12-2022

Enhancing the Performance of the MtCNN for the Classification of Cancer Pathology Reports: From Data Annotation to Model Deployment

Kevin De Angeli
kevindeangeli@utk.edu

To the Graduate Council:

I am submitting herewith a dissertation written by Kevin De Angeli entitled "Enhancing the Performance of the MtCNN for the Classification of Cancer Pathology Reports: From Data Annotation to Model Deployment." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Data Science and Engineering.

Hong-Jun Yoon, Major Professor

We have read this dissertation and recommend its acceptance:

Shang Gao, Russell Zaretzki, Audris Mockus

Accepted for the Council:

Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

To the Graduate Council:
I am submitting herewith a dissertation written by Kevin De Angeli entitled "Enhancing the Performance of the MtCNN for the Classification of Cancer Pathology Reports: From Data Annotation to Model Deployment." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Data Science and Engineering.

Hong-Jun Yoon, Major Professor

We have read this dissertation
and recommend its acceptance:

Shang Gao

Russell Zaretzki

Audris Mockus

Accepted for the Council:

Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Enhancing the Performance of the MtCNN for the Classification of Cancer Pathology Reports: From Data Annotation to Model Deployment

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Kevin De Angeli

December 2022

This work is dedicated to my family who have always supported and encouraged me to pursue my dreams.

Acknowledgements

This work was made possible with the help of my advisor Hong-Jun Yoon. His guidance and freedom allowed me to pursue my curiosity, experiment, and enjoy the entirety of my graduate education. I want to thank Shang Gao for all his patience and advise during my years at ORNL, and my committee members, Russell Zaretzki and Audris Mockus, for their feedback and support during my PhD career.

I would also like to thank Daniela Abraham, James Shollenberger, and Charlotte Chang who have been a continuous source of advice and inspiration.

I also owe great thanks to the Bredesen Center and ORNL for their welcoming attitude and continuous support during my years as a doctorate student.

Abstract

Information contained in electronic health records (EHR) combined with the latest advances in machine learning (ML) have the potential to revolutionize the medical sciences. In particular, information contained in cancer pathology reports is essential to investigate cancer trends across the country. Unfortunately, large parts of information in EHRs are stored in the form of unstructured, free-text which limit their usability and research potential. To overcome this accessibility barrier, cancer registries depend on expert personnel who read, interpret, and extract relevant information. Naturally, as the number of stored pathology reports increases every day, depending on human experts presents scalability challenges. Recently, researchers have attempted to automate the information extraction process from cancer pathology reports using ML techniques commonly found in natural language processing (NLP). However, clinical text is inherently different than other common forms of text, and state-of-the-art NLP approaches often exhibit mediocre performance. In this study, we narrow the literature gap by investigating methods to tackle overfitting and improve the performance of ML models for the classification of cancer pathology reports so that we can reduce the dependency on human expert annotators. We (1) show that using active learning can mitigate extreme class imbalance by increasing the representation of documents belonging to rare cancer types, (2) investigated the feasibility of ensemble learning and a mixture-of-expert variant to boost minority class performance, and (3) demonstrated that ensemble model distillation provides a

strategy for quantifying the uncertainty inherent in labeled data, offering an effective low-resource solution that can be easily deployed by cancer registries.

Table of Contents

List of Tables

List of Figures

List of Attachments

Chapter 2 - Additional file 6: Large dataset: micro/macro F-1 scores table - histology task. (AF_1_6.xlsx)

Chapter 2 - Additional file 7: Large dataset: micro/macro F-1 scores table - subsite task. (AF_1_7.xlsx)

Chapter 2 - Additional file 8: Small dataset: micro/macro F-1 scores table - histology task. (AF_1_8.xlsx)

Chapter 2 - Additional file 9: Small dataset: micro/macro F-1 scores table - subsite task. (AF_1_9.xlsx)

Chapter 3 - Class frequencies. (ClassDistributions.xlsx).

Chapter 3 - Micro and macro scores for each individual registry. (IndividualRegistryResults.xlsx).

Chapter 1

Introduction

1.1 Background

Electronic health records (EHR) were originally implemented as a solution to facilitate health care delivery [61]. The early forms of EHRs were developed between 1971-1992 [27], and by 2019, 95% of hospitals were already using EHRs [52]. According to HealthIT gov (healthit.gov), EHRs include medical history, diagnoses, medications, treatment plans, immunization dates, allergies, radiology images, and laboratory and test results. It is also noted by HealthIT gov that EHRs have numerous benefits including 1) improved patient care, 2) increase patient participation, 3) improved care coordination, 4) improved diagnostics and patient outcomes, and 5) practice efficiencies and cost savings. Today, EHRs constitute immerse databases with rich clinical information.

Information contained in EHRs is useful in numerous areas of medical research. For example, EHRs can be used with evidence based tools that assist with decision making and for clinical trial patient identification [27]. EHRs also facilitates the study of specific diseases and can be used in epidemiological research to estimate incidence, prevalence, and mortality rates [22].

One of the priorities of the Surveillance, Epidemiology, and End Results (SEER) program of the National Cancer Institute (NCI) is to analyze cancer trends and provide information on cancer statistics to reduce the cancer burden in the U.S. population. To accomplish its goals, the SEER program relies in the acquisition and analysis of diagnostic, treatment, and outcomes information through manual information processing by expert staff from vast amounts of unstructured sources (e.g. clinical visit notes, pathology reports, treatment summaries, etc.). This manual processing of information imposes natural limitations on the volume and types of data that can be collected. Furthermore, the reports can vary considerable because they are sourced from different health care providers and laboratories. Additionally, Human error and complex coding rules introduce additional variability within the data [39]. As a result, the current practice to manually extract information from cancer

pathology reports is an expensive, time-consuming task that becomes unscalable as the number of the stored reports continues to increase every day.

In 2016, NCI and several cancer registries of the Surveillance, Epidemiology, and End Results Program (SEER) partnered with the Department of Energy (DOE) to develop AI approaches for information extraction and classification of cancer pathology reports. In the last years, researchers have reported promising results when training deep learning (DL) models to automate the information extraction process [36, 38, 2, 35]. Although these approaches constitute a step forward automating the classification of reports, there exists performance requirements which limit the practicality of some of these solutions.

The Oak Ridge National Lab (ORNL) has been actively developing an API that encapsulates these to assist cancer registries. Research teams at ORNL continue to maintain, expand, and improve the API so that an increasing number of cancer registries can use it. A central part of this research is to contribute to those efforts and provide recommendations to improve the API and the general pipeline.

1.2 Automating the classification of cancer pathology reports

The fist researchers working on the classification of cancer pathology reports compared the performance of both traditional machine learning techniques and neural network methods. One notorious paper is [77] where the authors compared naive bayes, logistic regression, and support vector machine using TF-IDF against three variations of CNN: the first one pre-trained using google news, the second one pre-trained using PubMed, and the third one without pre-training. In that study, it was found that the CNN without pre-training outperformed the other methods.

Today, neural networks represent the state-of-the-art methods for most of the NLP problems. Previous research in deep learning models for pathology report

classification have investigated the use of multitask text convolutional neural networks (MtCNN) [2], hierarchical self-attention networks (HiSAN) [36, 38], and graph convolutional neural networks (GraphCNN) [117]. Out of these networks, the MtCNN and the HiSAN exhibited the best performance and are currently the two models integrated in the API that ORNL develops for NCI.

Most recent research developments in NLP have demonstrated that transformer-based architectures outperforms other neural network architectures [110, 55]. Researchers have investigated the potential of Transformers for classifying cancer pathology reports. One particularly relevant article is [35] where the authors compared the performance of BERT against the MtCNN and HiSAN. In this study, the authors emphasized the limitations of transformers for the classification of clinical text and hypothesized that two specific aspects that may be inhibiting BERT's effectiveness: pretraining and WordPiece tokenization. Their results showed that BERT does not perform significantly better than the other two architectures, and computationally cheaper architectures such as the MtCNN obtains competitive scores.

This dissertation focuses on the MtCNN, with some architectural and training variations that are described in each chapter. The reasons for this model selection are the following: 1) the performance differences between the MtCNN and other neural network models are small, 2) during the initial research stages, the HiSAN was a significantly slower model with little performance improvement, and 3) the CNN and its variant are ubiquitous in the deep learning literature, allowing us to explore a rich number of previous studies and developing work that could benefit thousands of practitioners. In figure 1.1 we present a general overview of the MtCNN model.

1.3 Dataset

For this study, we used six datasets from the Louisiana Tumor Registry (LTR), Kentucky Cancer Registry (KCR), Utah Cancer Registry (UCR), New Jersey State

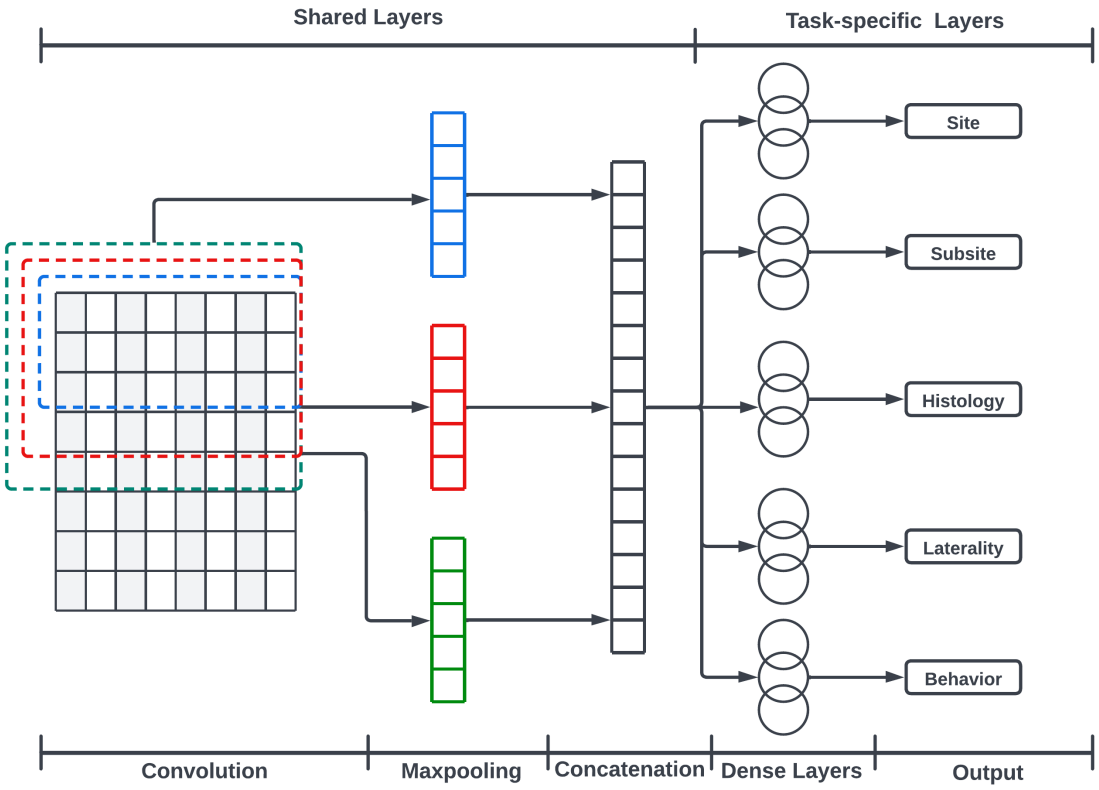


Figure 1.1: High level description of the MtCNN.

Cancer Registry (NJSCR), Seattle Cancer Registry (SCR), and New Mexico Cancer Registry (NMCR). The size of the individual datasets is listed in Table 4.2. In order to satisfy deidentification requirements, we used integers to represent each of the datasets instead of the actual names.

The problem of classifying cancer pathology report consists of five individual tasks. That is, each pathology reports has been labeled with a specific site, subsite, laterality, histology, and behavior. The number of classes in each task is shown in Table 4.1.

1.4 Current Challenges

The dataset exhibits extreme levels of class imbalance. For example, the top two histology classes (*adenocarcinoma, NOS* and *ductal carcinoma*) constitute 41.0% of the dataset. In the subsite task, the top two classes (*upper-outer quadrant of breast* and *prostate gland*) correspond to 17.3% of the data. It is also common to see less than ten instances of the most rare cancer types. Models trained with such datasets will exhibit bias towards the most prevalent majority classes because of their higher prior probabilities, while often ignoring the minority classes. Although the effects of class imbalance in machine learning have been well documented [43, 79, 4], previous researchers have concurred that class imbalance in the context of deep learning is understudied [54].

