false negatives. When the precision curve is upper than the recall curve, this indicates that the model produces more false negatives than false positives. It is difficult to extract this information from the PR curve (Figure 5.4(a)).

A further advantage of PR-T curves over PR curves is related to the area under the curve (AUC). In the example PR curve (Figure 5.4 (a)), the PR-AUC value misleads the analyst who can observe a high value (0.92). This value is due to the high precision at low levels of recall (which is insignificant in most of the applications). PR-T has not this drawback. We illustrated this on an example (See Figure 5.4). The areas under the PR and precision curves shaded with the gray line segments indicate the areas occupied by precision for recall in the range of 0.0 – r, where $r = 0.9$. In the case of PR curve (left-side part of the) the AUC is obviously exaggerated by the high values of precision at low recall (e.g. precision is about 1.0 when recall is below 0.3). This is not the case for PR-T where the grey part of AUC occupied by precision for recall $r < 0.9$ is not artificially inflated. PR-T is thus more appropriate to compare different models.

The interpolation required to approximate intermediate points on the PR curve; this might also give an artificially high indication of performance when the real points from where the interpolated points are derived are high. PR-T curves, by contrast, do not contain interpolated points: the values of precision and recall are real measures given the number of thresholds ($|T|$) used large enough. AUCs calculated from the PR-T curves are thus more reliable than PR-AUC and provide a true evaluation of the performance of a model.

The limitations we just mentioned for the PR curve also hold for the ROC curve for similar reasons. PR-T curves provide more information to evaluate a classification model than the PR and ROC curves and that the information PR-T curves encompass is more reliable.

5.8 Re-evaluation of representative results from the previous contribution with PR-T curves

In the previous chapter (Chapter 4), we evaluated the result with ROC, PR, precision, and FPR curves, where the last two curves were the threshold-wise representation similar to the PR-T curves. Since in the previous chapter, the hypotheses were based on FP, we selected the FP-oriented metrics -the precision and FPR- there to emphasize FP. Indeed, the advantage of precision and FPR curves encourages us formalizing the representation and proposing the PR-T curves as an alternative to ROC and PR curves. In this section, we re-evaluate and discuss a representative result from that chapter with PR-T curves.

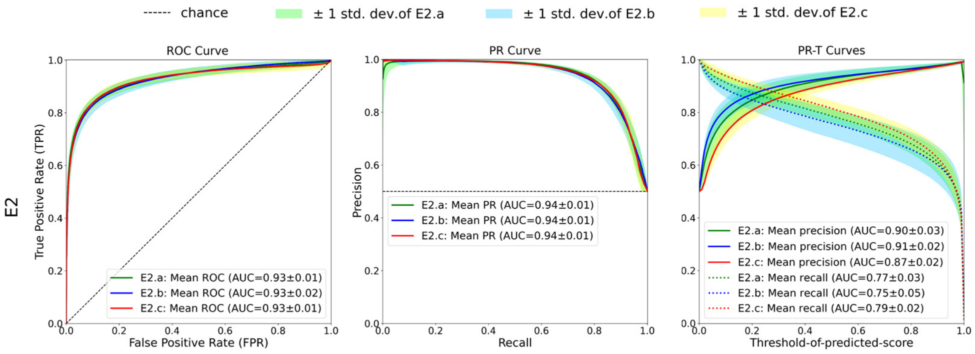


Figure 5.5: **The result is consistent with the findings in Chapter 4 and Section 5.7 - on PCam.** Mean curves with standard deviation (std. dev.) of experiments for balanced distribution (E2.a), for over-represented ¬ℂ (E2.b), and for over-represented ℂ) (E2.c) on PCam. The standard deviation is represented by green/blue/yellow colored shading.

.

To apply the PR-T curves in a use-case, we consider the hypothesis H2 result on the PCam test set; the test set is an example of a balanced one. We choose it since we already discuss the result on two imbalanced test sets in Section 5.7.

When considering the hypothesis H2 (Over-representing the ¬ℂ class in the training set reduces false-positives during cancer detection) from the previous chapter, we can test it with the precision curve of PR-T (Figure 5.5: rightmost part) since it reflects the FP. The conclusion is clearly the same as described in Chapter 4 since the precision curve in Figure 5.5 is the same one of the precision curve in Figure 4.13(iii) presented in that chapter. The conclusion is that PR-T curves also confirm the H2 hypothesis; for the classification task with balanced test and validation sets, the ¬ℂ-biased (E2.b) distribution produced fewer FPs than the two other distributions.

Additionally, the PR-T curve in Figure 5.5 also reflects the FN by the recall curve. The recall curve of E2.b (dotted blue curve) is the lowest among the three recall curves but with the highest standard deviation. It says that the E2.b produces the least FP with the highest FN, although the high standard deviation says that the FN production is less certain than that of two others

(E2.a and E2.c). One of the reasons could be the number of positive examples in the training set. To keep the same total number of training examples in all the experiments (for a fair comparison), we did not use all the positive examples (class $\mathbb{C}$) in E2.b, while in E2.a, we utilized little more and in E2.c almost all positive examples, which reduces the recall in E2.b. The reason for the high standard deviation could be of selecting a small subset (5k) of positive examples from a larger set (15k) randomly. It also indicates that all the cancer patches are not equally effective in CNN training.

The result on the balanced test set in Figure 5.5 also reflects another interesting finding. In Section 5.7, we showed the ineffectiveness of the ROC and PR curves in the cases of imbalanced test sets. In this section, we can see that the same conclusions hold even for a balanced test set. The ROC (Figure 5.5: leftmost part) and PR (Figure 5.5: middle part) curves fail to separate the performance of models for the balanced test set as well.

From the segmentation task, we choose the same hypothesis (H2) result on MLNTO data set to re-evaluate with the PR-T curves (see Figure 5.6). To test the H2, we consider the precision curve of PR-T (Figure 5.6: the solid curve of the rightmost plot), which says that the hypothesis is true, i.e., the $\neg\mathbb{C}$-biased training (E2.b) produces less FP since its precision curve (solid blue curve) is the highest one. Its lowest recall curve (dotted blue curve), however, says that E2.b produces the highest FN, which is expected since we did not utilize all the $\mathbb{C}$ patches to keep the training set size the same for all experiments. In summary, the findings are consistent with findings in Chapter 4 and Section 5.7.

5.9 Proposed methods for non-linear interpolation and mean PR curve computation

To compute a mean PR curve, we develop a new method, where we utilize the ROC curve components (TPR, FPR) in a way so that we can fix the stride (i.e., interval) on the x-axis of the PR curve and approximate AUC. We utilize the ROC curves of all trial-models as input. We either interpolate or retrieve points from each input ROC curve for a fixed set of equally spaced TPRs or recall, then convert those points to the PR space. It is possible because ROC components, TPR and FPR, are linearly correlated and straightforward to process, and because the ROC and PR have a one-to-one correspondence [Davis

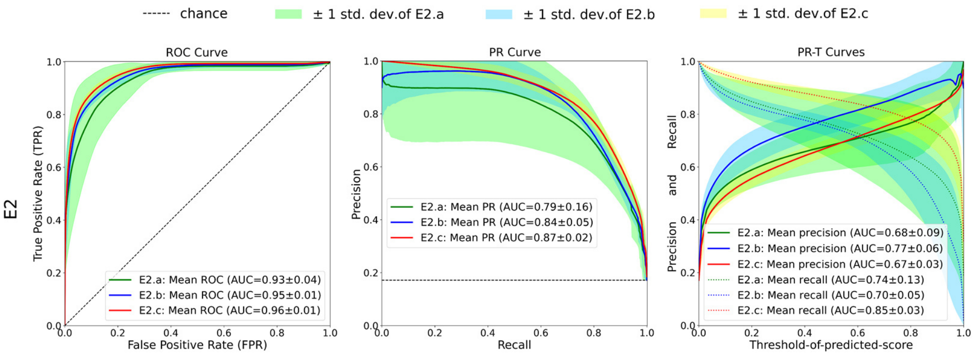


Figure 5.6: **The result is consistent with the findings of H2 in Chapter 4 and Section 5.7 - on MLNTO.** Mean curves with standard deviation for experiment settings E2.a (for balanced distribution in the training set, green), E2.b (for over-represented $\neg\mathbb{C}$, blue) and E2.c (for over-represented $\mathbb{C}$, red). Experiments test H2 in the case of E2 (Table 4.6). Same notations as in Figure 5.5

and Goadrich, 2006; Flach and Kull, 2015]. Like in [Davis and Goadrich, 2006; Boyd et al., 2013; Keilwagen et al., 2014], we interpolate a point in the non-linear PR space by utilizing the linear interpolation in the ROC space. Unlike them, however, we utilize the original components of the ROC (TPR, FPR) instead of discrete [Davis and Goadrich, 2006] or continuous [Keilwagen et al., 2014] TPs or deriving any complex coefficients [Boyd et al., 2013]. Because of using the linear space of the ROC curve and its original components directly, our method is straightforward to understand and computation friendly. Moreover, we also handle the special cases (start, end, intermediate points with ties) and AUC calculation. After the conversion of points from the ROC space to PR space, we apply the straightforward and popular *vertical averaging* [Fawcett, 2006], where, for a predefined list of equally spaced x-axis values, the corresponding y-axis values of all trial-models are averaged.