The data pipeline lacks an efficient label-annotation system. The amounts of unlabeled data are enormous, and labeling every single document is non-feasible. Thus, annotators select documents at random, take around 2 minutes to read them, and assign a label. However, this is not necessarily an efficient way to label documents. That is because some documents are naturally more informative or a better representation of their classes. Thus, one of the current challenges is how to design a system that will select the most informative subset of documents from a set of unlabeled documents.

primarySite : ['C184']

textPathFormalDx : ['gross and microscopic pathologic diagnosis : a . gastric biopsy : benign gastric mucosa with 1 . mucosal alteration suggestive of chemical gastropathy (i . e . bile reflux , nsaids) . 2 . mild chronic inflammation . 3 . no helicobacter - like organisms identified by routine staining . 4 . reparative change , no dysplasia nor malignancy identified . b . esophagus biopsy : benign squamous and glandular mucosa with 1 . active mild chronic inflammation with rare eosinophil and changes suggestive of chronic reflux . 2 . rare intestinal metaplasia is identified by routine staining consistent with barretts esophagus . 3 . reactive change , no dysplasia nor malignancy identified . c . transverse colon polyps : 1 . fragments of tubular adenoma with focal high grade glandular dysplasia / intraepithelial adenocarcinoma (ptis) . 2 . no invasive carcinoma identified . 3 . associated degenerating fecal material . d . polyps at 20 cm : 1 . fragments of tubular adenoma . 2 . fragments of hyperplastic polyp . 3 . no high grade glandular dysplasia nor malignancy identified . 4 . degenerating fecal material . e . sigmoid polyp : 1 . insufficient for diagnosis . 2 . degenerating fecal material . comment : the biopsy from part c (clinically transverse colon polyps) shows tubular adenoma with focal high grade glandular dysplasia confined to the epithelial surface . morphologic features are consistent with intraepithelial adenocarcinoma (ptis) . no invasive carcinoma is seen in the reviewed sections [...]]

textPathNatureOfSpecimens : ['specimen (s) : a gastric bx specimen (s) : b esophagus biopsy specimen (s) : c transverse colon polyp specimen (s) : d polyp * 20 cm specimen (s) : e sigmoid polyp']

Figure 1.2: An example of a de-identified pathology report taken from our dataset.

Table 1.1: Number of classes in each task.

Task	Site	Subsite	Laterality	Histology	Behavior
Classes	70	326	7	639	4

Table 1.2: Size of individual registries.

Registry	R1	R2	R3	R4	R5	R6
e-Path Reports	85,789	577,094	137,135	441,732	360,375	365,152

Data noise is a serious issue when training DL models for classifying cancer pathology reports. Documents often describe multiple specimens and biopsies involving different organs that are analyzed for diagnosis. Manual annotators read the results of each biopsy and assign a specific cancer site label for the entire report. Although this is a standard way to annotate data, this process leads to a large volume of data noise. That is because large portions of pathology reports focus on the analysis of specimens that are associated with a different site and which are not relevant to the context of their ground truth label. Pathology reports also include information such as names and addresses that contributes to additional noise. Training neural networks with noisy data can yield models that learn spurious correlations and shortcuts.

Label noise presents additional challenges. Annotators are charged with the task of selecting a class between a group with hundreds of options. Tasks such as cancer subsite and histology determination involve the identification of specific classes that often share similarities (e.g. "overlapping lesion of other and unspecified parts of mouth" and "mouth not-specified"). Human annotation errors will naturally occur when working with a large number of similar classes and documents where multiple specimens associated with different classes are reviewed. In addition, errors can derive from data processing. Particularly, labels are defined at the cancer/tumor/case (CTC) level. CTC is a data entity which encapsulates all diagnostic, staging, and treatment for a reportable neoplasm. Consequently, pathology reports created during diagnosis are assigned labels based on the CTC even if these documents analyze specimens associated with different labels.

Additional challenges includes performance decay at deployment time. Registries across the country have noted that the performance they observe when using the API is lower than what we observed when training the models in the lab. Current API development lacks experimental testing to quantify the generalization of the model and account for natural variations that may be encountered when deploying the models.

1.5 Performance Requirements

In terms of requirements, cancer registries must have solutions that involve low computational resources. While during lab training we have access to Summit and computers with high computational power, the models that we encapsulate in the API must be computationally efficient so that they can be run in standard, every day computers. This is an additional requirement that needs to be present when making modeling decisions.

Finally, in order to utilize deep learning models to replace humans in the annotation process, cancer registries across the country have specific requirements which impose additional challenges. In particular, cancer registries only tolerate a 3% error based on an estimate of the rate of human error. Thus, high model accuracy and uncertainty quantification (e.g. *knowing what it doesn't know*) are desired model requirements.

1.6 Research Objectives

The proposed dissertation will focus in the application and development of methods to tackle the challenges and requirements described in section 1.4 and 1.5. Our goal is to provide recommendations to improve the API and data pipeline so that we can ship a more efficient product to NCI and ultimately have a deeper understanding of cancer trends across the country. The study will be centered around the following research questions:

1. How can we intelligently select a distinct subset of a larger pool of unlabeled data so that manual annotator efforts are maximized? In other words, can we focus annotation efforts on pathology reports that will maximize the information gain and reduce overfitting? And what are the implications of selecting a particular subset of data in terms of class imbalance and rare classes? Chapter 2 will explore techniques found in the active learning literature. We review a

set of methods that could potentially be used by annotators to maximize the information gain from labeling data.

2. Can we quantify the performance drop that is reported by registries? And how can we design experiments so that they are more reflective of the real-world performance of our models? In chapter 3 we analyze how the distribution of rare cancer classes varies between in-lab training and real world deployment. We consider the use of ensemble learning to tackle overfitting and mixture-of-experts methods to boost rare classes performance.

3. Can we distill the knowledge of an ensemble into a low-resource model that can be used by cancer registries? In addition to the overfitting reduction benefits associated with ensembles, we want to use ensemble learning to derive soft labels that quantify the uncertainty inherent in labeled data. We hope that the derived soft labels will help us boost abstention rates so that the model can be used in a larger subset of data. In chapter 4, we tackle the problem of data and label noise with soft labels.

1.7 Dissertation Outline

The dissertation follows a *three journal publication* structure. That is, the three core chapters (2, 3, 4) are peer-reviewed journal publications that have been published in three distinct journals.

Chapter 2 contains our paper titled "Deep active learning for classifying cancer pathology reports" which was published in BMC Bioinformatics. In this paper, we compared the performance of each active learning strategy using two differently sized datasets and two different classification tasks. Our results showed that active learning techniques have implications in terms of class imbalance because it helps sample more documents from minority classes.

Chapter 3 is a version of our publication "Class imbalance in out-of-distribution datasets: Improving the robustness of the TextCNN for the classification of rare cancer types" published in the Journal of Biomedical Informatics (JBI). In this study, we adopted a new experimental setup that replicates more closely what registries observe when deploying the API in the real world. Here, we investigate different techniques to deal with class imbalance and proposed a variation of mixture-of-expert to improve performance in rare cancer types.

Chapter 4 presents our paper "Using Ensembles and Distillation to Optimize the Deployment of Deep Learning Models for the Classification of Electronic Cancer Pathology Reports" recently published by the Journal of the American Medical Informatics Association (JAMIA Open). In this paper, we aim to reduce overfitting and model overconfidence by distilling the knowledge of an ensemble of deep learning models into a single model. We argue that ensemble predictions provide a useful strategy for quantifying the uncertainty inherent in labeled data and thereby enable the construction of soft labels with estimated probabilities for multiple classes for a given document.

Finally, Chapter 5 presents a summary of the main contributions and findings of this work. Here, we also discuss recommendations for future areas of research.

Chapter 2

Deep Active Learning for Classifying Cancer Pathology Reports

Disclosure Statement

A version of this chapter was originally published in BMC Bioinformatics:

De Angeli, K., Gao, S., Alawad, M. et al. Deep active learning for classifying cancer pathology reports. BMC Bioinformatics 22, 113 (2021). https://doi.org/10.1186/s12859-021-04047-1

Authors' contributions KD: investigation, methodology, software, visualization, writing. SG: conceptualization, formal analysis, investigation, methodology, writing, supervision. MA and HY: conceptualization, formal analysis, methodology, writing, and supervision. NS, XW, ED, JD, AS, LC, and LP: data curation. GT: funding acquisition, supervision.

No revisions to this chapter have been made since the original publication.

2.1 Abstract

Background

Automated text classification has many important applications in the clinical setting; however, obtaining labelled data for training machine learning and deep learning models is often difficult and expensive. Active learning techniques may mitigate this challenge by reducing the amount of labelled data required to effectively train a model. In this study, we analyze the effectiveness of 11 active learning algorithms on classifying subsite and histology from cancer pathology reports using a Convolutional Neural Network (CNN) as the text classification model.

Results

We compare the performance of each active learning strategy using two differently sized datasets and two different classification tasks. Our results show that on all tasks and dataset sizes, all active learning strategies except diversity-sampling strategies

outperformed random sampling, i.e., no active learning. On our large dataset (15K initial labelled samples, adding 15K additional labelled samples each iteration of active learning), there was no clear winner between the different active learning strategies. On our small dataset (1K initial labelled samples, adding 1K additional labelled samples each iteration of active learning), marginal and ratio uncertainty sampling performed better than all other active learning techniques. We found that compared to random sampling, active learning strongly helps performance on rare classes by focusing on underrepresented classes.

Conclusions

Active learning can save annotation cost by helping human annotators efficiently and intelligently select which samples to label. Our results show that a dataset constructed using effective active learning techniques requires less than half the amount of labelled data to achieve the same performance as a dataset constructed using random sampling.

2.2 Background

The latest developments in natural language processing (NLP) have made notable progress in automating classification and information extraction from clinical texts. The current state-of-the-art techniques are generally deep learning (DL) architectures such as Convolutional Neural Networks (CNNs), which have been shown to outperform traditional machine learning techniques and rule-based approaches in clinical text applications [3, 37, 118]. However, a common drawback of DL models is that they tend to require a large amount of training data to achieve high performance. This is a significant problem particularly in clinical applications where obtaining gold-standard labels is difficult and subject to constraints.

A key goal of active learning is to maximize the effectiveness of obtaining additional labelled training data for a given machine learning model. This is achieved by using the model itself to actively select the set of unlabeled data that will be most informative to the model if it were labelled. For example, data associated with common classes that the model is already familiar with may be less informative than data from unseen classes or data on which the model has low confidence. Compared to randomly labelling additional data, active learning enables the model to reach higher performance using fewer additional labelled samples, thereby increasing the efficiency and effectiveness of human annotators [88]. This approach is especially useful for applications such as clinical text classification where annotated data is expensive and time-consuming to obtain.

Cancer pathology reports are an important application where active learning can have real-world impact. Cancer is the second leading cause of death in the United States*. As part of its mission, the National Cancer Institute's (NCI) Surveillance, Epidemiology, and End Results (SEER) program works with population-based cancer registries around the country to collect and publish cancer data including patient demographics, primary tumor site, tumor morphology and stage at diagnosis, first course of treatment, and follow-up for vital status [70]. This information is critical for cancer research and surveillance. While some of this information is stored in structured databases and can be easily extracted, key data elements such as primary tumor site and tumor morphology are generally recorded within unstructured cancer pathology reports that are written during the time of diagnosis. Each year, skilled human Certified Tumor Registrars (CTRs) must manually annotate hundreds of thousands of cancer pathology reports to extract these data elements. Recent research has made major strides toward automating portions of this process [3, 37]; unfortunately, these approaches still do not have the same level of accuracy as human annotators and often have low performance on rare cancer types with few training examples [37, 34]. Active learning can help address these weaknesses by

*https://www.cdc.gov/nchs/fastats/leading-causes-of-death.htm

prioritizing expert annotation of pathology reports that maximize the effectiveness of these automated approaches, thus helping close the gap in performance.

While active learning has been effectively applied to a wide variety of applications, including image classification, speech recognition, and natural language parsing, [44, 33, 80, 101], its application to text classification on cancer pathology reports and other clinical text is relatively limited. Unlike other text classification tasks, classifying cancer pathology reports presents a distinctive set of challenges. These reports are characterized by the lack of a universal structure, variation in linguistic patterns, and the use of specific jargon. Furthermore, documents can be several pages long, only a few keywords or keyphrases in the document may be relevant to a specific classification task, and there may be long-distance linguistic dependencies across different sections of a document [37]. In addition, certain tasks may contain a large number of classes (i.e. 525 histology types), and the number of documents per class tends to be highly unbalanced.

In this paper, we evaluate the effectiveness of different active learning techniques on the task of classifying cancer pathology reports. We test a wide selection of different active learning strategies that have been successfully applied in other applications. Using a CNN-based approach as our learning model, we focus on two classification tasks – identifying cancer subsite and histology. Our contributions are as follows:

- We perform a detailed comparative evaluation of different active learning techniques on two different text classification tasks using cancer pathology reports.

- We examine and compare the performance of active learning when being applied on two scenarios. In the first set of experiments, we use a large dataset with 15K initial labelled samples, adding 15K additional labelled samples each iteration of active learning. For the second set of experiments, we use a small dataset

with 1K initial labelled samples, adding 1K additional labelled samples each iteration.

- Given that clinical text classification tasks generally have high class imbalance, we perform a detailed analysis of how different active learning techniques affect low prevalence classes.

- To our knowledge, this is the first work to perform a critical comparison of different active learning techniques for clinical text classification that utilizes a deep learning model as the classifier.

2.2.1 Related Work

The benefits of classic active learning techniques that incorporate machine learning models have been widely studied for a variety of tasks [88, 71, 89, 109, 105]. In the context of NLP, notable work includes Settles et al. [87] who evaluated 15 query strategies for the task of sequence labeling using conditional random fields on eight different corpora. Even though performance of individual active learning strategies varied across different corpus, the best results were obtained with information density, sequence vote entropy, sequence entropy, and least confidence.

Although active learning and deep learning have each been researched extensively, the current literature at the intersection of both focuses mostly on image classification [94]. Wang et al. [108] proposed a framework that combines uncertainty sampling techniques with pseudo-labeling to reduce human annotation in the context of image classification using CNNs; their technique uses softmax confidence thresholds in the decision process. Zhang et al. [120] explored deep active learning with CNNs for text classification by applying an algorithm called Expected Gradient Length. This work is related to our study; however, it has a specific focus on word embeddings and representation learning.

Few studies have explored AL in the specific context of clinical NLP. These existing works are older studies that focus on using traditional machine learning approaches

rather than state-of-the-art deep learning models. Chen et al. [14] applied seven active learning algorithms using logistic regression for the task of binary clinical text classification. In a similar work, Kholghi et al. [56] applied least confidence and information density to the task of medical concept extraction, and showed that active learning can help classifiers to reach a target performance score using as little as 23% of the total data available. Figueroa et al. [30] applied three active learning algorithms – distance-based, diversity-based, and a combination of both – on five datasets. They used support vector machines as the base classifier for clinical text classification and concluded that diversity algorithms respond better on datasets with high diversity, and distance algorithms perform better on datasets with low uncertainty.

We extend these previous studies by performing a thorough evaluation of 11 different active learning techniques using a modern CNN model as our classifier. We compare performance on two different clinical text classification tasks – extracting subsite and histology from cancer pathology reports. We show that under certain conditions, some active learning strategies clearly beat out others; we expect that these results may provide a useful starting point for other applications of active learning in the context of clinical text classification.

2.3 Methods

2.3.1 Active Learning

In the active learning scenario, we begin with an initial training set $\mathcal{L}_0$ of labeled samples and use it to train a classification model with parameter estimates θ_0. Then, we apply this model on a set of unlabeled data $\mathcal{U}_0$ and use a query strategy $\phi(x_i|\theta_0)$ to assign an informativeness measure to each sample x_i in $\mathcal{U}_0$; this informativeness measure indicates how helpful that sample would be to the classification model if it were to be trained on that sample. We then obtain ground truth labels for a subset of n samples in $\mathcal{U}_0$ with the highest informativeness value $\phi(x_i|\theta_0)$. This subset is moved

from $\mathcal{U}_0$ to $\mathcal{L}_0$ to form the new larger training set $\mathcal{L}_1$ and new smaller unlabeled set $\mathcal{U}_1$.

A new classification model with parameter estimates θ_1 is then trained on $\mathcal{L}_1$ and applied to $\mathcal{U}_1$. Once again, the query strategy $\phi(x_i|\theta_1)$ is used to select the most informative n samples from $\mathcal{U}_1$ to label and add to $\mathcal{L}_1$ to form $\mathcal{U}_2$ and $\mathcal{L}_2$. This process is repeated until the classification model attains the desired performance or until no samples remain in $\mathcal{U}$.

In the following subsections, we describe the various active learning query strategies that we evaluate in this work.

Uncertainty Sampling

One of the most common active learning query strategies is *uncertainty sampling* [62], where ϕ is calculated based on the prediction confidence of the classifier – the assumption is that the lower the confidence of a given sample, the more informative it will be for the model. Within the uncertainty sampling domain, ***least confidence (LC)*** is a simple algorithm which calculates ϕ based on Equation 2.1:

$$\phi^{LC}(x) = 1 - P(y^*|x;\theta) \tag{2.1}$$

where y^* is the predicted class for a given sample, i.e., the class with the highest softmax value, and $P(y^*|x;\theta)$ represents the softmax value associated with that class.

Another uncertainty based query strategy introduced by Schein et al. [85] is ***marginal sampling (MC)***. Whereas least confidence only considers the highest softmax value from each predicted sample, marginal sampling utilizes the difference in confidence between the two most likely classes for each predicted sample (Equation 2.2):

$$\phi^{MS}(x) = 1 - (P(y_1^*|x;\theta) - P(y_2^*|x;\theta)) \tag{2.2}$$

where, y_1^* and y_2^* represents the two classes associated with the highest and second highest softmax values, respectively.

A third uncertainty sampling query strategy, named ***ratio of confidence (RC)*** in this paper, considers the ratio between the top two classes with the highest softmax values (Equation 2.3):

$$\phi^{RC}(x) = \frac{P(y_1^*|x;\theta)}{P(y_2^*|x;\theta)} \tag{2.3}$$

Finally, Shannon Entropy [91], or ***entropy sampling (ES)***, has also been widely used as an uncertainty-based query strategy. Under this approach, ϕ is an entropy-based metric described in Equation 2.4:

$$\phi^{ES}(x) = \sum_{c=1}^{C} P(y_i|x;\theta) * \log P(y_i|x;\theta) \tag{2.4}$$

where $\sum_{c=1}^{C}$ represents the summation over all possible classes, and $P(y_i|x;\theta)$ is the softmax value associated with class y_i. Thus, unlike previous uncertainty sampling techniques, ***ES*** takes into consideration the softmax distribution across all possible classes.

Diversity Sampling

Diversity sampling (DS) algorithms aim to maximize the diversity of the training dataset and calculate ϕ based on a similarity measure between the samples in the training set. Traditionally, diversity sampling algorithms were applied to machine learning approaches that utilized fixed-length input vectors such as TF-IDF, and the similarity measures for ϕ could be applied directly on these input vectors. However, in the context of deep learning models, the inputs are typically matrices of word embeddings that may or may not be zero-padded.

Therefore, to effectively utilize diversity sampling, we first generate a fixed-length document vector representation for each document on which we can then apply the similarity metric. In our study, these document vectors are the outputs from the penultimate layer of our text CNN model (described in detail in the TextCNN subsection). This vector represents the most important features of each document

captured by the convolution filters that are used to make the classification decision for the given task.

We implement two DS algorithms which are named ***Euclidean Cluster-Based Sampling (EC)*** and ***Cosine Cluster-Based Sampling (CC)***. We begin by separating documents in our training set by class and representing the document embeddings for each class as a unique cluster; within a given cluster, we assume that documents closer to the cluster centroid are less informative than documents that are further away from the cluster centroid. Given a sample in the unlabeled set, we calculate ϕ based on how far the document embedding is to the nearest cluster centroid. The difference between the algorithms is the metric used: Euclidean distance or cosine similarity. Algorithm 1 describes the implementation details.

Query-by-committee

The core idea behind Query-by-committee (QBC) based active learning [90, 88] is to train multiple predictive models (the committee) and calculate ϕ based on the disagreement between the models. The committee makes predictions on the holdout set, and samples are ranked based on how much disagreement there exists within the committee. Samples associated with the highest disagreement are selected and added to the training dataset.

In this work, we utilize a committee of 24 CNNs (described in greater detail in the TextCNN subsection) for all QBC-based methods. Each CNN is independently trained on the training data available during each iteration of active learning. Then, we test three different methods to measure the disagreement between the committee members. In our first method, which is named ***Softmax Sum (SS)***, we average the softmax score vectors from all CNNs in the committee for each document in the holdout set. Then, we apply a method similar to *Least Confidence* and rank the documents based on the maximum softmax score across all possible classes.

Algorithm 1: Diversity Sampling

Input:
Set of holdout document embeddings, $\mathcal{U}^*$
Set of document embeddings for the current training dataset, X^*
The training dataset labels, Y
Output: A subset of $\mathcal{U}$

```
Y_centers = []
for y in unique(Y) do
    Find mean of documents in X* that belong to class y, and store it in
     Y_centers;
end
min_dist_lst = []
for u in U* do
    min_d = infinity
    for y_m in Y_centers do
        dist = compute distance from u to y_m
        if min_d ≤ dist then
            min_d=dist
        end
    end
    Store min_d in min_dist_lst
end
Sort documents based on minimum distance, min_dist_lst
Select documents with greatest distance
```

Documents with the lowest max-softmax values are labelled and moved to the training set. Our implementation is described in Equation 2.5:

$$\phi^{SS}(x) = 1 - (\sum_{h}^{\mathcal{H}} P(y^{**}|x;\theta^h)/\mathcal{H}) \tag{2.5}$$

where $\mathcal{H}$ is the number of members in the committee and y^{**} represents the softmax value associated with the class that has the maximum average softmax score across the committee.

For our second method, we apply ***Vote Entropy (VE)***, originally implemented by [5] for the task of part-of-speech tagging. For each document in the holdout set, this method first aggregates the class predictions among the committee members and then utilizes entropy as a measure of disagreement. Our implementation is described in Equation 2.6:

$$\phi^{VE}(x) = -\sum_{c} \frac{V(c,x)}{H} \log \frac{V(c,x)}{H} \tag{2.6}$$

where $V(c,x)$ represents the number of committee members that predict class c for document x.

Lastly, we utilize a modified version of Kullback-Leibler (KL) Divergence originally proposed by Pereira et. al. [73], called ***Kullback-Leibler Divergence to the Mean (KL-D)***. KL Divergence is a common method to measure the difference between two probability distributions. In KL-D, we quantify the disagreement within the committee by calculating the mean KL Divergence between each committee member's softmax vector and the average softmax vector of the whole committee. Our implementation is described in Equation 2.7:

$$\phi^{KL-D}(x) = \frac{1}{H} \sum_{h}^{\mathcal{H}} \sum_{c}^{\mathcal{C}} P(y=c|x;\theta^h) \log(\frac{P(y=c|x;\theta^h)}{\frac{1}{H}\sum_{h}^{\mathcal{H}} P(y=c|x;\theta^h)}) \tag{2.7}$$

Density-Weighted Method

Previous work has suggested that methods such as *uncertainty sampling* and QBC are predisposed to select outliers [82]. To solve this issue, Settles and Craven [87] proposed the method of ***Information Density (ID)***. This method accounts for both uncertainty and diversity by weighting the informativeness scores assigned by any uncertainty sampling technique with a similarity term subject to parameter β. In practice, this method attempts to select samples that the model is uncertain about but that are also similar to other samples in the dataset. Our implementation is shown in Equation 2.8:

$$\phi^{ID}(x) = \phi^{uncertainty}(x) * (\frac{1}{N}\sum_{n=1}^{N} sim(x, x^n))^{\beta} \tag{2.8}$$

where N represents the total number of samples in the holdout set.

In terms of similarity metrics, the authors of the original paper applied exponential Euclidean distance, KL-divergence, and cosine similarity. They reported the last one to be the most effective. For our implementation, we utilize marginal sampling for $\phi^{uncertainty}$, cosine similarity for $sim(x, x^u)$, and $\beta = 1$. We utilize the softmax vectors from the CNN to calculate $\phi^{uncertainty}$ and the document embeddings generated by the penultimate layer of the CNN to calculate $sim(x, x^u)$.

Meta Learning

We propose a novel active learning strategy which consists of training a separate machine learning algorithm to predict which samples will be most informative to the base CNN classifier. The intuition is that if we are able to predict what documents will be misclassified by our model based on the confidence scores generated by the CNN, then we could query those samples and add them to our training dataset.

To achieve this, we first create a new training meta-dataset which consists of the logit vectors ($\vec{x}^{meta}$) obtained by running the trained CNN model on the validation

and test datasets used for active learning. This new dataset also contains a binary label (y^{meta}) that represents whether the CNN correctly classified the corresponding ground truth labels. Then, we use this meta-dataset to train a separate random forest classifier with 100 trees; this random forest learns how likely the CNN will misclassify a given document based on the CNN's relative confidence across the possible classes. For each document in the holdout set, we obtain the CNN's logit vectors and then calculate ϕ based on the random forest's confidence that the CNN will misclassify that document (number of individuals trees out of 100). We refer to this method as ***Meta Learning (ML)***. To the best of our knowledge, we have not seen an implementation of this technique in the current literature.

2.3.2 Application to Cancer Pathology Reports

Cancer pathology reports are a critical resource for cancer surveillance and research. A cancer pathology report is a medical document written by a pathologist that records the cancer diagnosis of cells and tissues examined under a microscope. Cancer pathology reports are generally multi-page documents with highly technical language and contain a variety of detailed information, including but not limited to patient information, specimen details, descriptions of the sample as seen by the naked eye and under a microscope, cancer diagnosis, pathologist and laboratory information, and additional comments. While we are unable to share specific examples of pathology reports from our experimental dataset due to privacy restrictions, example pathology reports can easily be found online.

As part of its mission, the NCI SEER program collects hundreds of thousands of cancer pathology reports each year in partnership with cancer registries around the United States. Human experts must then manually annotate these reports for key data elements related to cancer primary site and morphology. To help ease the burden on human annotators, previous work has applied deep learning techniques such as CNNs to automatically extract these key data elements. In these studies, a dataset

of cancer pathology reports is matched with gold standard human-annotations for key data elements, such as site, subsite, histology, and behavior. A machine learning model is then trained to predict these key data elements – this is generally treated as a single-task or multi-task document classification problem where the input is a cancer pathology report and the output is one or more data elements [3, 37]. However, existing methods still do not achieve high enough accuracy to fully replace human annotators, especially on cancer types with low prevalence and few training examples [37, 34].

Active learning can help address this gap in performance by identifying pathology reports that are especially difficult for automated methods so that human annotators can prioritize annotation of these reports. This has the potential to improve the performance of these automated methods more than if humans experts annotated a random selection of additional cancer pathology reports. To better understand the potential benefits of active learning, we simulate a low data scenario and a high data scenario. Our dataset, tasks, models, evaluation metrics, and experimental setup are described in greater detail in the following sections.