The overview described above is formalized in Algorithm 3 for the mean PR curve and in Algorithm 4 for the PR-AUC computations. Moreover, we detail our interpolation technique in the text, how we determine the start and endpoints in the usual case, and how we handle the special cases with ties since those are the cases usually handled in a wrong way in the literature.

Interpolating a point on the PR space. As mentioned earlier, we interpolate a point linearly at the ROC space and convert it to the PR space, we consider the following equation to linearly interpolate a point at the ROC

Algorithm 3: Mean PR points computation

```
Input: $TPR_{mean} = \{tpr_m \in \mathbb{R} : 0.0 \leq tpr_m \leq 1.0\}$ is the set of
       equally spaced TPRs (or recalls), where $m = 1, 2, \cdots, 101$
       when stride, $\triangle tpr = 0.01$; $ROCs = \{ROC_i \in \mathbb{R}^2\}$ is the set
       of ROC curves computed for $nT$ trials according to Eq. 2.5,
       where $i = 1, 2, \cdots, nT$; $|ROC_i|$ is the size of the ROC curve
       of $i^{th}$ trial; $pos$ (resp. $neg$) is the number of positive (resp.
       negative) examples.
Output: $PREC_{mean}$, a list of mean precision points
$PRECs \leftarrow [\,][\,]$
for $i$=1 to $nT$ do
    for $m$=1 to $|TPR_{mean}|$ do
        $fpr \leftarrow FPR\_FOR\_TPR(tpr_m, ROC_i)$
        $PRECs[i][m] \leftarrow \frac{tpr_m * pos}{tpr_m * pos + fpr * neg}$
    if $PRECs[i][1] == \infty$ then $PRECs[i][1] \leftarrow PRECs[i][2]$
/* Compute vertical mean with standard deviation */
$PREC_{mean} \leftarrow vertical\_mean(PRECs) \pm vertical\_std(PRECs)$
Function FPR_FOR_TPR($tpr_m$, $ROC$):
    $FPR = \{fpr_j \in \mathbb{R} : 0.0 \leq fpr_j \leq 1.0\}$ and
     $TPR = \{tpr_j \in \mathbb{R} : 0.0 \leq tpr_j \leq 1.0\}$ are the multisets of FPR
     and TPR that constitute the $ROC$, where the frequency of any
     $fpr \in FPR$ (and $tpr \in TPR$), $frq(fpr) \geq 1$ ( and
     $frq(tpr) \geq 1$); $j = 1, 2, \cdots, |ROC|$.
    $j \leftarrow 1$
    while $j < |ROC|$ and $tpr_j < tpr_m$ do
        $j \leftarrow j + 1$  // finding the location of $tpr_m$ in ROC
    if $tpr_m == tpr_j$ then
        if $frq(tpr_j) > 1$ then
            $FPR_j \leftarrow$ the corresponding subset of FPR for $tpr_j$
            case j==1 return $maximum(FPR_j)$
            case $1 < j < |ROC|$ return $median(FPR_j)$
            case other return $minimum(FPR_j)$
        else
            return $fpr_j$
    else
        return $fpr_{j-1} + (fpr_j - fpr_{j-1}) \frac{(tpr_m - tpr_{j-1})}{(tpr_j - tpr_{j-1})}$
```

We consider 101 equally spaced x-axis values to approximate AUC while the stride on the x-axis is 0.01. We also tested stride 0.001 and 1001 values for a large and a small test set, but we did not find differences on AUCs.

Algorithm 4: Mean PR-AUC computation

```
Input: Same input as Algorithm 3.
Output: PR-AUC$_{mean}$, the area under the mean PR curve
Require: Function from Algorithm 5
$AUCs \leftarrow [\,]$
for i=1 to nT do
    $RECALL_i \leftarrow PREC_i \leftarrow [\,]; AUCs[i] \leftarrow 0; j \leftarrow 1$
    for m=1 to $|TPR_{mean}|$ do
        $R, P \leftarrow RECALL\_PREC\_FOR\_AUC(tpr_m, ROC_i)$
        $RECALL_i[j : j + |R|] \leftarrow R$
        $PREC_i[j : j + |R|] \leftarrow P$
        $j \leftarrow j + |R|$
    if $PREC_i[1] == \infty$ then $PREC_i[1] \leftarrow PREC_i[2]$
    for k=1 to j-1 do
        /* Add AUC computed with trapezoidal rule */
        $AUCs[i] \leftarrow AUCs[i] + \frac{PREC_i[k]+PREC_i[k+1]}{2}(RECALL_i[k + 1] - RECALL_i[k])$
PR-AUC$_{mean} \leftarrow mean(AUCs) \pm std(AUCs)$
```

We present the function $RECALL\_PREC\_FOR\_AUC$ utilized in this algorithm at Algorithm 5 since both together do not fit on a single page.

Algorithm 5: Functions for mean PR-AUC computation

```
Function RECALL_PREC_FOR_AUC($tpr_m$, $ROC$):
    $FPR = \{fpr_j \in \mathbb{R} : 0.0 \leq fpr_j \leq 1.0\}$ and
     $TPR = \{tpr_j \in \mathbb{R} : 0.0 \leq tpr_j \leq 1.0\}$ are the multisets of FPR
     and TPR, respectively that constitute the $ROC$, where the
     frequency of any $fpr \in FPR$ (and $tpr \in TPR$), $frq(fpr) \geq 1$
     ( and $frq(tpr) \geq 1$); $j = 1, 2, \cdots, |ROC|$.
    $PREC_m \leftarrow \{\}$; $RECALL_m \leftarrow \{tpr_m\}$; $j \leftarrow 1$
    while $j < |ROC|$ and $tpr_j < tpr_m$ do
        $j \leftarrow j + 1$  // finding the location of $tpr_m$ in ROC
    if $tpr_m == tpr_j$ then
        if $frq(tpr_j) > 1$ then
            $FPR_j \leftarrow$ the corresponding subset of FPR for $tpr_j$
            if $j$==$1$ then
                return $RECALL_m$,
                 $PRECISION(tpr_m, maximum(FPR_j))$
            else if $1 < j < |ROC|$ then
                for $k = 1$ to $|FPR_j|$ do
                    $RECALL_m[k] \leftarrow tpr_m$
                    $PREC_m[k] \leftarrow PRECISION(tpr_m, FPR_j[k])$
                return $RECALL_m$, $PREC_m$
            else
                return $RECALL_m$,
                 $PRECISION(tpr_m, minimum(FPR_j))$
        else
            return $RECALL_m$, $PRECISION(tpr_m, fpr_j)$
    else
        $fpr_m \leftarrow fpr_{j-1} + (fpr_j - fpr_{j-1})\frac{(tpr_m - tpr_{j-1})}{(tpr_j - tpr_{j-1})}$
        return $RECALL_m$, $PRICISION(tpr_m, fpr_m)$
Function PRECISION($tpr$, $fpr$):
    return $\frac{tpr*pos}{tpr*pos+fpr*neg}$
```

This algorithm is a part of Algorithm 4.

space:

$$fpr_m = fpr_{j-1} + (fpr_j - fpr_{j-1})\frac{(tpr_m - tpr_{j-1})}{(tpr_j - tpr_{j-1})} \tag{5.2}$$

where (fpr_{j-1}, tpr_{j-1}) and (fpr_j, tpr_j) are two adjacent points of the ROC curve, (fpr_m, tpr_m) is the point to be interpolated for a predefined TPR (or recall), tpr_m, and $tpr_{j-1} < tpr_m < tpr_j$. We then convert the interpolated point (fpr_m, tpr_m) from ROC space to PR space (Alg. 3, line 5; Alg. 5, line 24) by computing $fp_m (= fpr_m * neg)$ and $tp_m (= tpr_m * pos)$. Then, by placing them at the equation of precision (Eq. 2.2), we compute the corresponding interpolated precision, $prec_m$. Since $recall_m = tpr_m$, $(recall_m, prec_m)$ is the corresponding interpolated point.

Start- and endpoint. When computing precision for the start point, there is however an exception as follows: if tp_0 and fp_0 are both zero, the precision is undefined for that point. Then, as usual practice, the precision at the start point, $prec_0 = prec_1$ (Alg. 3, line 6; Alg. 4, line 9), where $prec_1$ is the precision of the next point of the start point. The precision for the endpoint of the PR curve is $pos/(pos + neg)$, if the x-value, i.e., the recall is not a tie, where pos and neg are the numbers of positive and negative test examples.

To further check if the method of calculating interpolated point was correct, we considered all the examples provided in related work [Davis and Goadrich, 2006; Saito and Rehmsmeier, 2017] as examples of correct interpolation. While in [Saito and Rehmsmeier, 2017], the endpoint did not work well in the case of ties, in our case, we fix it. For mean PR curve computation, Saito and Rehmsmeier [2017] did not describe how they handle ties in other points; we handle and describe these cases in the following paragraphs.

Handling spacial cases. We suggest to compute the mean PR curve and PR-AUC by handling the tied scores of the x-axis more rationally. Usually, for a tied value in the x-axis, the maximum of the corresponding y-values is taken for further computation [Boyd et al., 2013], which is irrational in some cases (as described below) for the PR curve.