Dataset, Tasks, and Pre-processing

Our dataset consists of cancer pathology reports obtained from the Louisiana Tumor Registry (LTR), Kentucky Cancer Registry (KCR), Utah Cancer Registry (UCR), and New Jersey State Cancer Registry (NJSCR) of the SEER Program[†]. The study was executed in accordance to the institutional review board protocol DOE000152. Each pathology report in our dataset is associated with a unique tumor ID; the same tumor ID may be associated with one or more pathology reports. For each tumor ID, one or more human CTRs manually assigned ground truth labels for key data elements such as cancer site and histology based on all data available for that tumor ID. We note that these ground truth labels are at the tumor level rather than at the report level; as a consequence of this labelling scheme, tumor IDs associated

[†]NJSCR is no longer in the SEER Program, but is included in the current data release.

with multiple pathology reports may have a tumor-level label that does not reflect the content within individual pathology reports. Thus, for this study, we only utilize tumor IDs associated with a single pathology report. The resulting LTR, KCR, UCR, and NJSCR datasets consist of 61123, 46859, 21705, and 70665 pathology reports respectively, yielding a total of 200,352 documents for our experiment.

For this study, we focus on identifying two key data elements that are of importance to NCI – subsite, which is used to identify cancer topology and is indicated by a 3-digit code, and histology, which is used to identify cancer morphology and is indicated by a 4-digit code. Furthermore, these two tasks were chosen because they have a very large number of possible classes and thus are especially challenging for automated machine learning methods. In our dataset, each pathology report is labelled with one of 317 possible subsites and one of 525 possible histologies. For a full list of possible subsite and histology labels and their details, we refer readers to the official SEER program coding and staging manual[‡]. The two figures in Additional File A.1 show the number of occurrences per label of the 50 most frequent classes for histology and subsite. We can see from the figures that there is extreme class imbalance – some classes are represented by less than a few hundred pathology reports, while others are represented by tens of thousands of reports.

Similar to our previous studies, we applied standard text pre-processing techniques such as lowercasing and tokenization to clean our corpus [3, 37]; these steps are described in detail in Additional File A.2. After pre-processing, the average pathology report is 610 word tokens. To reduce the vocabulary size, all words with document frequency less than five were replaced with an "*unknown_word*" token, all decimals were converted to a "*decimal*" word token, and all integers larger than 100 were converted to a "*large_integer*" word token. We limit the maximum length for each cancer pathology report to 1500 word tokens; reports longer than 1500 tokens are truncated and reports shorter than 1500 tokens are zero-padded.

[‡]https://seer.cancer.gov/tools/codingmanuals/index.html

For our active learning setup, we require four data splits: (1) an initial annotated train set to train the starting model used for active learning, (2) an annotated validation set to use for early stopping to prevent overfitting, (3) an annotated test set for performance evaluation, and (4) an unannotated holdout set on which active learning is applied to select new entries to annotate with ground truth labels and add to the train set. Using our cleaned corpus, we create two datasets to simulate active learning situations with different amounts of labelled data. For our first dataset, we begin with a labelled training set of 15K samples; this dataset represents an active learning scenario with a fairly large amount of labelled data. For our second dataset, we begin with only 1K labelled reports for the initial training set; this dataset represents an active learning scenario with a small amount of labelled data. We use the exact same validation and test sets for both scenarios, and the holdout set is comprised of all remaining samples. Table 2.1 shows the size of each dataset partition that was used to simulate each active learning scenario.

TextCNN

For our classification model, we use a word-level text CNN because it is widely used for clinical NLP and text classification tasks [119, 112]. The CNN architecture is implemented based on the same architecture we have used in previous studies [3, 37]. The model hyperparameters are listed in Table 2.2. Our CNN uses randomly initialized word embeddings, which perform as well as or better than other pre-trained word embeddings when applied to our particular dataset, model, and experimental setup [76]. When training our CNN, we checkpoint after each epoch and stop training if validation accuracy does not improve for five consecutive epochs; we test using the checkpoint from the epoch with the best validation accuracy.

Evaluation Metrics: F1 Score

To evaluate the performance of each active learning method, we calculate the micro F1 score (Equation 3.5) of the CNN after each iteration of active learning. We note that micro F1 score is equivalent to classification accuracy in classification tasks such as ours in which each sample is assigned to exactly one class. Micro F1 score is an important metric because it reflects the overall percentage of reports classified correctly.

Because micro F1 score measures overall accuracy regardless of class, in classification tasks with extreme class imbalance, the micro F1 score mostly reflects the performance on majority classes. Therefore, we also report the macro F1 score (Equation 3.6) of the CNN after each iteration. Macro F1 score equally weighs the F1 score on each unique class regardless of class size. As a result, macro F1 score is more heavily influenced by performance on minority classes. In our application, it is important that automated classifiers correctly identify cancer subsites and histologies even if they are rare; as such, macro F1 score is an useful indicator of the effectiveness of each active learning method on the rare classes.

$$\text{Precision} = \frac{\textit{True Positive}}{\textit{True Positives} + \textit{False Positives}} \tag{2.9}$$

$$\text{Recall} = \frac{\textit{True Positives}}{\textit{True Positives} + \textit{False Negatives}} \tag{2.10}$$

$$\text{Micro F1} = 2 * \frac{\textit{Precision * Recall}}{\textit{Precision} + \textit{Recall}} \tag{2.11}$$

$$\text{Macro F1} = \frac{1}{|C|} \sum_{C_i}^{C} F1(C_i) \tag{2.12}$$

In Equation 3.6, C_i represents the subset of training samples belonging to class i, and $|C|$ is the total number of possible classes.

Evaluation Metrics: Class Imbalance

Extreme class imbalance is a common and problematic issue in clinical classification tasks, and it is often difficult to effectively train a classifier on a class with very few labelled samples. To better understand how active learning can reduce extreme class imbalance, we analyze the class imbalance of the training dataset after each iteration of active learning. We use a modified version of Shannon Entropy as a balance metric, described in Equation 2.13:

$$Balance = \frac{-\sum_{i=1}^{C} \frac{c_i}{n} * \log(\frac{c_i}{n})}{\log C} \tag{2.13}$$

where n represents the number of documents in the training set, C is the total number of possible classes (525 for histology and 317 for subsite), and c_i is the number of training samples that belong to class i. This equation outputs a value of 0 when the training dataset is perfectly imbalanced (i.e., contains only samples from a single class) and a value of 1 when the training dataset is perfectly balanced (i.e., all classes have the same number of training samples).

Evaluation Metrics: Proportion of Unique Classes

We also track how active learning affects the number of unique classes seen by the model after each iteration. Intuitively, a model that has never seen samples from a rare class will never accurately predict that class; therefore we want to expose the model to as many unique classes as possible. To measure this, after each iteration of active learning, we simply calculate the ratio between the number of classes present in the training dataset and the total number of classes within the entire dataset.

Experimental Setup

To compare the performance of the different active learning strategies, we benchmark using two different datasets simulating low and high resource settings (see Table 2.1). For each of these two datasets, we test the effectiveness of the CNN models on two

different tasks – subsite and histology. These are treated as two independent single-task classification problems – we train one CNN to predict subsite and a separate CNN to predict histology. This results in a total of four different active learning experiments.

For any given active learning strategy, we first train a CNN on the initial training set. Then, we use that active learning strategy with the trained CNN to select a subset of reports from the holdout set with the highest ϕ; we select 1K samples per active learning iteration for the small dataset and 15K samples per active learning iteration for the large dataset. These selected reports are removed from the holdout set and added to the training set along with their ground truth labels, and a new CNN is trained from scratch on the new training set. For the large dataset, this process is repeated until there are no remaining documents in the holdout set. For the small dataset, we repeat this process nine times total until the training set consists of 10K samples. Figure 2.1 shows a general flowchart of the computational pipeline followed during each experiment.

At each iteration of active learning, we report the micro and macro F1 scores on the test set, the class imbalance of the training set, and the proportion of unique classes seen by the model. For micro F1 score, we calculate 95% confidence intervals using a bootstrapping procedure [23] described in detail in Additional File A.3. We note that we do not use bootstrapping on the macro F1 score because it tends to undersample the minority classes, which are critical for accurately representing macro F1 score. Also, we note that for all active learning strategies and at every iteration of active learning, the test and validation sets are fixed to maintain consistency.

After each iteration of active learning, we train a new CNN from scratch (i.e., cold start) rather than continue training the weights from the previous CNN (i.e. warm start); this is because we found that warm start results in lower accuracy, especially in the later iterations of active learning. We provide a comparison plot between active learning with cold start, active learning with warm start, and no active learning from one of our experiments in Additional File A.4

Table 2.1: Data split and number of classes for the two tasks analyzed.

Dataset	Initial Training	Validation	Testing	Holdout	Classes	
Large	15,000	18,032	20,036	147,284	525 (Histology)	317 (Subsite)
Small	1,000	18,032	20,036	161,284	525 (Histology)	317 (Subsite)

Table 2.2: CNN Hyperparameters.

Input Length	Word Embed Dim	Num Filters	Conv Window Sizes	Dropout	Optimizer	Learning Rate	Batch Size
1500	300	100	3,4, and 5	0.5	Adam	0.0001	128

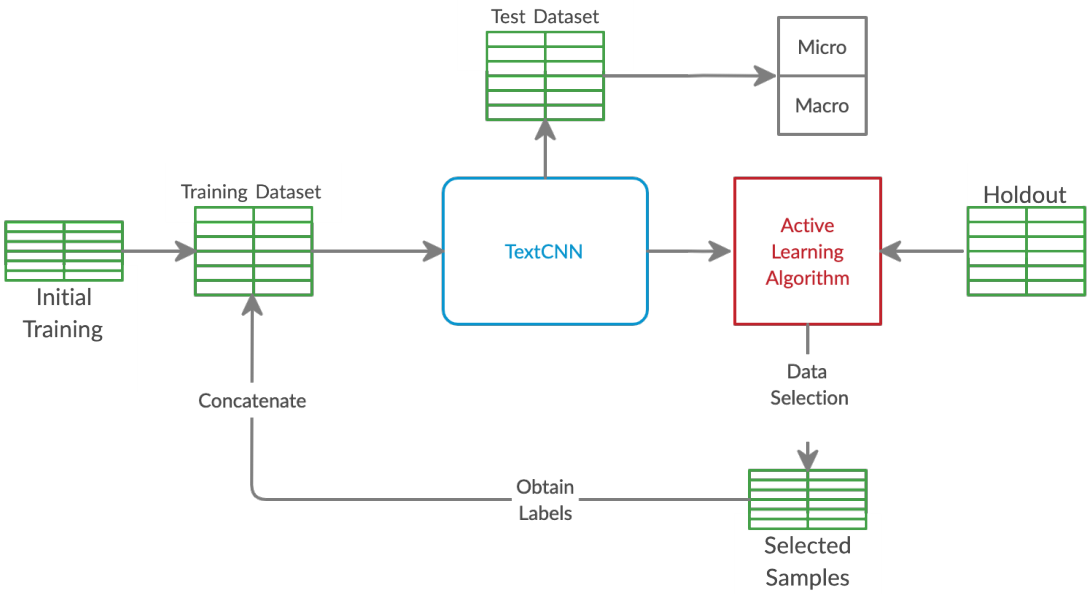


Figure 2.1: Flowchart of the computational pipeline used during the active learning experiments.

All experiments are run using Tensorflow 1.15 and a single NVIDIA V100 GPU (we note that the QBC models are trained in parallel, with each committee member trained on a single NVIDIA V100 GPU). For reference, we report the training and inference time for one of our experiments using the large dataset in Additional File A.5. We note that the text CNN model that we use is a relatively simple DL model with approximately 22M learnable parameters, 20M of which are associated with the learnable word embeddings. Using our experimental settings (see Table 2.2), the model will train even on lower-end GPUs with less than 4GB memory.

2.4 Results

2.4.1 Histology - Large Dataset

The table in Additional File A.6 shows the micro and macro F1 scores for the histology task using our large dataset with 15K initial training samples; we also plot the results with shaded 95% confidence intervals (Figures 2.2 and 2.3). After accounting for the confidence intervals, all active learning strategies implemented in this paper except for the diversity-based methods performed significantly better than the baseline of no active learning, i.e., random sampling. We note that this difference in performance between active learning and random sampling decreases in the later iterations; this is expected because by the last iteration of active learning, all methods (including random sampling) are training on the exact same data, i.e., all data available in the holdout set.