We consider three special cases: tied TPR[5] for ROC values at the start, end, and in-between points, that may happen. As illustrated in Figure 5.7, to

[5] Here, we consider TPR/recall from the ROC components since it is the x-axis value for the PR curve, which we need to pre-define for the mean PR curve computation.

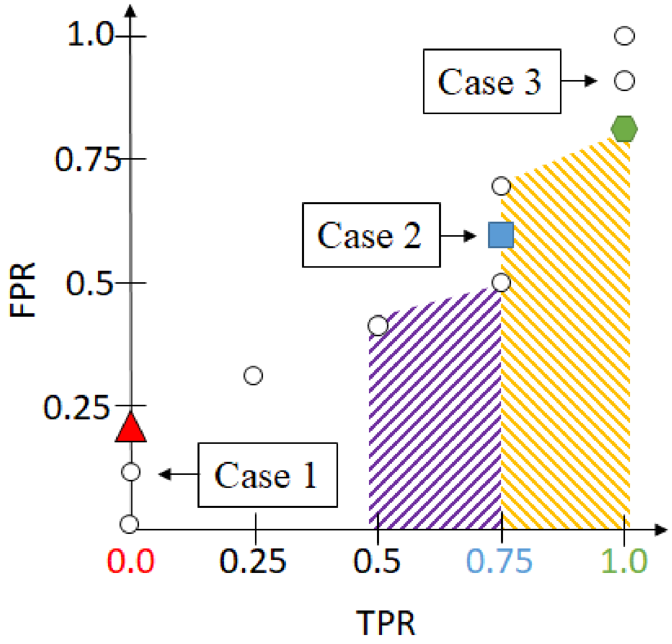


Figure 5.7: **Determining FPRs for tied TPRs for start, end, and in-between points.**

compute the mean PR curve by utilizing ROC space, the red triangle, blue square, and green hexagon are the selected points for the start, in-between, and endpoints. For the PR-AUC, the selected points are the same except for the in-between points with ties. We select the minimum FPR for the left-side interval (0.5 to 0.75 -purple hatched to the left area) and the maximum FPR for the right-side interval (0.75 to 1.0 -yellow hatched to the right area) for the in-between points during PR-AUC computation. All the special points selected from the ROC space are then converted to the points in PR space (Alg. 3, line 5; Alg. 5, 24).

Start point: The TPR of the start point becomes a tie for multiple FPR when some negative examples get the highest predicted score. In Figure 5.7 (Case 1), two negative examples get the highest precedence over all the positive examples while they are ordered in descending for curve computation (the first point at the (0.0, 0.0) is the default start when all examples are predicted as negative or threshold is equal to the ∞). As a cost of giving the highest precedence to negative examples over positives examples, taking the corresponding maximum FPR (the red triangle in the Figure) is rational. The selection of the start point is common to both mean PR curve and PR-AUC computation (See Alg. 3, line 16; Alg. 5, lines 9 and 10).

Intermediate point of the start- and endpoints with ties TPR (See Figure 5.7 Case 2): Mean PR curve requires having one FPR per TPR (or recall) for one trial-model; thus, we consider the median (the blue square in Figure 5.7) of the corresponding FPRs as a selected FPR for the tied TPR as recommended in [Boyd et al., 2013] (Alg. 3, line 17). On the other hand, for PR-AUC of a trial-model, we keep all the real FPRs for a tied TPR (Alg. 5, lines 11 to 15) since using the median of FPRs might under (resp. over) estimate the areas computed for the left side from 0.5 to 0.75 (resp. right from 0.75 to 1.0) intervals of the tied TPR.

End point. A tie in TPR (or recall) happens at the endpoint (TPR=1.0) when all the positive examples get higher predicted scores than at least some of the negative examples, which illustrates the model's efficiency. Thus, as a reward, the earliest corresponding FPR (i.e., the minimum) should be chosen (see the green hexagon in Figure 5.7 Case 3). This selection of the endpoint is common to both mean PR curve and PR-AUC computation (Alg. 3, line 18; Alg. 5, lines 16 and 17).

In short, for PR-AUC, in the case of tied TPR (or Recall), we suggest to use the minimum FPR for the left-side interval and the maximum FPR for the right-side interval. For the mean PR curve, it should be the same, except for tied TPR for intermediate points for which the median is more appropriate.

Once the mean PR points are computed by the Algorithm 3, plotting the predefined TPR_{mean} (i.e., recall) at the x-axis and corresponding computed $PREC_{mean}$ at the y-axis provides the mean PR curve.

5.10 Conclusion

In this chapter, we discuss the limitations of the ROC and PR curves, popularly utilized to evaluate the performance of binary classification. It is acknowledged that the ROC curve is not appropriate for imbalanced data sets, and usually, the PR curve is prescribed in that case. We have illustrated that even the PR curve is not appropriate, specifically with large test sets. Considering limitations of the ROC and PR, we developed a new metric, named PR-T curves. We have shown the specialities of PR-T curves which are as follows: PR-T curves are superior to both PR and ROC curves in distinguishing between the performances of the two models; the positive class was under-represented in both test sets we utilized, however, unlike the ROC and

PR, the performance separation of that class (through recall) is highlighted by PR-T curves; they do not require interpolation to compute mean curves and AUCs; they are able to reflect false prediction type and can be helpful to the practitioners to decide on the type of false prediction that needed to be reduce. Unlike ROC and PR, PR-T is also independent of the test examples order, which is an interesting property to avoid handling ties of predicted scores in the calculation. Theoretically and experimentally, we have shown that PR-T curves is both more accurate and more informative than PR curve while keeping the precision recall trade-off information. Additionally, we also propose end-to-end methods to compute the mean PR curve and PR-AUC since they are absent in the literature but still useful to ensure a fair comparison with any newly proposed metrics.

This chapter uses two image data sets and two CNN architectures; it is thus not exhaustive. The efficiency of the PR-T should be demonstrated on other data sets and other machine learning algorithms, especially on classical ones, e.g., Naive Bayes. Determining optimal points in PR-T is tricky since it has two separate curves; although we discuss it from the perspective of finding the solution to the Pareto front, we did not illustrate it with an example problem. PR-T curves and associated AUCs can however be applicable for any balanced, imbalanced, or non-image data sets in the same way we applied here. Although we did not apply the metric to a multi-class setting, the metric is also applicable in that case in the same way the ROC and PR curves are applied to multi-class problems. Moreover, we present our algorithms to compute the mean PR curve and PR-AUC by applying the vertical averaging, while the algorithms can also be customized for the threshold averaging. In future work, we will work on application-dependent techniques to find the optimal points on the curves, and test its efficiency on other data sets and machine learning algorithms.

We are preparing a manuscript on the contribution of proposing the PR-T curves to submit to a journal and preparing a manuscript on the contribution of proposing algorithms for the mean PR curve and PR-AUC to submit to a conference.

Chapter 6
Conclusions

This thesis has contributed to the class distribution analysis as well as the analysis of balance and imbalance data problem for deep learning models applied to whole slide images (WSI) data for computer-aided cancer detection. It is a domain-specific research. For this domain, there were no studies about the impact of class distribution in the training phase of models, although it is one of the crucial hyper-parameters that regulates the performance of learning-based models. We consider several research questions and propose several hypotheses that emerged from current state-of-the-art on cancer detection in WSIs and class distribution analysis in other non-medical domains. We test and answer them through several experiment settings (e.g. different class distributions, CNN architectures, and some other hyper-parameters) and data sets. Among the data sets, one includes WSIs of multiple types of cancer and other data sets contain breast cancer WSIs only. All the WSIs are created from lymph-node biopsies.

To conduct the analysis, we designed an end-to-end framework for training and testing deep learning models (both CNNs and FCNNs). We developed approaches for extracting and categorizing patches from WSIs as an alternative to the usual random patch selection method for training. Our approach facilitates the coverage of all desire areas of a WSI without repetition of any particular area. The patch categorization is useful to create an expected class distribution.

Through this analysis, we found several interesting findings, e.g., (1) the default choice, balanced distribution is not optimal to train a model for the task of cancer detection in WSIs, (2) most of the errors of such model come from FP due to inter-class similarities and intra-class variations, (3) the natural distribution which is biased to non-ROI provide the best result in terms of FP reduction, (4) the non-cancer-biased is preferable for the distribution in ROI classes (cancer and non-cancer) since this distribution resolves the confusions between cancer and non-cancer classes due to the similarities between the two classes and ensures the coverage of all variations from the most heterogeneous class, non-cancer. This analysis also proved the usefulness of usually neglected mixed patches (containing both cancer and non-cancer at the same time) and non-ROI class. Among the two different tasks, segmentation and classification, the analysis shows that the classification task is less sensitive to the different distributions in the training set than the segmentation one. Moreover, we also test our proposed hypotheses while predicting data from two differently distributed test sets and found consistent results. We, however, did not study the distribution of the validation set. This set can also have an impact since it is used for tuning the models during training. We aim to address this challenge in future work. Nevertheless, the outcomes of the analysis provide useful remarks that are helpful in both creating a new WSI data set and utilizing the existing ones optimally.

In addition, with regard to evaluation for machine learning models, we developed a new patch-based evaluation approach and a multi-threshold-based metric (PR-T curves) to represent precision and recall as functions of continuous threshold in two separate curves. We also developed end-to-end algorithms for mean precision-recall (PR) curve and mean PR-AUC computations.