Diversity sampling, which makes decisions based on the similarity between document embeddings created by the CNN, did not produce significant improvements over random sampling. These results suggest that the document embeddings generated by the CNN, which are optimized for classification, may not adequately capture the information necessary to distinguish informative documents. Furthermore, euclidean

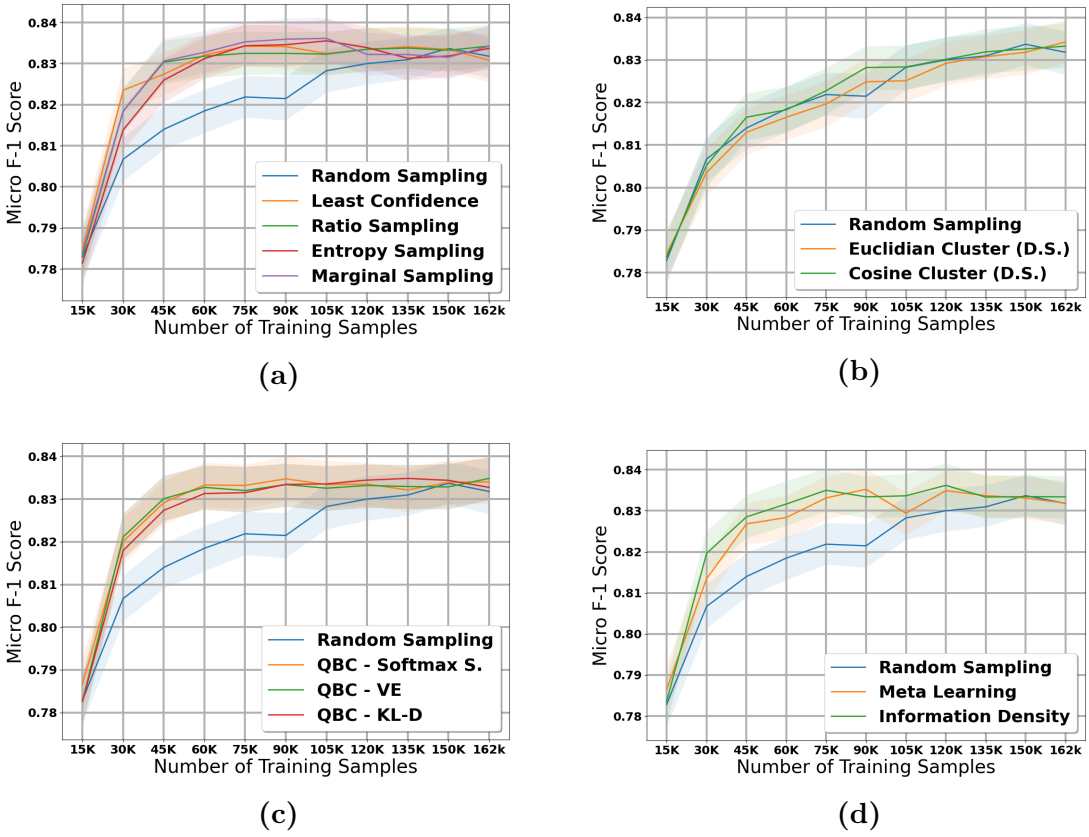


Figure 2.2: Micro score results for the 11 active learning algorithms applied during the large dataset experiment on histology. Blue line represents random sampling.

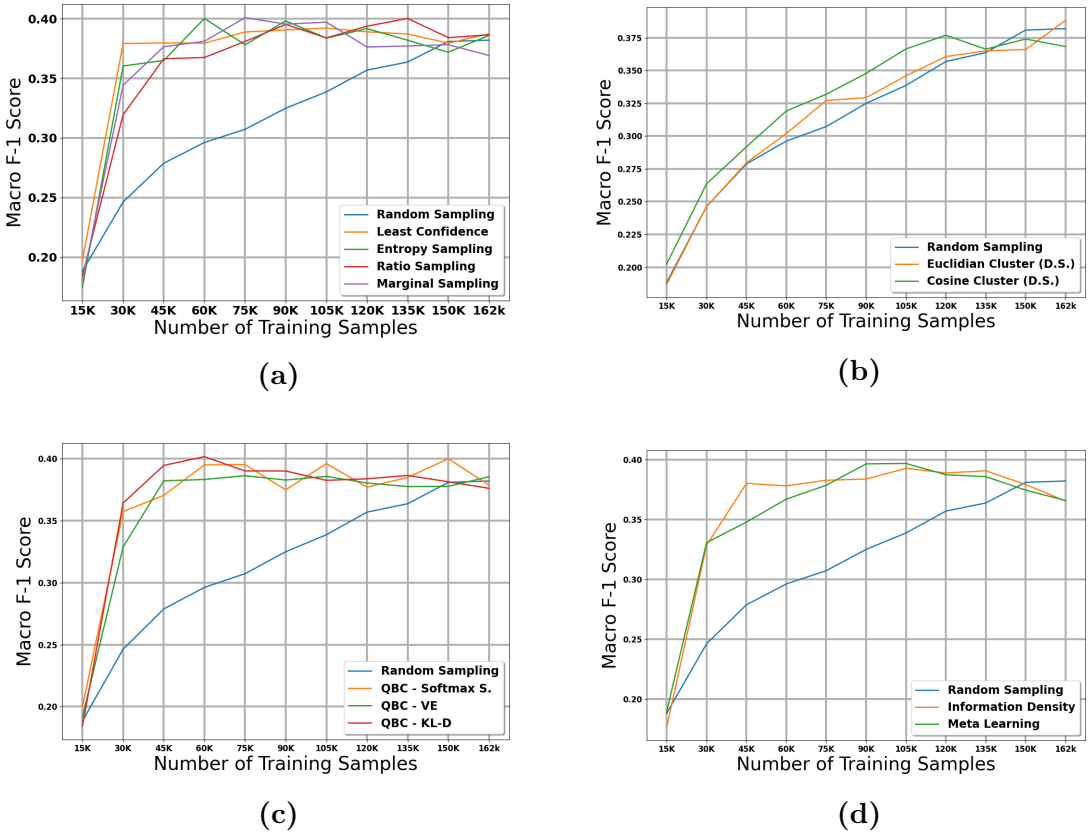


Figure 2.3: Macro score results for the 11 active learning algorithms applied during the large dataset experiment on histology. Blue line represents random sampling.

and cosine distance from the nearest class centroid may not be the best indicator of how informative a document is.

After excluding the diversity sampling strategies, no single active learning strategy stands out as a clear winner in terms of micro F1 scores after accounting for confidence intervals. In terms of macro F1 scores, QBC with KL divergence obtained the strongest macro scores in early iterations, but most of the active learning methods managed to reach the maximum macro F1 score of ∼0.40.

One interesting observation is that the highest macro scores tend to appear towards the middle iterations of the experiment and then tend to go down during the last few iterations. This is a pattern that is not observed for micro scores, where the maximum score is attained near the middle iterations and remains high throughout the rest of the iterations. We expect that this is because in the later iterations, most of the data remaining in the holdout set are less informative samples from majority classes; adding these samples does not negatively affect overall accuracy but may increase class imbalance and thus reduce performance on minority classes.

2.4.2 Subsite - Large Dataset

The table in Additional File A.7 shows the micro and macro F1 scores for the subsite task using our large dataset with 15K initial training samples; Figures 2.4 and 2.5 provide plots of our results with 95% confidence intervals. The results on our subsite task were very similar to our results from our histology task. After accounting for confidence intervals, the diversty-based methods failed to perform significantly better than random sampling, and Euclidian distance actually performed worse than random sampling. All other methods attained maximum micro and macro F1 scores much earlier than random sampling.

Once again, after excluding the diversity-based strategies, it is difficult to distinguish a clear winner in terms of micro F1 score after accounting for confidence intervals. The weakest method appears to be Meta Learning, which performed

similarly to random sampling in the first three iterations of the experiment. In terms of macro F1 score, the QBC methods once again attained strong macro scores at early iterations of the experiment; however, most of the other methods manage to reach the maximum macro F1 score of ∼0.35 by the middle iterations of the experiment.

We note that in the subsite experiment, we do not observe the same drop in macro F1 scores toward the later iterations of active learning that we observed in the histology experiment. We expect that this is because there are fewer unique classes in the subsite task compared to the histology task, and thus the effect of class imbalance is less severe.

2.4.3 Histology - Small Dataset

The table in Additional File A.8 shows the micro and macro F1 scores for the histology task using our small dataset with 1K initial training samples; we also plot the results with shaded 95% confidence intervals in Figures 2.6 and 2.7. Compared to the experiments on the large dataset, we notice several important similarities and differences. First, the diversity sampling strategies not only failed to outperform random sampling, but performed significantly worse in these experiments. Secondly, most active learning strategies that had solid performance in the large dataset no longer show strong performance in this small dataset – the QBC strategies, least confidence, entropy sampling, information density, and meta learning all failed to perform better than random sampling in most of the early and middle iterations.

After taking into account confidence intervals, the marginal sampling and ratio sampling techniques were the only two active learning techniques that significantly outperformed the random sampling baseline in terms of micro F1 score. Interestingly, these two methods do not attain the best performance in macro F1 score, class balance, and proportion of unique classes seen by the model.

When examining macro F1 score, all active learning strategies except for the diversity sampling strategies and information density perform much stronger than the

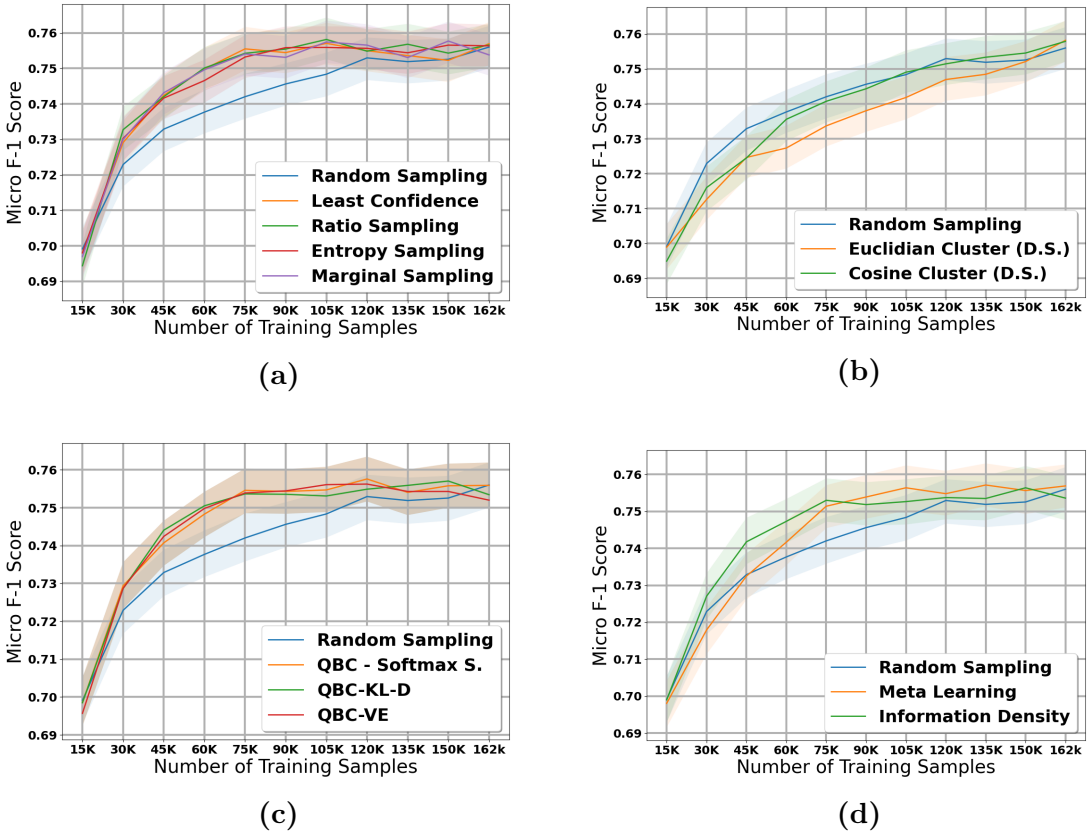


Figure 2.4: Micro score results for the 11 active learning algorithms applied during the large dataset experiment on subsite. Blue line represents random sampling.

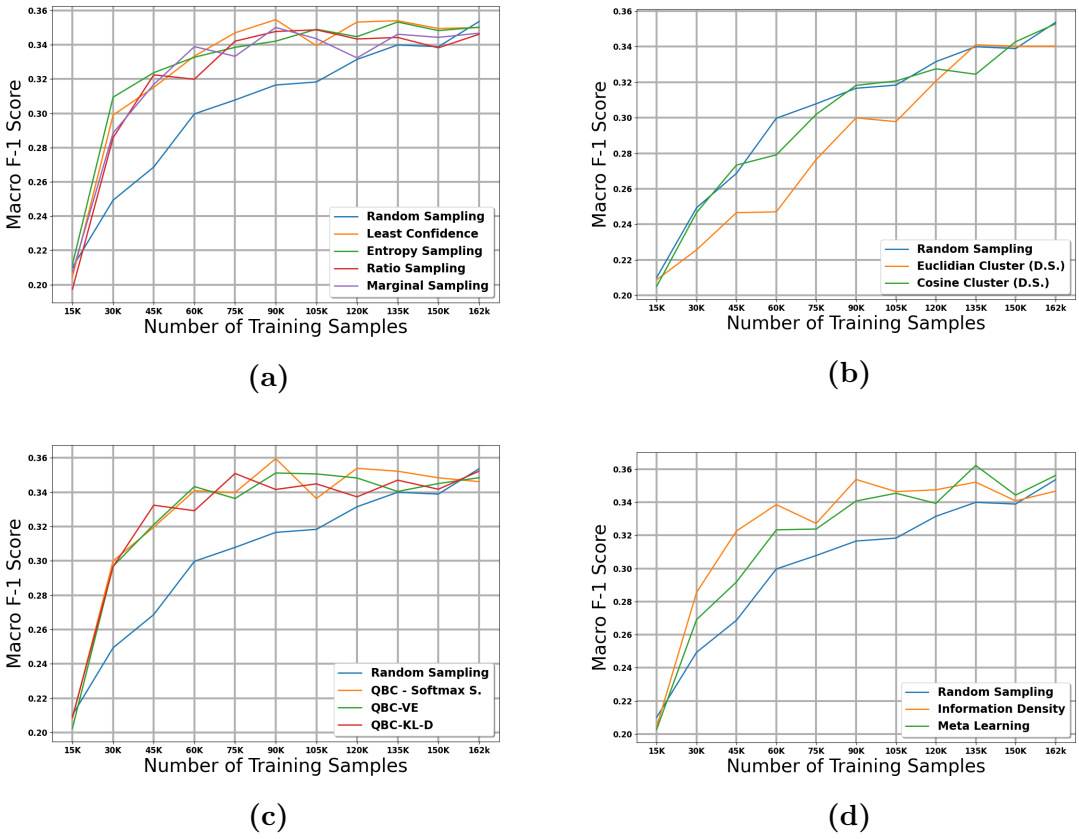


Figure 2.5: Macro score results for the 11 active learning algorithms applied during the large dataset experiment on subsite. Blue line represents random sampling.

random sampling baseline. These results suggest that in general, the active learning strategies in this paper focus on minority classes at the expense of the majority classes; in some cases this may increase macro F1 score but reduce the potential gains in micro F1 score. We explore this phenomena in greater detail in our Discussion.