The patch-based evaluation approach is preferable to the pixel-based one for huge images like WSI; it is also closer to the way pathologists examine images to look for cancer-affected areas. The PR-T curves can distinguish between the performances of the two models for both balanced and imbalanced test sets, they do not require interpolation to compute mean curves and AUCs, they can reflect false prediction type, and independent of the test examples order. The algorithms to compute the mean PR and PR-AUC are helpful to avoid common mistakes (regarding interpolation and handling of special cases) usually done in literature in their computations and ensures fair comparison. Our approach, metric, and algorithms solve the limitations of the existing state-of-the-art ones.

The efficiency testing of the proposed metric, PR-T curves, however, is not exhaustive. It would be interesting to see the efficiency of this metric on some other classical machine learning algorithms (e.g., Naive Bayes) and non-image data sets. Moreover, finding optimal points in terms of both precision and recall would be interesting. In future work, we will work on application-dependent techniques to find the optimal points on the PR-T curves (e.g., finding an optimal solution to the Pareto Front) and test the efficiency of PR-T on more machine learning algorithms and data sets.

Future work. Along with the future work mentioned previously, the probable extensions of this thesis are as follows:

- *A dedicated algorithm to detect micro-metastasis.* Macro-metastases are easy to detect by both pathologists and automatic systems. Micro-metastases, however, are difficult to detect or overlooked without the highest resolution of WSI. A dedicated model could be trained with the training examples containing micros at the highest resolution to solve the problem. The prediction of WSIs in the highest resolution, however, is very time and resource consuming. To cope with this, FCNNs could be an option. They accept any sizes of patches during prediction. Unlike CNNs, they do not require to match with the training patch size. The FCNNs themselves have some limitations. Because of the downsampling, some important pieces of information are lost at the encoder stage of an FCNN; at the decoder stage thus it could not regain the accurate annotation. Toward this problem, we could utilize FCNNs without their decoder part, which is similar to a CNN without dense layers.

- *Algorithms to reduce FP.* Our analysis recommends a less FP-producing class distribution. The next work could be to develop an algorithm that reduces FP in cancer detection from WSIs. FP either cause wrong treatment or demand extra time to recheck it manually by pathologists. According to our observation, the FP area is usually as small as micro-metastasis; thus, rechecking all the predicted micro-metastases with different dedicated models could be a solution. The final decision could be taken either by ensembling the decisions of all models or by a majority voting scheme. Moreover, the ground truth might not be 100% perfect in the case of WSIs. It is normal to have some parts of a gigantic WSI be overlooked or miss-classified, which might misguide the model during training and produce FPs. Furthermore, there are some confusing and challenging locations (e.g. germinal center, blur area) in the WSIs that might cause FP during prediction. Training image fine-tuning and hard or confusing examples finding and fixing the trained model accordingly could be a solution towards this problem.

- *A content and context aware CNN for high recall and precision at the same time.* A system trained with patches from the high-resolution

WSIs leads to high recall. Because of the limitation of resources such as memory, the context information may not be enough and precision might be reduced. On the other hand, a system trained with patches from the down-resolution WSIs gives high precision since enough context information can be given, however, downsampling the resolution causes losing the important details and reduces recall. Towards this problem, a two-branched CNN architecture could be a solution. In the architecture, one branch would accept patches from the highest resolution WSIs for the content information and serve for the recall, while the other branch would accept patches from the lower resolution WSIs for the context information and serve for the precision. The feature sets from the two branches could then be concatenated before the classification layer.

- *Exploring different pre-processing and post-processing of data for CADs.* In our systems, we applied different conventional data augmentation techniques, however, did not conduct any comparative studies. For example, which types of data augmentation techniques are applicable to WSI data, and why? Which one is more effective, and why? Analyzing these research questions would be useful to deal with data scarcity. The same investigation could be beneficial for different data normalization techniques to cope with the viability of WSI data due to the difference in their preparation process in labs. In addition to the pre-processing of data just mentioned, the post-processing of the predicted outputs is also an important factor to get a desirable accuracy of the prediction. For example, filtering, smoothing, thresholding, ensembling are popular post-processing on outputs. A dedicated analytical study could be conducted on this topic as well.

- *Exploring unlabeled WSI data.* Training a deep model requires a large set of data. In biomedical research, however, getting labeled data is costly since it requires domain experts to annotate [Otálora et al., 2021]. To resolve this issue, transfer learning is popularly utilized. In this case, learned features from a base domain are transferred to a target domain. The performance, however, depends on how similar the two domains (base and target) are [Yosinski et al., 2014]. As mentioned earlier, labeled-data scarcity is common in biomedical domains, e.g., cancer detection in WSIs; thus, getting a large data set of the same domain to transfer learned features is not always possible. To this end, exploring the use of unlabeled data from the same domain could be beneficial. Using autoencoder, the representation of unlabeled data, the unsupervised features, could be learned. These unsupervised features could be explored as a base for a small target labeled data in a feature transfer manner. Moreover, to increase the hard negative and positive examples -the reasons of FP and FN- in the training set, the unlabeled data could be a source of harvesting those examples.

List of Figures

List of Tables

List of Algorithms

Bibliography

Abels, E. and Pantanowitz, L. (2017). Current state of the regulatory trajectory for whole slide imaging devices in the usa. *Journal of pathology informatics*, 8. DOI: 10.4103/jpi.jpi_11_17.

Abou Elassad, Z. E., Mousannif, H., and Al Moatassime, H. (2020). A proactive decision support system for predicting traffic crash events: A critical analysis of imbalanced class distribution. *Knowledge-Based Systems*, 205:106314.

Aeffner, F., Adissu, H. A., Boyle, M. C., Cardiff, R. D., Hagendorn, E., Hoenerhoff, M. J., Klopfleisch, R., Newbigging, S., Schaudien, D., Turner, O., et al. (2018). Digital microscopy, image analysis, and virtual slide repository. *ILAR journal*, 59(1):66–79.

Afzal, S., Maqsood, M., Nazir, F., Khan, U., Aadil, F., Awan, K. M., Mehmood, I., and Song, O.-Y. (2019). A data augmentation-based framework to handle class imbalance problem for alzheimer's stage detection. *IEEE Access*, 7:115528–115539.

Agatonovic-Kustrin, S. and Beresford, R. (2000). Basic concepts of artificial neural network (ann) modeling and its application in pharmaceutical research. *Journal of pharmaceutical and biomedical analysis*, 22(5):717–727.

Alzubaidi, L., Al-Shamma, O., Fadhel, M. A., Farhan, L., Zhang, J., and Duan, Y. (2020). Optimizing the performance of breast cancer classification by employing the same domain transfer learning from hybrid deep convolutional neural network model. *Electronics*, 9(3):445.

Antropova, N., Huynh, B. Q., and Giger, M. L. (2017). A deep feature fusion methodology for breast cancer diagnosis demonstrated on three imaging modality datasets. *Medical physics*, 44(10):5162–5171.

Araújo, T., Aresta, G., Castro, E., Rouco, J., Aguiar, P., Eloy, C., Polónia, A., and Campilho, A. (2017). Classification of breast cancer histology images using convolutional neural networks. *PloS one*, 12(6):1–14.

Bacanin, N., Bezdan, T., Tuba, E., Strumberger, I., and Tuba, M. (2020). Optimizing convolutional neural network hyperparameters by enhanced swarm intelligence metaheuristics. *Algorithms*, 13(3):67.

Baloch, B. K., Kumar, S., Haresh, S., Rehman, A., and Syed, T. (2019). Focused anchors loss: Cost-sensitive learning of discriminative features for imbalanced classification. In *Asian Conference on Machine Learning*, pages 822–835.

Barisoni, L., Jennette, J. C., Colvin, R., Sitaraman, S., Bragat, A., Castelli, J., Walker, D., and Boudes, P. (2012). Novel quantitative method to evaluate globotriaosylceramide inclusions in renal peritubular capillaries by virtual microscopy in patients with fabry disease. *Archives of pathology & laboratory medicine*, 136(7):816–824.

Barisoni, L., Lafata, K. J., Hewitt, S. M., Madabhushi, A., and Balis, U. G. (2020). Digital pathology and computational image analysis in nephropathology. *Nature Reviews Nephrology*, 16(11):669–685.

Batista, G. E., Prati, R. C., and Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD explorations newsletter*, 6(1):20–29.

Bejnordi, B. E., Veta, M., van Diest, P. J., and et al. (2017). Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. *JAMA*, 318(22):2199–2210.

Bera, K., Schalper, K. A., Rimm, D. L., Velcheti, V., and Madabhushi, A. (2019). Artificial intelligence in digital pathology—new tools for diagnosis and precision oncology. *Nature reviews Clinical oncology*, 16(11):703–715.

Boyd, K., Costa, V. S., Davis, J., and Page, C. D. (2012). Unachievable region in precision-recall space and its effect on empirical evaluation. In *Proceedings of the International Conference on Machine Learning*, volume 2012, page 349. NIH Public Access.

Boyd, K., Eng, K. H., and Page, C. D. (2013). Area under the precision-recall curve: point estimates and confidence intervals. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 451–466. Springer.

Bradley, A. P. (2014). Half-auc for the evaluation of sensitive or specific classifiers. *Pattern Recognition Letters*, 38:93–98.