2.4.4 Subsite - Small Dataset

The table in Additional File A.9 shows the micro and macro F1 scores for the subsite task using our small dataset with 10K samples; we also plot the results with shaded 95% confidence intervals in Figures 2.8 and 2.9. The findings in this experiment are similar to our findings in the histology experiment with the small dataset.

In terms of micro F1 score, marginal sampling and ratio sampling are the only two active learning strategies that significantly outperform the baseline of random sampling, and meta learning and the diversity sampling strategies significantly underperformed random sampling. However, in terms of macro F1 score, all methods except for the diversity sampling strategies outperform the random sampling baseline. Combined with the histology results on our small dataset, these results suggest that the best active learning strategy depends on the size of the initial training set and the amount of labelled data added per iteration of active learning.

2.5 Discussion

Our experiments show that the best choice of active learning strategy is dependent on the size of the initial labelled training set and the amount of labelled data added per iteration. In our large dataset experiments, there was no clear winner in terms of micro F1 score (overall accuracy) – least confidence, ratio sampling, entropy sampling, marginal sampling, QBC softmax, QBC VE, QBC KL-D, and information density all had similar performance. If macro F1 score and performance on minority classes is of high importance, QBC KL-D had the strongest performance on early

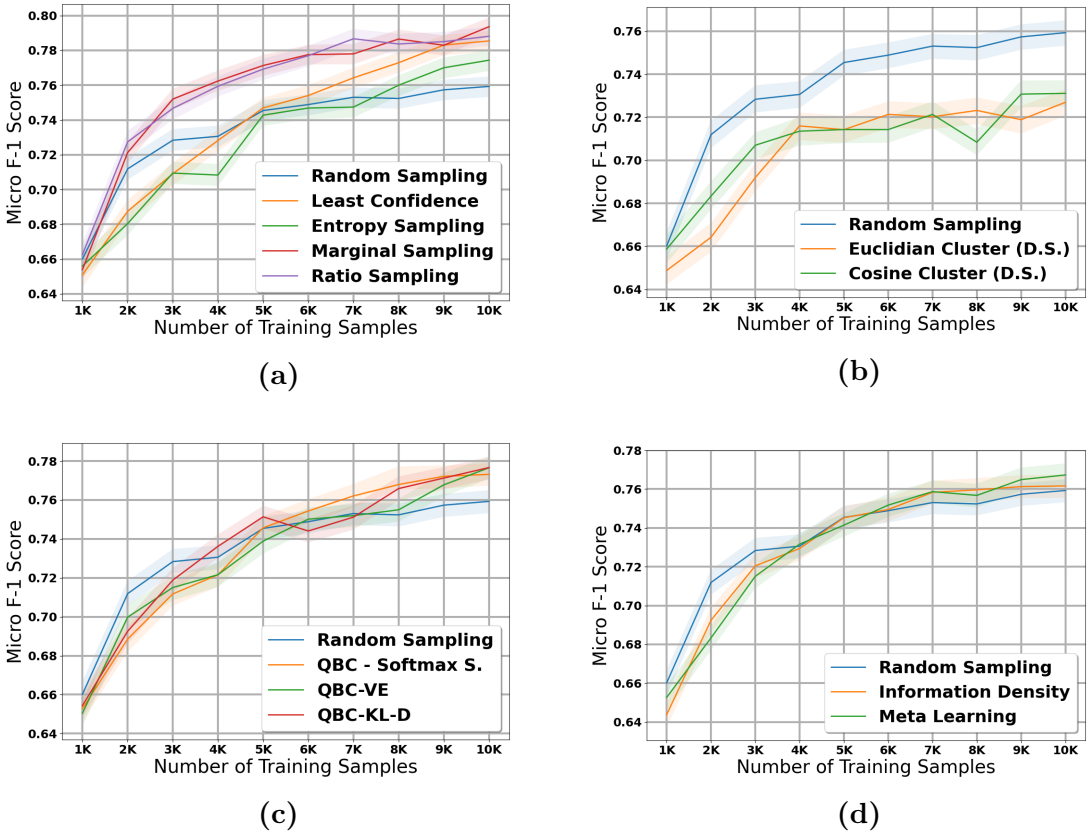


Figure 2.6: Micro score results for the 11 active learning algorithms applied during the small dataset experiment on histology. Blue line represents random sampling.

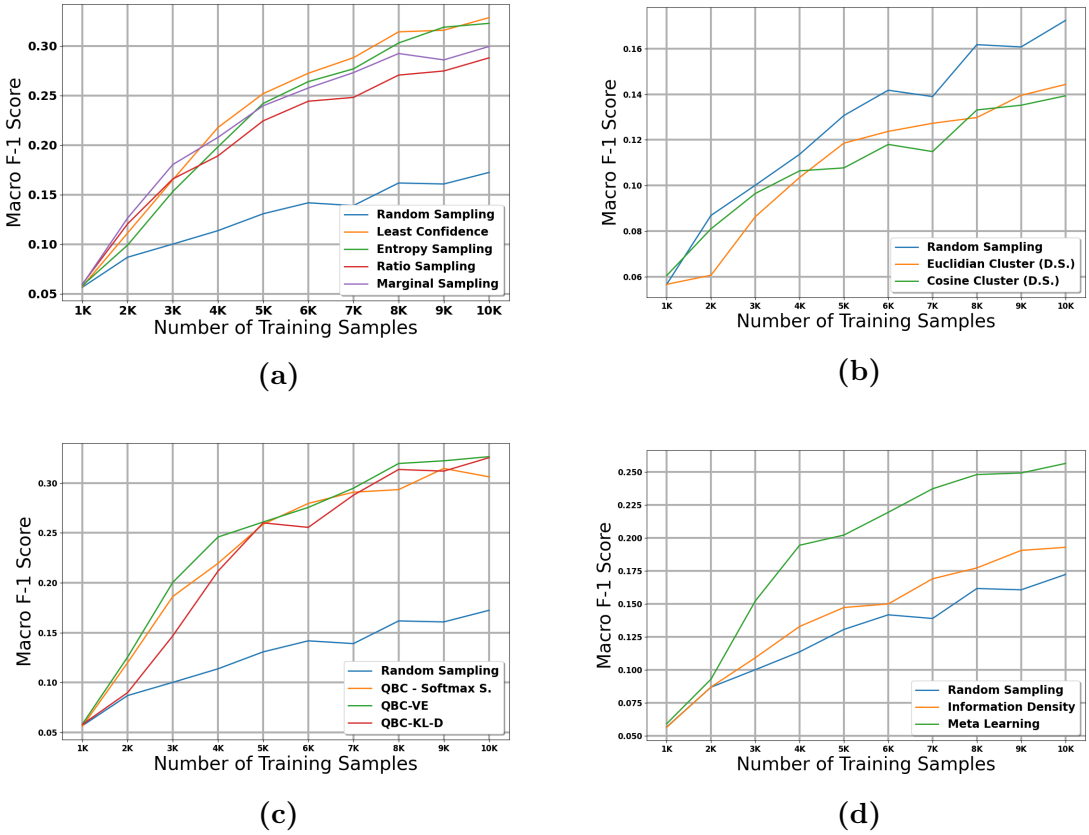


Figure 2.7: Macro score results for the 11 active learning algorithms applied during the small dataset experiment on histology. Blue line represents random sampling.

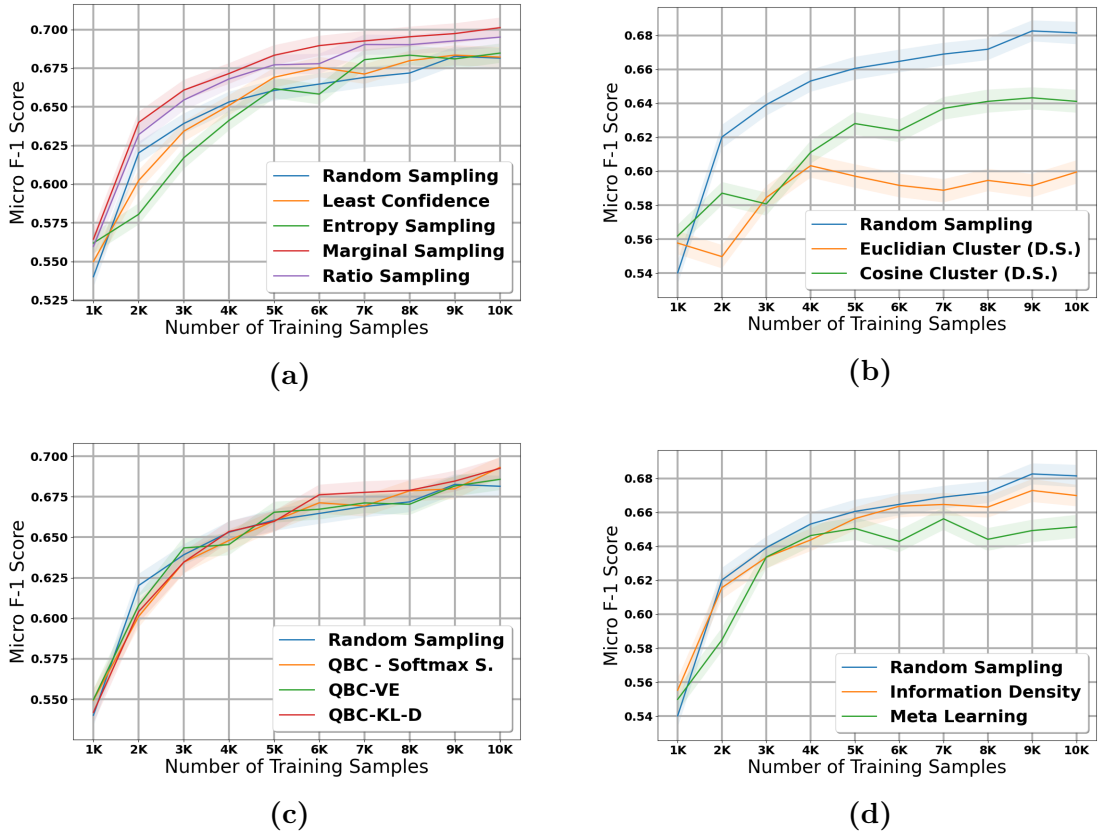


Figure 2.8: Micro score results for the 11 active learning algorithms applied during the small dataset experiment on subsite. Blue line represents random sampling.

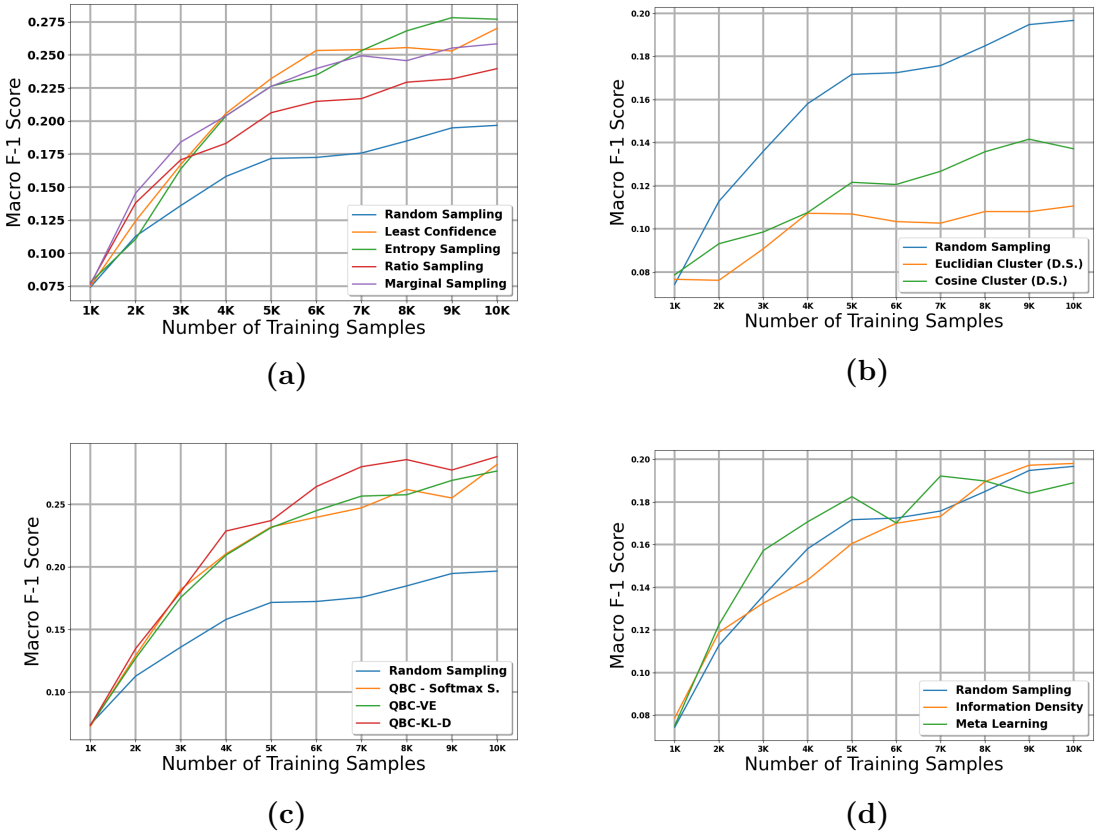


Figure 2.9: Macro score results for the 11 active learning algorithms applied during the small dataset experiment on subsite. Blue line represents random sampling.

iterations; however this difference is not huge and the other high-performing active learning strategies achieve similar macro F1 score in later iterations. Therefore, the choice of best active learning strategy may simply come down to choosing the most computationally efficient method that is fastest to run.