Brodersen, K. H., Ong, C. S., Stephan, K. E., and Buhmann, J. M. (2010). The binormal assumption on precision-recall curves. In *2010 20th International Conference on Pattern Recognition*, pages 4263–4266. IEEE.

Buda, M., Maki, A., and Mazurowski, M. A. (2018). A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259.

Bueno, G., Déniz, O., Fernández-Carrobles, M. D. M., Vállez, N., and Salido, J. (2014). An automated system for whole microscopic image acquisition and analysis. *Microscopy research and technique*, 77(9):697–713.

Bylinskii, Z., Judd, T., Oliva, A., Torralba, A., and Durand, F. (2018). What do different evaluation metrics tell us about saliency models? *IEEE transactions on pattern analysis and machine intelligence*, 41(3):740–757.

Carrington, A. M., Fieguth, P. W., Qazi, H., Holzinger, A., Chen, H. H., Mayr, F., and Manuel, D. G. (2020). A new concordant partial auc and partial c statistic for imbalanced data in the evaluation of machine learning algorithms. *BMC medical informatics and decision making*, 20(1):1–12.

Chandrashekar, G. and Sahin, F. (2014). A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28.

Chang, H., Nayak, N., Spellman, P. T., and Parvin, B. (2013a). Characterization of tissue histopathology via predictive sparse decomposition and spatial pyramid matching. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 91–98. Springer.

Chang, H., Zhou, Y., Spellman, P., and Parvin, B. (2013b). Stacked predictive sparse coding for classification of distinct regions in tumor histopathology. In *Proceedings of the IEEE international conference on computer vision*, pages 169–176.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357.

Chawla, N. V., Lazarevic, A., Hall, L. O., and Bowyer, K. W. (2003). Smoteboost: Improving prediction of the minority class in boosting. In *European confer-*

ence on principles of data mining and knowledge discovery, pages 107–119. Springer.

Chen, H., Dou, Q., Wang, X., Qin, J., and Heng, P. (2016). Mitosis detection in breast cancer histology images via deep cascaded networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30.

Chollet, F. et al. (2015). Keras. `https://keras.io`.

Christensen, A. M., Passalacqua, N. V., and Bartelink, E. J. (2019). *Forensic anthropology: current methods and practice.* Academic Press. ISBN-13: 978-0124186712.

Cohen, T. and Welling, M. (2016). Group equivariant convolutional networks. In *International conference on machine learning*, pages 2990–2999.

Coleman, W. B. and Tsongalis, G. J. (2010). *Essential concepts in molecular pathology.* Academic Press. eBook ISBN: 9780080922188.

Cook, J. and Ramadas, V. (2020). When to consult precision-recall curves. *The Stata Journal*, 20(1):131–148.

Cortes, J., Perez-García, J. M., Llombart-Cussac, A., Curigliano, G., El Saghir, N. S., Cardoso, F., Barrios, C. H., Wagle, S., Roman, J., Harbeck, N., et al. (2020). Enhancing global access to cancer medicines. *CA: a cancer journal for clinicians*, 70(2):105–124.

Cracknell, M. J. and Reading, A. M. (2014). Geological mapping using remote sensing data: A comparison of five machine learning algorithms, their response to variations in the spatial distribution of training data and the use of explicit spatial information. *Computers & Geosciences*, 63:22–33.

Crawford, K. (2016). Artificial intelligence's white guy problem. *The New York Times*, 25(06).

Cunningham, P., Cord, M., and Delany, S. J. (2008). Supervised learning. In *Machine learning techniques for multimedia*, pages 21–49. Springer.

Dargan, S., Kumar, M., Ayyagari, M. R., and Kumar, G. (2019). A survey of deep learning and its applications: A new paradigm to machine learning. *Archives of Computational Methods in Engineering*, pages 1–22.

Davis, J. and Goadrich, M. (2006). The relationship between precision-recall and roc curves. In *Proceedings of the 23rd international conference on Machine learning*, pages 233–240.

Dean, J. (2014). *Big data, data mining, and machine learning: value creation for business leaders and practitioners.* John Wiley & Sons. ISBN: 978-1-118-61804-2.

Deisenroth, M. P., Faisal, A. A., and Ong, C. S. (2020). *Mathematics for machine learning.* Cambridge University Press. ISBN-13: 978-1108455145.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. IEEE.

Deniz, E., Şengür, A., Kadiroğlu, Z., Guo, Y., Bajaj, V., and Budak, Ü. (2018). Transfer learning based histopathologic image classification for breast cancer detection. *Health information science and systems*, 6(1):1–7.

Dheeru, D. and Karra Taniskidou, E. (2017). UCI machine learning repository. urlhttp://archive.ics.uci.edu/ml.

DiFranco, M. D., O'Hurley, G., Kay, E. W., Watson, R. W. G., and Cunningham, P. (2011). Ensemble based system for whole-slide prostate cancer probability mapping using color texture features. *Computerized medical imaging and graphics*, 35(7-8):629–645.

Dimitriou, N., Arandjelović, O., and Caie, P. D. (2019). Deep learning for whole slide image analysis: an overview. *Frontiers in medicine*, 6:264.

Djuric, U., Zadeh, G., Aldape, K., and Diamandis, P. (2017). Precision histology: how deep learning is poised to revitalize histomorphology for personalized cancer care. *NPJ precision oncology*, 1(1):1–5.

Evans, A. J., Bauer, T. W., Bui, M. M., Cornish, T. C., Duncan, H., Glassy, E. F., Hipp, J., McGee, R. S., Murphy, D., Myers, C., et al. (2018). Us food and drug administration approval of whole slide imaging for primary diagnosis: a key milestone is reached and new questions are raised. *Archives of pathology & laboratory medicine*, 142(11):1383–1387.

Everingham, M., Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *Int. J. Comput. Vision*, 88(2):303–338.

Fan, K., Wen, S., and Deng, Z. (2019a). Deep learning for detecting breast cancer metastases on wsi. In *Innovation in Medicine and Healthcare Systems, and Multimedia*, pages 137–145. Springer.

Fan, K., Wen, S., and Deng, Z. (2019b). Deep learning for detecting breast cancer metastases on wsi. In *Innovation in Medicine and Healthcare Systems, and Multimedia*, pages 137–145. Springer.

Farahani, N., Parwani, A. V., and Pantanowitz, L. (2015). Whole slide imaging in pathology: advantages, limitations, and emerging perspectives. *Pathology and Laboratory Medicine International*, 7:23–33.

Farmer, M. E. and Jain, A. K. (2005). A wrapper-based approach to image segmentation and classification. *IEEE transactions on image processing*, 14(12):2060–2072.

Fawcett, T. (2006). An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874.

Ferlay, J., Colombet, M., Soerjomataram, I., Parkin, D. M., Piñeros, M., Znaor, A., and Bray, F. (2021). Cancer statistics for the year 2020: An overview. *International Journal of Cancer*.

Flach, P. and Kull, M. (2015). Precision-recall-gain curves: Pr analysis done right. *Advances in neural information processing systems*, 28:838–846.

Frank, A. and Asuncion, A. (2010). UCI machine learning repository. url-http://archive.ics.uci.edu/ml.

Freeman, E. A. and Moisen, G. G. (2008). A comparison of the performance of threshold criteria for binary classification in terms of predicted prevalence and kappa. *Ecological modelling*, 217(1-2):48–58.

Fu, G.-H., Xu, F., Zhang, B.-Y., and Yi, L.-Z. (2017). Stable variable selection of class-imbalanced data with precision-recall criterion. *Chemometrics and Intelligent Laboratory Systems*, 171:241–250.

Garcia-Rojo, M., Sanchez, A., Bueno, G., and De Mena, D. (2016). Standardization of pathology whole slide images according to dicom 145 supplement and storage in pacs. *Diagnostic Pathology*, 1(8).

Geiger, T. R. and Peeper, D. S. (2009). Metastasis mechanisms. *Biochimica et Biophysica Acta (BBA)-Reviews on Cancer*, 1796(2):293–308.

Gelasca, E. D., Byun, J., Obara, B., and Manjunath, B. (2008). Evaluation and benchmark for biological image segmentation. In *2008 15th IEEE International Conference on Image Processing*, pages 1816–1819. IEEE.

Giagkiozis, I. and Fleming, P. J. (2014). Pareto front estimation for decision making. *Evolutionary computation*, 22(4):651–678.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

Graham, S., Epstein, D., and Rajpoot, N. (2020). Dense steerable filter cnns for exploiting rotational symmetry in histology images. *IEEE Transactions on Medical Imaging*, 39(12):4124–4136.

Grau, J., Grosse, I., and Keilwagen, J. (2015). Prroc: computing and visualizing precision-recall and receiver operating characteristic curves in r. *Bioinformatics*, 31(15):2595–2597.

Grixti, J. M. and Ayers, D. (2020). Long noncoding rnas and their link to cancer. *Non-coding RNA research*, 5(2):77–82.

Guet, D. (2021). Choosing the right resolution to train deep learning models on histopathology images. https://keeneye.ai/resources/choosing-the-right-resolution-to-train-deep-learning-models-on-histopathology-images. Accessed: 2021-06-20.

Gupta, S. and Madoff, D. C. (2007). Image-guided percutaneous needle biopsy in cancer diagnosis and staging. *Techniques in vascular and interventional radiology*, 10(2):88–101.

Gurcan, M. N., Boucheron, L. E., Can, A., Madabhushi, A., Rajpoot, N. M., and Yener, B. (2009). Histopathological image analysis: A review. *IEEE reviews in biomedical engineering*, 2:147–171.

Halicek, M., Shahedi, M., Little, J. V., Chen, A. Y., Myers, L. L., Sumer, B. D., and Fei, B. (2019). Head and neck cancer detection in digitized whole-slide histology using convolutional neural networks. *Scientific reports*, 9(1):1–11.

Hamad, R. A., Kimura, M., and Lundström, J. (2020). Efficacy of imbalanced data handling methods on deep learning for smart homes environments. *SN Computer Science*, 1(4):1–10.

Han, W., Johnson, C., Gaed, M., Gómez, J. A., Moussa, M., Chin, J. L., Pautler, S., Bauman, G. S., and Ward, A. D. (2020). Histologic tissue components provide major cues for machine learning-based prostate cancer detection and grading on prostatectomy specimens. *Scientific reports*, 10(1):1–12.

Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., Pal, C., Jodoin, P.-M., and Larochelle, H. (2017). Brain tumor segmentation with deep neural networks. *Medical image analysis*, 35:18–31.

He, K., Zhang, X., Ren, S., and Sun, J. (2016a). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

He, K., Zhang, X., Ren, S., and Sun, J. (2016b). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

He, L., Long, L. R., Antani, S., and Thoma, G. (2010). Computer assisted diagnosis in histopathology. *Sequence and genome analysis: methods and applications*, 15:271–87.

He, L., Long, L. R., Antani, S., and Thoma, G. R. (2012). Histology image analysis for carcinoma detection and grading. *Computer methods and programs in biomedicine*, 107(3):538–556.

Hinz, T., Navarro-Guerrero, N., Magg, S., and Wermter, S. (2018). Speeding up the hyperparameter optimization of deep convolutional neural networks. *International Journal of Computational Intelligence and Applications*, 17(02):1850008.

Hu, Z., Tang, J., Wang, Z., Zhang, K., Zhang, L., and Sun, Q. (2018). Deep learning for image-based cancer detection and diagnosis- a survey. *Pattern Recognition*, 83:134–149.

Ibanez, L., Schroeder, W., Ng, L., and Cates, J. (2003). The itk software guide. ISBN: 1930934106.

Indolia, S., Goswami, A. K., Mishra, S., and Asopa, P. (2018). Conceptual understanding of convolutional neural network-a deep learning approach. *Procedia computer science*, 132:679–688.

Indu, M., Rathy, R., and Binu, M. (2016). "slide less pathology": Fairy tale or reality? *Journal of oral and maxillofacial pathology: JOMFP*, 20(2):284.

Jaccard, N., Rogers, T. W., Morton, E. J., and Griffin, L. D. (2017). Detection of concealed cars in complex cargo x-ray imagery using deep learning. *Journal of X-ray Science and Technology*, 25(3):323–339.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678.

Jiang, Y., Metz, C. E., and Nishikawa, R. M. (1996). A receiver operating characteristic partial area index for highly sensitive diagnostic tests. *Radiology*, 201(3):745–750.

Johnson, J. M. and Khoshgoftaar, T. M. (2019). Survey on deep learning with class imbalance. *Journal of Big Data*, 6(1):27.

Kandel, I. and Castelli, M. (2020). A novel architecture to classify histopathology images using convolutional neural networks. *Applied Sciences*, 10(8):2929.

Kaushal, C. and Singla, A. (2020). Automated segmentation technique with self-driven post-processing for histopathological breast cancer images. *CAAI Transactions on Intelligence Technology*, 5(4):294–300.

Keilwagen, J., Grosse, I., and Grau, J. (2014). Area under precision-recall curves for weighted and unweighted data. *PloS one*, 9(3):1–13.

Kellenberger, B., Marcos, D., and Tuia, D. (2018). Detecting mammals in uav images: Best practices to address a substantially imbalanced dataset with deep learning. *Remote sensing of environment*, 216:139–153.

Khalid, S., Khalil, T., and Nasreen, S. (2014). A survey of feature selection and feature extraction techniques in machine learning. In *2014 science and information conference*, pages 372–378. IEEE.

Khan, A. M., Rajpoot, N., Treanor, D., and Magee, D. (2014). A nonlinear mapping approach to stain normalization in digital histopathology images using image-specific color deconvolution. *IEEE Transactions on Biomedical Engineering*, 61(6):1729–1738.

Khan, K., Khan, R. U., Ahmad, K., Ali, F., and Kwak, K.-S. (2020). Face segmentation: A journey from classical to deep learning paradigm, approaches, trends, and directions. *IEEE Access*, 8:58683–58699.

Khan, S., Islam, N., Jan, Z., Din, I. U., and Rodrigues, J. J. C. (2019). A novel deep learning based framework for the detection and classification of breast cancer using transfer learning. *Pattern Recognition Letters*.

Khan, S. H., Hayat, M., Bennamoun, M., Sohel, F. A., and Togneri, R. (2017). Cost-sensitive learning of deep feature representations from imbalanced data. *IEEE transactions on neural networks and learning systems*, 29(8):3573–3587.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Koller, O., Ney, H., and Bowden, R. (2015). Deep learning of mouth shapes for sign language. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 85–91.

Komura, D. and Ishikawa, S. (2018). Machine learning methods for histopathological image analysis. *Computational and Structural Biotechnology Journal*, 16:34–42.

Krizhevsky, A., Hinton, G., et al. (2009). Learning multiple layers of features from tiny images. `https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf`.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012a). Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C. J. C., Bottou, L., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012b). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105.

Kubat, M., Matwin, S., et al. (1997). Addressing the curse of imbalanced training sets: one-sided selection. In *Icml*, volume 97, pages 179–186. Nashville, USA.

Kumar, M. D., Babaie, M., Zhu, S., Kalra, S., and Tizhoosh, H. R. (2017). A comparative study of cnn, bovw and lbp for classification of histopathological images. In *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–7. IEEE.

Kumar, N., Gupta, R., and Gupta, S. (2020). Whole slide imaging (wsi) in pathology: current perspectives and future directions. *Journal of Digital Imaging*, 33:1034–1040.

Ladha, L. and Deepa, T. (2011). Feature selection methods and algorithms. *International journal on computer science and engineering*, 3(5):1787–1797.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.

LeCun, Y. A., Bottou, L., Orr, G. B., and Müller, K.-R. (2012). Efficient backprop. In *Neural networks: Tricks of the trade*, pages 9–48. Springer.

Lee, D. K., In, J., and Lee, S. (2015). Standard deviation and standard error of the mean. *Korean journal of anesthesiology*, 68(3):220.

Levi, G. and Hassner, T. (2015). Age and gender classification using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 34–42.

Li, C., Li, X., Li, X., Rahaman, M. M., Li, X., Wu, J., Yao, Y., and Grzegorzek, M. (2021a). A state-of-the-art survey of artificial neural networks for whole-slide image analysis: from popular convolutional neural networks to potential visual transformers. *arXiv preprint arXiv:2104.06243*.

Li, S., Yi, M., Dong, B., Tan, X., Luo, S., and Wu, K. (2021b). The role of exosomes in liquid biopsy for cancer diagnosis and prognosis prediction. *International journal of cancer*, 148(11):2640–2651.

Lian, C., Ruan, S., and Denoeux, T. (2015). An evidential classifier based on feature selection and two-step classification strategy. *Pattern Recognition*, 48(7):2318–2327.

Lian, C., Ruan, S., Denœux, T., Li, H., and Vera, P. (2016). Robust cancer treatment outcome prediction dealing with small-sized and imbalanced data from fdg-pet images. In *International Conference on medical image computing and computer-assisted intervention*, pages 61–69. Springer.

Lin, H., Chen, H., Dou, Q., Wang, L., Qin, J., and Heng, P.-A. (2018). Scannet: A fast and dense scanning framework for metastastic breast cancer detection from whole-slide image. In *2018 IEEE Winter Conference on Applications of Computer Vision*, pages 539–546. IEEE.

Litjens, G., Bandi, P., Ehteshami Bejnordi, B., Geessink, O., Balkenhol, M., Bult, P., Halilovic, A., Hermsen, M., van de Loo, R., Vogels, R., et al. (2018). 1399 h&e-stained sentinel lymph node sections of breast cancer patients: the camelyon dataset. *GigaScience*, 7(6):giy065.

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., Van Der Laak, J. A., Van Ginneken, B., and Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical image analysis*, 42:60–88.

Liu, B. (2011). Supervised learning. In *Web data mining*, pages 63–132. Springer.

Liu, Y., Gadepalli, K., Norouzi, M., Dahl, G. E., Kohlberger, T., Boyko, A., Venugopalan, S., Timofeev, A., Nelson, P. Q., Corrado, G. S., et al. (2017). Detecting cancer metastases on gigapixel pathology images. *arXiv preprint arXiv:1703.02442*.

Liu, Y., Kohlberger, T., Norouzi, M., Dahl, G. E., Smith, J. L., Mohtashamian, A., Olson, N., Peng, L. H., Hipp, J. D., and Stumpe, M. C. (2019). Artificial intelligence–based breast cancer nodal metastasis detection: insights into the black box for pathologists. *Archives of pathology & laboratory medicine*, 143(7):859–868.