From the algorithms implemented in this paper, the QBC techniques are the most computationally expensive because each iteration of active learning requires training a full committee of models. Other techniques involving document embeddings and clustering techniques (information density and diversity sampling) are also compuationally expensive because they involve repetitive computation of distance between large vectors. The most computationally inexpensive active learning approaches are the uncertainty sampling strategies because they require the least additional computation beyond what is already provided by the base classification model. Consequently, these strategies are favored in settings with a high amount of initial labelled data and a large amount of labelled data added per iteration of active learning.

In our small dataset experiments, the marginal sampling and ratio sampling techniques obtained significantly better micro F1 scores than all other active learning techniques. However, these two methods did not maintain the best overall macro F1 scores; the best macro F1 scores were obtained by least confidence, entropy sampling, and the QBC approaches.

One possible explanation for this phenomenon is the impact of the majority classes on overall micro and macro F1 score. Both our histology and subsite tasks are characterized by extreme class imbalance. In the histology task, the 10 most common classes make up $\sim$60% of the dataset, and in the subsite task, the 10 most common classes make up $\sim$50% of the dataset; the class distributions are available in Additional File A.1. Thus, performing well on the majority classes greatly impacts micro F1 score while having a slight effect on macro F1 score.

On the large dataset with 15k initial labelled samples, the base model observes hundreds or thousands of samples from the majority classes. After a few additional

iterations of active learning, it is likely that the classifier achieves strong performance on the majority classes, and thus additional gains in overall micro F1 score must account for the minority classes. We see this trend across all the successful active learning strategies – the best performing strategies have strong performance in both micro and macro F1 score.

However, on the small dataset with 1K initial labelled samples, the classifier is unlikely to have strong performance on the majority classes (or any other classes) mainly because it has not seen a sufficient number of samples. Thus, active learning methods that focus on maximizing performance on the majority classes achieve better overall micro F1 score than strategies that focus on minority classes because the majority classes make up a larger portion of the test set; however, this may come at the expense of performance on the minority classes and therefore reduce macro F1 score. We observe this trend in our small dataset results – the two best performing strategies, marginal sampling and ratio sampling, do not have the best macro F1 scores after the initial two iterations of active learning.

A large number of unique labels with extreme class imbalance is a common property of many clinical text applications such as ours. To better understand how active learning affects performance on minority classes, we plot the class imbalance within the training dataset (Figure 2.10, Additional Files A.10, A.11, A.12, and A.13) and the number of unique classes seen in the training dataset (Figure 2.11, Additional Files A.14, A.15, A.16, and A.17) after each iteration of active learning. Not surprisingly, for any given active learning strategy, there is a direct correlation between the macro F1 scores, the class balance, and the number of unique classes seen by the model.

Furthermore, we also examine how active learning affects the distribution of the training data compared to random sampling by visualizing the document embeddings in the final training set for the small histology experiment after 10 iterations of ratio sampling and 10 iterations of random sampling (Additional File A.18). Documents embeddings are extracted from the penultimate layer of the CNN and reduced to 2D

via t-distributed stochastic neighbor embedding (TSNE), and we color each document embedding based off whether it belongs to a majority class (number of total samples in dataset above average) or minority class (number of total samples in dataset below average). After 10 iterations of random sampling, 89.4% of documents in the training set belonged to majority classes and 10.6% belonged to the minority classes, while after 10 iterations of ratio sampling 72.1% of documents in the training set belonged to majority classes and 27.9% belonged to minority classes. Compared to random sampling, ratio sampling increases the overall percentage of minority classes in the training set; our visualization shows that many of these minority class documents form new small, well-defined clusters or expand the size of other small, existing clusters. We hypothesize that these documents play a large role in improving macro F1 score. We also note that some of the documents from the minority classes end up in the center without any clear clustering. This is likely because active learning may choose samples from extremely rare classes that do not yet cluster due to lack of training data or ambiguous edge cases that are difficult to classify; these samples may negatively affect overall micro F1 score in our small dataset experiments.

Our analysis of class balance and unique labels supports our hypothesis that in high data availability environments with a large amount of initial labelled data, boosting performance on minority classes is important for micro F1 score. On the other hand, in low data availability environments with a small amount of initial labelled data, it is more important to focus on majority classes to improve micro F1 score. In the early iterations of our large dataset experiments, we see that the most effective active learning strategies (uncertainty sampling and QBC) also generate the highest class balance and unique classes in the training set. On the other hand, the two best active learning strategies in our small data experiments – marginal and ratio sampling – have lower class balance and unique classes compared to least confidence, entropy sampling, and the QBC strategies. Overall, this analysis suggests that in applications with a high number of unique labels and extreme class imbalance, active

learning can play an important role in mitigating class imbalance such that rare classes have better representation in a given labelled dataset.

As mentioned in our results, Figure 2.3 and Additional File A.6 show that macro F1 scores dropped in the later iterations of the large dataset histology experiments. The balance and unique class proportion plots explain this unintuitive phenomenon. The overall class balance of the training dataset is much higher in the early iterations than in the later iterations. This is because by the later iterations, there are few or no samples from the minority classes left in the holdout set; consequently, the dominant classes and the least informative documents start to fill out the training dataset. As we have seen, lower class balance also correlates with lower macro F1 scores. Thus, in the later iterations, while active learning achieves high micro F1 scores, the performance on uncommon classes decreases because they make up a much smaller portion of the training set.

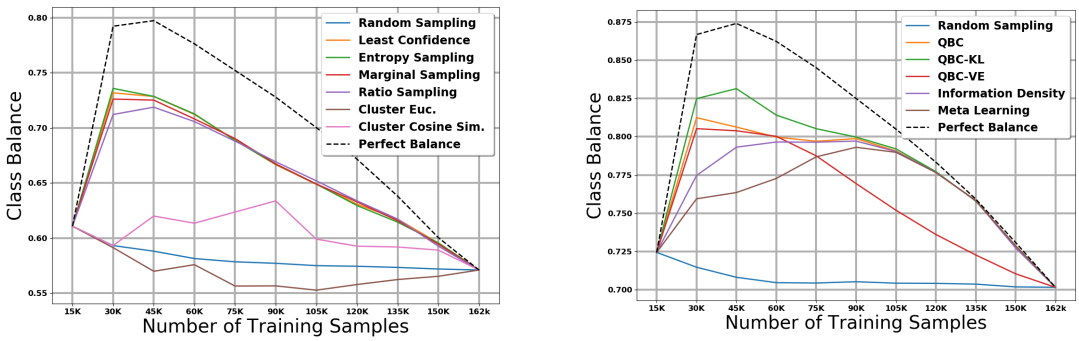
In Table 2.3, we summarize our findings regarding the effectiveness of each active learning strategy across three key characteristics – (1) overall performance in terms of micro and macro F1 scores, (2) effectiveness after only a single iteration of active learning with 1K additional labelled samples, which may be important in low resource settings where additional labels are difficult or expensive to obtain, and (3) computational cost to implement the strategy. Based on this table, we conclude that the ratio sampling and marginal sampling strategies are strong contenders for best overall active learning strategy because they have overall strong performance in both the high data availability and low data availability settings, have the best performance when additional labelled data is extremely limited, and are computationally very simple.

2.6 Conclusions

In this work, we evaluated the effectiveness of 11 different active learning strategies in the context of classifying cancer subsite and histology from cancer pathology reports.

Table 2.3: Summary of effectiveness of each active learning strategy in across different key characteristics.

	Overall Micro (Large)	Overall Macro (Large)	Overall Micro (Small)	Overall Macro (Small)	Single Iteration Micro (Small)	Single Iteration Macro (Small)	Compute Cost
Least Con.	High	High	Med	High	Low	Med	Low
Entropy Sam.	High	High	Med	High	Low	Med	Low
Ratio Sam.	High	High	High	Med	High	High	Low
Marginal Sam.	High	High	High	Med	High	High	Low
Euclidian C. (D.S.)	V. Low	V. Low	V. Low	V. Low	V. Low	V. Low	Med
Cosine C. (D.S.)	V. Low	V. Low	V. Low	V. Low	V. Low	V. Low	Med
Information Den.	High	High	Low	Low	Low	Low	Med
Meta Learning	Med	Med	Low	Low	Low	Low	Med
QBC - S.S.	High	High	Med	High	Low	High	High
QBC - VE	High	High	Med	High	Low	High	High
QBC - KL-D	High	High	Med	High	Low	High	High

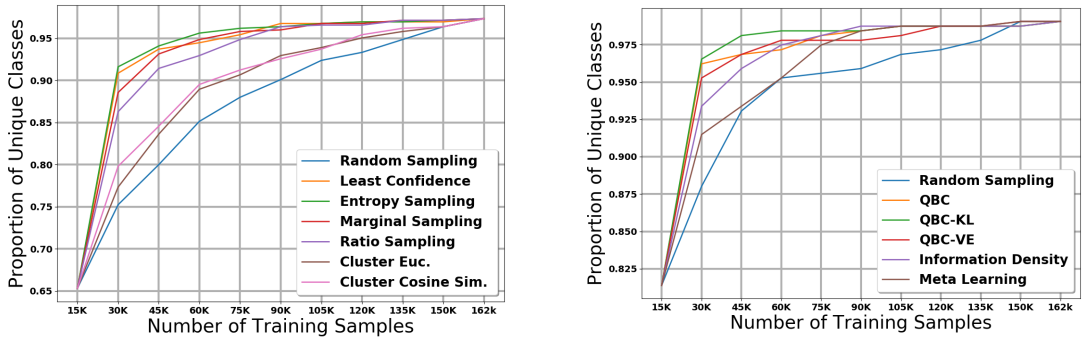


(a) **(b)**

Figure 2.10: Class imbalance Plots. Black line represents the upper limit (most balance dataset possible). Y-values are computed with Eq. 2.13: $y = 0$ represents no balance, and $y = 1$ represents full balance.

Our dataset is characterized by a large number of unique labels with extreme class imbalance, and we use a text CNN as the base classification model. For each of our two classification tasks, we tested under two different active learning scenarios – (1) a high data availability setting where we start with 15K labelled samples and added an additional 15K labelled samples after each iteration of active learning, and (2) a low data availability setting where we start with 1K labelled samples and added an additional 1K labelled samples after each iteration of active learning. After each iteration of active learning, we reported the micro and macro F1 scores of the classifier as well as the class balance and unique labels in the training dataset.

We showed that in the high data availability setting, the uncertainty sampling and QBC strategies obtained the best overall micro F1 scores, and the QBC KL-D strategy obtained the best overall macro F1 score. In terms of micro F1 score, there was no single clear winner. In the low data availability setting, ratio and marginal sampling achieved the strongest overall micro F1 scores but underperformed slightly in macro F1 scores; least confidence, entropy sampling, and the QBC strategies obtained the best macro F1 scores. Ratio and marginal sampling are strong contenders for the overall best active learning strategy based on overall performance in the high and low data availability settings, performance when additional labelled data is extremely limited, and low computation cost. Compared to a model trained on all available data, active learning can obtain similar performance using less than half the data. Furthermore, on tasks with a large number of unique labels with extreme class imbalance, active learning can significantly mitigate the effects of class imbalance and improve performance on the rare classes.



(a) **(b)**

Figure 2.11: Proportion of classes seen by the models at each iteration. The y values consists of the number of unique classes present in the training dataset divided by the total number of classes in each task (525 for histology and 317 for subsite)

.

Chapter 3

Class imbalance in out-of-distribution datasets: Improving the robustness of the TextCNN for the classification of rare cancer types

Disclosure Statement

A version of this chapter was originally published in the Journal of Biomedical Informatics:

Kevin De Angeli, Shang Gao, Ioana Danciu, De Angeli, K., Gao, S., Danciu, I. et al. Class imbalance in out-of-distribution datasets: Improving the robustness of the TextCNN for the classification of rare cancer types, Journal of Biomedical Informatics 125, 2022. https://doi.org/10.1016/j.jbi.2021.103957.

Authors' contributions: KD: conceptualization, investigation, methodology, software, validation, visualization, writing original draft, writing, review, editing. SG: Conceptualization, Methodology, Writing, Review, Editing, Project administration. ID: Conceptualization, Methodology, Writing, Reviewing and editing. NS, XW, ED, JD, AS, LC, and LP: Data Curation. GT, LP: Funding acquisition, supervision. HY: conceptualization, formal analysis, investigation, methodology, writing, supervision, Software.

No revisions to this chapter have been made since the original publication.

3.1 Abstract

In the last decade, the widespread adoption of electronic health record documentation has created huge opportunities for information mining. Natural language processing (NLP) techniques using machine and deep learning are becoming increasingly widespread for information extraction tasks from unstructured clinical notes. Disparities in performance when deploying machine learning models in the real world have recently received considerable attention. In the clinical NLP domain, the robustness of convolutional neural networks (CNNs) for classifying cancer pathology reports under natural distribution shifts remains understudied. In this research, we aim to quantify and improve the performance of the CNN for text classification on out-of-distribution (OOD) datasets resulting from the natural evolution of clinical text

in pathology reports. We identified class imbalance due to different prevalence of cancer types as one of the sources of performance drop and analyzed the impact of previous methods for addressing class imbalance when deploying models in real-world domains. Our results show that our novel class-specialized ensemble technique outperforms other methods for the classification of rare cancer types in terms of macro F1 scores. We also found that traditional ensemble methods perform better in top classes, leading to higher micro F1 scores. Based on our findings, we formulate a series of recommendations for other ML practitioners on how to build robust models with extremely imbalanced datasets in biomedical NLP applications.