Lonsdale, J., Thomas, J., Salvatore, M., Phillips, R., Lo, E., Shad, S., Hasz, R., Walters, G., Garcia, F., Young, N., et al. (2013). The genotype-tissue expression (gtex) project. *Nature genetics*, 45(6):580–585.

Lu, C. and Mandal, M. (2015). Automated analysis and diagnosis of skin melanoma on whole slide histopathological images. *Pattern Recognition*, 48(8):2738–2750.

Macskassy, S. and Provost, F. (2004). Confidence bands for roc curves: Methods and an empirical study. Proceedings of the First Workshop on ROC Analysis in AI. August 2004.

Marler, R. T. and Arora, J. S. (2004). Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization*, 26(6):369–395.

Masko, D. and Hensman, P. (2015). The impact of imbalanced training data for convolutional neural networks. `https://www.kth.se/social/files/588617ebf2765401cfcc478c/PHensmanDMasko_dkand15.pdf`. Bachelor thesis.

McAuliffe, M. J., Lalonde, F. M., McGarry, D., Gandler, W., Csaky, K., and Trus, B. L. (2001). Medical image processing, analysis and visualization in clinical research. In *Proceedings 14th IEEE Symposium on Computer-Based Medical Systems. CBMS 2001*, pages 381–386. IEEE.

McClintock, D. (2018). Overview of digital pathology's current state: Technologies, systems, capabilities, limitations, and opportunities. `https://www.executivewarcollege.com/wp-content/uploads/McClintock.THUR_.8.10am.FINAL_.pdf`. Accessed: 2021-06-20.

McClish, D. K. (1989). Analyzing a portion of the roc curve. *Medical Decision Making*, 9(3):190–195.

McQuin, C., Goodman, A., Chernyshev, V., Kamentsky, L., Cimini, B. A., Karhohs, K. W., Doan, M., Ding, L., Rafelski, S. M., Thirstrup, D., et al. (2018). Cellprofiler 3.0: Next-generation image processing for biology. *PLoS biology*, 16(7):1–17.

Mejbri, S. (2019). *Deep learning applied to multivariate medical data*. PhD dissertation, Université Toulouse III-Paul Sabatier.

Mejbri, S., Franchet, C., Reshma, I.-A., Mothe, J., Brousset, P., and Faure, E. (2019). Deep analysis of cnn settings for new cancer whole-slide histological images segmentation: the case of small training sets. In *6th International Conference on Bioimaging*.

Melo, R. C., Raas, M. W., Palazzi, C., Neves, V. H., Malta, K. K., and Silva, T. P. (2020). Whole slide imaging and its applications to histopathological studies of liver disorders. *Frontiers in medicine*, 6:310.

Michie, D., Spiegelhalter, D. J., Taylor, C. C., and Campbell, J., editors (1995). *Machine Learning, Neural and Statistical Classification*. Ellis Horwood, USA. ISBN: 013106360X.

Mitchell, T. M. et al. (1997). Machine learning. ISBN: 0070428077.

Naylor, P., Laé, M., Reyal, F., and Walter, T. (2017). Nuclei segmentation in histopathology images using deep neural networks. In *2017 IEEE 14th international symposium on biomedical imaging (ISBI 2017)*, pages 933–936. IEEE.

Ng, A. (2019). Machine learning yearning: Technical strategy for ai engineers in the era of deep learning. *Retrieved online at https://www. mlyearning. org*. Kindle Edition.

Noh, H., Hong, S., and Han, B. (2015). Learning deconvolution network for semantic segmentation. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1520–1528. IEEE Computer Society.

Oksuz, K., Cam, B. C., Akbas, E., and Kalkan, S. (2018). Localization recall precision (lrp): A new performance metric for object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 504–519.

Otálora, S., Marini, N., Müller, H., and Atzori, M. (2021). Combining weakly and strongly supervised learning improves strong supervision in gleason pattern classification. *BMC Medical Imaging*, 21(1):1–14.

Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1):62–66.

Ozenne, B., Subtil, F., and Maucort-Boulch, D. (2015). The precision–recall curve overcame the optimism of the receiver operating characteristic curve in rare diseases. *Journal of clinical epidemiology*, 68(8):855–859.

O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G. V., Krpalkova, L., Riordan, D., and Walsh, J. (2019). Deep learning vs. traditional computer vision. In *Science and Information Conference*, pages 128–144. Springer.

Padera, T. P., Meijer, E. F., and Munn, L. L. (2016). The lymphatic system in disease processes and cancer progression. *Annual review of biomedical engineering*, 18:125–158.

Padilla, R., Netto, S. L., and da Silva, E. A. (2020). A survey on performance metrics for object-detection algorithms. In *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pages 237–242. IEEE.

Pang, M., Georgoudaki, A., Lambut, L., Johansson, J., Tabor, V., Hagikura, K., Jin, Y., Jansson, M., Alexander, J., Nelson, C. M., et al. (2016). Tgf-β1-induced emt promotes targeted migration of breast cancer cells through the lymphatic system by the activation of ccr7/ccl21-mediated chemotaxis. *Oncogene*, 35(6):748–760.

Pham, H. H. N., Futakuchi, M., Bychkov, A., Furukawa, T., Kuroda, K., and Fukuoka, J. (2019). Detection of lung cancer lymph node metastases from whole-slide histopathologic images using a two-step deep learning approach. *The American journal of pathology*, 189(12):2428–2439.

Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M. P., Shyu, M.-L., Chen, S.-C., and Iyengar, S. (2018). A survey on deep learning: Algorithms, techniques, and applications. *ACM Computing Surveys (CSUR)*, 51(5):1–36.

Prati, R. C., Batista, G. E., and Monard, M. C. (2011). A survey on graphical methods for classification predictive performance evaluation. *IEEE Transactions on Knowledge and Data Engineering*, 23(11):1601–1618.

Prati, R. C., Batista, G. E., and Silva, D. F. (2015). Class imbalance revisited: a new experimental setup to assess the performance of treatment methods. *Knowledge and Information Systems*, 45(1):247–270.

Rawat, W. and Wang, Z. (2017). Deep convolutional neural networks for image classification: A comprehensive review. *Neural Comput.*, 29(9):2352–2449.

Rendón, E., Alejo, R., Castorena, C., Isidro-Ortega, F. J., and Granda-Gutiérrez, E. E. (2020). Data sampling methods to deal with the big data multi-class imbalance problem. *Applied Sciences*, 10(4):1276.

Reshma, I. A., Cussat-Blanc, S., Ionescu, R. T., Luga, H., and Mothe, J. (2021). Natural vs balanced distribution in deep learning on whole slide images for cancer detection. In *Proceedings of the 36th Annual ACM Symposium on Applied Computing*, pages 18–25.

Reshma, I. A., Gaspard, M., Franchet, C., Brousset, P., Faure, E., Mejbri, S., and Mothe, J. (2019). Training set class distribution analysis for deep learning model – application to cancer detection. In *1st International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI)*.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 9351 of *LNCS*, pages 234–241. Springer. (available on arXiv:1505.04597 [cs.CV]).

Roth, H. R., Lu, L., Liu, J., Yao, J., Seff, A., Cherry, K., Kim, L., and Summers, R. M. (2015). Improving computer-aided detection using convolutional neural networks and random view aggregation. *IEEE transactions on medical imaging*, 35(5):1170–1181.

Roux, L., Racoceanu, D., Loménie, N., Kulikova, M., Irshad, H., Klossa, J., Capron, F., Genestie, C., Le Naour, G., and Gurcan, M. N. (2013). Mitosis detection in breast cancer histological images an icpr 2012 contest. *Journal of pathology informatics*, 4.

Saito, T. and Rehmsmeier, M. (2015). The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. *PloS one*, 10(3):1–21.

Saito, T. and Rehmsmeier, M. (2017). Precrec: fast and accurate precision–recall and roc curve calculations in r. *Bioinformatics*, 33(1):145–147.

Salvi, M., Michielli, N., and Molinari, F. (2020). Stain color adaptive normalization (scan) algorithm: Separation and standardization of histological stains in digital pathology. *Computer methods and programs in biomedicine*, 193:105506.

Sarker, I. H. (2021). Machine learning: Algorithms, real-world applications and research directions. *SN Computer Science*, 2(3):1–21.

Schiffman, J. D., Fisher, P. G., and Gibbs, P. (2015). Early detection of cancer: past, present, and future. *American Society of Clinical Oncology Educational Book*, 35(1):57–65.

Schneider, C. A., Rasband, W. S., and Eliceiri, K. W. (2012). Nih image to imagej: 25 years of image analysis. *Nature methods*, 9(7):671–675.

Shalev-Shwartz, S. and Ben-David, S. (2014). *Understanding machine learning: From theory to algorithms*. Cambridge university press. Online ISBN: 9781107298019.

Sharma, H., Zerbe, N., Heim, D., Wienert, S., Behrens, H.-M., Hellwich, O., and Hufnagl, P. (2015). A multi-resolution approach for combining visual information using nuclei segmentation and classification in histopathological images. In *VISAPP (3)*, pages 37–46.

Shen, W., Zhou, M., Yang, F., Yang, C., and Tian, J. (2015). Multi-scale convolutional neural networks for lung nodule classification. In *International confer ence on information processing in medical imaging*, pages 588–599. Springer.

Shirazi, A. Z., Fornaciari, E., Bagherian, N. S., Ebert, L. M., Koszyca, B., and Gomez, G. A. (2020). Deepsurvnet: deep survival convolutional network for brain cancer survival rate classification based on histopathological images. *Medical & biological engineering & computing*, 58(5):1031–1045.

Shorten, C. and Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):60.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556.*

Singla, A. and Patra, S. (2017). A fast automatic optimal threshold selection technique for image segmentation. *Signal, Image and Video Processing*, 11(2):243–250.

Sirinukunwattana, K., Pluim, J. P., Chen, H., Qi, X., Heng, P.-A., Guo, Y. B., Wang, L. Y., Matuszewski, B. J., Bruni, E., Sanchez, U., et al. (2017). Gland segmentation in colon histology images: The glas challenge contest. *Medical image analysis*, 35:489–502.

Slaoui, M. and Fiette, L. (2011). Histopathology procedures: from tissue sampling to histopathological evaluation. In *Drug safety evaluation*, pages 69–82. Springer.

Sofaer, H. R., Hoeting, J. A., and Jarnevich, C. S. (2019). The area under the precision-recall curve as a performance metric for rare binary events. *Methods in Ecology and Evolution*, 10(4):565–577.

Spanhol, F. A., Oliveira, L. S., Petitjean, C., and Heutte, L. (2016). Breast cancer histopathological image classification using convolutional neural networks. In *2016 international joint conference on neural networks (IJCNN)*, pages 2560–2567. IEEE.

Srinidhi, C. L., Ciga, O., and Martel, A. L. (2020). Deep neural network models for computational histopathology: A survey. *Medical Image Analysis*, page 101813.

Sun, W., Tseng, T.-L. B., Zhang, J., and Qian, W. (2017a). Enhancing deep convolutional neural network scheme for breast cancer diagnosis with unlabeled data. *Computerized Medical Imaging and Graphics*, 57:4–9.

Sun, W., Zheng, B., and Qian, W. (2017b). Automatic feature learning using multichannel roi based on deep structured algorithms for computerized lung cancer diagnosis. *Computers in biology and medicine*, 89:530–539.

Sun, Y., Kamel, M. S., Wong, A. K., and Wang, Y. (2007). Cost-sensitive boosting for classification of imbalanced data. *Pattern Recognition*, 40(12):3358–3378.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9.

Tabian, I., Fu, H., and Sharif Khodaei, Z. (2019). A convolutional neural network for impact detection and characterization of complex composite structures. *Sensors*, 19(22):4933.

Tallarida, R. J. and Murray, R. B. (1987). Area under a curve: trapezoidal and simpson's rules. In *Manual of Pharmacologic Calculations*, pages 77–81. Springer.

Tellez, D., Litjens, G., van der Laak, J., and Ciompi, F. (2019). Neural image compression for gigapixel histopathology image analysis. *IEEE transactions on pattern analysis and machine intelligence*.

Thabtah, F., Hammoud, S., Kamalov, F., and Gonsalves, A. (2020). Data imbalance in classification: Experimental evaluation. *Information Sciences*, 513:429–441.

Tharwat, A. (2020). Classification assessment methods. *Applied Computing and Informatics*, 17(1):168–192.

Toğaçar, M., Özkurt, K. B., Ergen, B., and Cömert, Z. (2020). Breastnet: A novel convolutional neural network model through histopathological images for the diagnosis of breast cancer. *Physica A: Statistical Mechanics and its Applications*, 545:123592.

Vahadane, A., Peng, T., Sethi, A., Albarqouni, S., Wang, L., Baust, M., Steiger, K., Schlitter, A. M., Esposito, I., and Navab, N. (2016). Structure-preserving color normalization and sparse stain separation for histological images. *IEEE transactions on medical imaging*, 35(8):1962–1971.

Vapnik, V. (2013). *The nature of statistical learning theory*. Springer science & business media. eBook ISBN: 978-1-4757-3264-1, DOI: 10.1007/978-1-4757-3264-1.

Veeling, B. S., Linmans, J., Winkens, J., Cohen, T., and Welling, M. (2018). Rotation equivariant cnns for digital pathology. In *International Conference on Medical image computing and computer-assisted intervention*, pages 210–218. Springer.

Vu, T. H., Mousavi, H. S., Monga, V., Rao, G., and Rao, U. A. (2015). Histopathological image classification using discriminative feature-oriented dictionary learning. *IEEE transactions on medical imaging*, 35(3):738–751.

Walach, E. and Wolf, L. (2016). Learning to count with cnn boosting. In *European Conference on Computer Vision*, pages 660–676. Springer.

Wang, D., Khosla, A., Gargeya, R., Irshad, H., and Beck, A. H. (2016a). Deep learning for identifying metastatic breast cancer. *CoRR*, abs/1606.05718.

Wang, S., Liu, W., Wu, J., Cao, L., Meng, Q., and Kennedy, P. J. (2016b). Training deep neural networks on imbalanced data sets. In *Neural Networks (IJCNN), 2016 International Joint Conference on*, pages 4368–4374. IEEE.

Webster, J. and Dunstan, R. (2014). Whole-slide imaging and automated image analysis: considerations and opportunities in the practice of pathology. *Veterinary pathology*, 51(1):211–223.

Weinstein, J. N., Collisson, E. A., Mills, G. B., Shaw, K. R. M., Ozenberger, B. A., Ellrott, K., Shmulevich, I., Sander, C., and Stuart, J. M. (2013). The cancer genome atlas pan-cancer analysis project. *Nature genetics*, 45(10):1113–1120.

Weiss, G. M. and Provost, F. (2001). The effect of class distribution on classifier learning: an empirical study. *Rutgers Univ.* DOI : https://doi.org/10.7282/t3-vpfw-sf95.

Weiss, G. M. and Provost, F. (2003). Learning when training data are costly: The effect of class distribution on tree induction. *Journal of Artificial Intelligence Research*, 19:315–354.

Wolpert, D. H. (1996). The lack of a priori distinctions between learning algorithms. *Neural Computation*, 8(7):1341–1390.

Wright, A. M., Smith, D., Dhurandhar, B., Fairley, T., Scheiber-Pacht, M., Chakraborty, S., Gorman, B. K., Mody, D., and Coffey, D. M. (2013). Digital slide imaging in cervicovaginal cytology: a pilot study. *Archives of pathology and laboratory medicine*, 137(5):618–624.

Wu, T., Huang, H., Du, G., and Sun, Y. (2008). A novel partial area index of receiver operating characteristic (roc) curve. In *Medical Imaging 2008: Image Perception, Observer Performance, and Technology Assessment*, volume 6917, page 69170B. International Society for Optics and Photonics.

Wu, Y., Ding, Y., and Feng, J. (2020). Smote-boost-based sparse bayesian model for flood prediction. *EURASIP Journal on Wireless Communications and Networking*, 2020:1–12.

Yang, H., Lu, K., Lyu, X., and Hu, F. (2019). Two-way partial auc and its properties. *Statistical methods in medical research*, 28(1):184–195.

Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. (2014). How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328.

Yu, K.-H., Lee, T.-L. M., Yen, M.-H., Kou, S., Rosen, B., Chiang, J.-H., and Kohane, I. S. (2020). Reproducible machine learning methods for lung cancer detection using computed tomography images: Algorithm development and validation. *Journal of medical Internet research*, 22(8):e16709.

Yu, T. (2012). Rocs: receiver operating characteristic surface for class-skewed high-throughput data. *PloS one*, 7(7):e40598.

Yuan, X., Xie, L., and Abouelenien, M. (2018). A regularized ensemble framework of deep learning for cancer detection from multi-class, imbalanced training data. *Pattern Recognition*, 77:160–172.

Zeiler, M. D. and Fergus, R. (2014). Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer.

Zhang, H., Zhang, H., Pirbhulal, S., Wu, W., and Albuquerque, V. H. C. D. (2020). Active balancing mechanism for imbalanced medical data in deep learning–based classification models. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 16(1s):1–15.

Zheng, Y., Jiang, Z., Zhang, H., Xie, F., Shi, J., and Xue, C. (2019). Adaptive color deconvolution for histological wsi normalization. *Computer methods and programs in biomedicine*, 170:107–120.

Zhou, T., Canu, S., Vera, P., and Ruan, S. (2021). Latent correlation representation learning for brain tumor segmentation with missing mri modalities. *IEEE Transactions on Image Processing*, 30:4263–4274.

Zhou, X., Li, C., Rahaman, M. M., Yao, Y., Ai, S., Sun, C., Wang, Q., Zhang, Y., Li, M., Li, X., et al. (2020). A comprehensive review for breast histopathology image analysis using classical and deep neural networks. *IEEE Access*, 8:90931–90956.

Zhu, Z., Gallant, A. L., Woodcock, C. E., Pengra, B., Olofsson, P., Loveland, T. R., Jin, S., Dahal, D., Yang, L., and Auch, R. F. (2016). Optimizing selection of training and auxiliary data for operational land cover classification for the lcmap initiative. *ISPRS Journal of Photogrammetry and Remote Sensing*, 122:206–221.