3.2 Introduction

One of the tasks of the Surveillance, Epidemiology, and End Results (SEER) program of the National Cancer Institute (NCI) is to provide statistics and analyze cancer trends in the US. Every year, cancer registries receive thousands of electronically transmitted pathology reports from pathology laboratories. These pathology reports consist of unstructured clinical text. Specialized human annotators are required to extract valuable information. This process is costly and time consuming. Therefore, developing reliable models to classify cancer pathology reports automatically remains one of the priorities of SEER.

Recently, numerous machine learning researchers have shown that models which are trained in labs often exhibit a significant performance drop when they are deployed in the real world. Some researchers have identified certain aspects of the modeling process and model training as the source of performance disparity [18]. Others have perceived the issue as a robustness problem and focused on building models that learn generalizable features so that they can maintain their performance under natural distribution shifts. Performance drop at deployment time is a serious issue that could affect every machine learning practitioner and the reliability of AI systems [18].

Significant research progress has been made in developing deep learning models for information extraction from cancer pathology reports [36, 39, 38, 20]. Difficulties

in sharing data between healthcare systems has made analyses of the performance of models on out-of-distribution (OOD) datasets very challenging. As registries around the country are legally required to collect cancer pathology reports for all residents of their state, natural statistical variations arise in the datasets. These variations may occur due to different data acquisition pipelines as the pathology reports can come from different laboratories, disparate disease prevalence patterns, or the evolution of language and reporting protocols over time [96].

Class imbalance occurs when the proportion of samples belonging to one or more classes in a dataset varies drastically. Class imbalance is of extreme importance for the clinical NLP domain because many clinical conditions such as certain cancer histologies are very rare. For example, extracting the histology code from pathology reports is a task that involves text classification with over 600 classes. Some types of cancers are extremely common (e.g. adenocarcinoma, ductal carcinoma), while others occur less frequently (e.g. squamous cell carcinoma). These phenomena lead to datasets with extreme levels of class imbalance, which impacts the performance of the classifiers. Models trained with such datasets will exhibit bias towards the most prevalent majority classes because of their higher prior probabilities, while often ignoring the minority classes. Although the effects of class imbalance in machine learning have been well documented [43, 79, 4], previous researchers have concurred that class imbalance in the context of deep learning is understudied [54]. Additionally, most of the existing work on class imbalance has been done in the computer vision field [6].

The class imbalance problem can become crucial when deploying models to classify pathology reports on unseen registries from states and geographical regions outside the training data (OOD datasets). The distribution between the minority groups may differ widely across registries, and classifiers' bias towards the top classes leads to deteriorated deployment performance. Few researchers in the clinical NLP domain have focused on characterizing and improving the performance of deep learning models under natural distribution shifts. Our aim is to narrow this literature gap with

a specific focus on minority classes (rare cancer types). Our contributions are as follows:

- We show that natural distribution shifts have a considerable impact in performance when classifying cancer pathology reports.
- We identify class imbalance and class distribution as one of the sources of performance drop at deployment time, and analyze some of the existing methods to solve the problems associated with these issues.
- To strengthen the consistency of our results, we compare the methods under two different classification tasks commonly found in cancer pathology reports: histology and subsite.
- We demonstrate that an ensemble of 12 CNNs can improve the generalization power at deployment time. However, we show that most of the performance gain comes from the majority classes.
- We propose a novel implementation of ensemble learning where each model specializes in a different group of classes. Our class-specialized ensemble outperforms other class imbalance techniques in terms of macro F1 scores when testing on unseen registries while maintaining competitive micro F1 scores.

Our research is at the intersection of robustness and class imbalance for clinical NLP. Our novel ensemble model, which improves performance in rare cancers, is generalizable and can also benefit other practitioners working on problems related to bias and fairness in machine learning.

3.3 Previous Work

3.3.1 Class Imbalance

In this section, we present existing work in the class imbalance literature that is relevant to our research. Generally, class imbalance techniques in the context of machine learning are grouped into data-level techniques and algorithm-level methods. We note that our specific problem involves extreme levels of imbalance (described in detail in our methods 3.4.7), which are uncommonly observed in previous works. Nevertheless, some of their methods, results, and findings are still applicable.

Data-level techniques focus on manipulating the distribution of the training dataset in order to reduce the imbalance present in the original data. The two most basic paradigms in this group are: 1) random oversampling (ROS), where samples from minority classes are duplicated, and 2) random undersampling (RUS), where samples from majority classes are discarded.

Masko et al. [67] presented a comprehensive study of the effects ROS using CNNs for image classification. They used the MNIST, ImageNet, CIFAR-10, and CIFAR-100 datasets. They performed experiments with relatively small levels of imbalance and showed that ROS improved the baseline scores.

The effects of RUS have also been studied extensively. For example, Kubat et al. [57] presented an algorithm that selectively removed samples from the majority classes. The downside of their study is that they only focus on 2-class datasets. Hulse et al. [106] developed an extensive analysis of seven sampling techniques using 35 benchmark datasets and 11 classifiers. They found that the performance of the sampling techniques is dependent upon the machine learning model and showed that, in some circumstances, RUS can outperform other classical techniques.

Dynamic sampling is a technique which combines both RUS and ROS. Pouyanfar et al. [75] developed this sampling strategy based on the way humans often operate: repeating a certain task until the error is reduced. Thus, the researchers created an

algorithm which adjusts the distribution of classes in the training dataset based on a performance metric (e.g. F1 score). As a result, majority classes are expected to be undersampled while minority classes will be oversampled. Chawla et al. [13] proposed a novel technique called SMOTE which also uses a combination of RUS and ROS. It had previously been noticed that simply oversampling minority documents with replacement does not improve minority performance significantly [66]. For this reason, the authors developed a special case of oversampling which selects synthetically created samples from the minority classes.

Although the simplicity and efficiency of methods that combine ROS and RUS may seem appealing, in applications with extreme levels of class imbalance such as ours (where the top classes appears 17k times and the bottom class appear only once), oversampling repeatedly from the same documents within minority classes will only force the model to memorize features that may not even be useful for the respective classes. In addition, event though SMOTE is a standard class imbalance tool for traditional machine learning [45, 29, 104], it still has important limitations when it comes to deep learning models. Some of the challenges come from the implementation of the algorithm itself. For example, in problems such as ours when the input to a model is a matrix composed of word vectors, sampling the K nearest neighbors is not adequate. Moreover, previous researchers have demonstrated that in situations where high-dimensional data are common, SMOTE does not improve model performance [7]

Algorithm-level methods for class imbalance focus on modifying the learning process without altering the distribution of the dataset [54]. The most popular paradigm in this group is cost-sensitive learning, where models are penalized for the classification of certain (minority) classes. The cost associated with the misclassification of each class is assigned using a cost matrix, where an entry $C_{i,j}$ in this matrix represents the cost of predicting class i for the true class j. In the context of text classification, Padurariu et al. showed that cost-sensitive methods can outperform data-sampling methods [72]. Previous researchers have noted that

the biggest challenge of cost-sensitive methods is building an effective cost matrix [54]. Depending on the specific problem, experts could use previous knowledge to define costs. However, in complex problems with a lot of classes and extreme level of class imbalance, coming up with an optimal cost matrix is a serious challenge.

Some authors have built novel approaches for class imbalance that borrow ideas from both data and algorithm-level methods. This is the case of Lee et al. [59] who showed that a particular implementation of transfer learning, also known as two-phase learning, can outperform other classical class-imbalance techniques. Their application involves the classification of plankton images using CNNs. During the first phase, they trained a model with a subset of the data using some threshold N. In this subset of data, samples are rejected so that the frequency of each class present in the dataset does not exceed $N = 5000$ (found experimentally). The authors' reasoning is that the model trained with the thresholding data is less biased, and it can learn features that are relevant for the minority classes, but it loses population information. Therefore, to recover the lost information, they fine-tune the model with the entire dataset. The authors compared two-phase learning with other models trained with noise addition, data augmentation, and a combination of both. Our class-specialized ensemble method presented in this paper was partially inspired by their two-phase learning implementation.

For a detailed analysis of previous results in the imbalance literature, we refer to [54], a survey paper where the authors review 15 deep learning methods for class imbalance. Their extensive review discusses all three types of techniques: data-level methods, algorithmic-level methods, and hybrid-methods. For an overview of imbalance methods focusing specifically in text classification, we recommend [72]. Additionally, [79] provides a comparative study of different data-level methods.

3.3.2 Robustness

Numerous authors have recently identified and evaluated the disparities in a model's performance during deployment [18, 100, 96, 68, 97, 48, 24, 111, 42]. From the pool of existing research, notable work includes [18], where the authors identified underspecification as a key factor diminishing the reliability of machine learning systems. In their paper, they performed a series of stress tests and showed how different modeling aspects, even as simple as a random seed, can lead to almost unpredictable performance when deploying a model in the real world. Although the authors provided substantial examples of the performance discrepancy when testing in OOD datasets, a distinct solution was not provided.

In computer vision, prior work has often focused on the ImageNet dataset using CNNs. For example, in [100], the authors analyzed the reliability of robustness techniques which were developed using datasets with synthetic distribution shifts. They showed that most of the existing techniques are not effective under natural distribution shifts, and they found that most improvement comes from data size and diversity. Conversely, Hendrycks et al. [48] argued that using synthetic data can improve the performance of a model on OOD data. In addition, they built three robustness benchmarks for image classification and introduced a new data augmentation technique. Djolonga et al. [24] also used the ImageNet dataset, but their analysis focused on the effects that data/model scale and transfer learning have in OOD performance. Their conclusion is that given the limitations associated with data and model scaling, transfer learning is the most promising approach in the short term. In this study, we analyzed the effects of transfer learning through our two-phase learning implementation.

In the clinical field, Stacke et al. [96] presented a technique to quantify how robust a model is to domain shifts and how to identify new data for which the model would struggle to generalize. They achieve this by measuring the differences in feature representation by an arbitrary model. Their specific application is tumor classification

from images. Although the authors provide a useful metric to quantify the robustness of a model, they do not focus on the aspects of the learning process which enhance the models' performance.

In the context of NLP, Wu et al. [111] approached the OOD robustness problem by modifying existing models to produce multiple disentangled representations. They argue that it is important for a model to separate between general, target-specific, and source-specific features. Intuitively, their approach is an ensemble of models combined together into a single architecture. The down-side of their study is that they used datasets with very few classes, making it hard to predict the efficiency of their methods in more challenging problems.

3.3.3 Ensemble Methods

Ensemble learning [74] is a machine learning technique that solves a given task with multiple models. The purpose of applying multiple models is to obtain collective decisions from them, thus reducing the likelihood of incorrect selections. Aggregating decisions from multiple models adds generalizability to the outcomes, improving overall task performance and avoiding overfitting the training dataset. This is a desirable feature for the classification of under-represented class labels.

Since ensembles of classifiers combine decisions from multiple models, the individual models should exhibit some level of diversity. Bootstrap aggregation [9], also known as bagging, is a popular technique that infuses variability via the bootstrapping of the training samples. However, a recent study [69] demonstrated that the intrinsic variability from the randomized initial values of trainable parameters in artificial neural network-based models adds enough variability.

Ensemble learning does not necessarily require the models to be trained in the same feature space. Combining models of multiple local experts trained by different portions of the feature space is an alternative ensembling technique. In this approach, inferring the final decision can be done with an additive classifier by concatenating

the outputs of local experts (stacked generalization) [121]. Another way to infer the final decision is by the use of a gating network to determine a generalized linear rule, a method known as mixture-of-experts (MoE) [93]. Our class specialized ensemble technique borrows some ideas from the MoE method.

In the ensemble learning literature, one particular work that is relevant to our research is [49]. Here, the authors train an ensemble of models where there is a generalist (trained in the entire dataset) and multiple specialists (trained on a confusable set of classes in the dataset). Using the MNIST dataset, the authors shows that the specialist ensemble outperforms their baseline ensemble by ∼3%. Their research shares some conceptual similarities with MoE, and therefore it is applicable to our research. However, their implementation of "ensemble of specialists" is completely different: we do not separate the models between generalists and specialists, and we focus specifically on class imbalance and rare classes.

3.4 Methods

3.4.1 CNN Architecture

The baseline for our experiments is a standard TextCNN used extensively in previous work involving cancer pathology reports classification [20, 2, 1, 83] and clinical text classification in general [115, 51, 81, 46, 113]. In addition to being an universally used architecture, previous work showed that, for the task of pathology report classification, the TextCNN has competitive performance with other machine learning models [2], including BERT-based approaches [35]. We used this base TextCNN for every model in this study with some training variations described in greater detail in the following subsections. The network consists of an embedding layer followed by three parallel convolution layers with filter sizes of 3,4, and 5 consecutive words and 300 filters each, a global max pooling layer, and a dense layer. The network has ∼91

million trainable parameters, where ∼90 million of them belong to the embedding layer.

3.4.2 CNN with Class Weights

When training DL models, one can simply implement cost-sensitive learning by using custom class weights. These weights dictate how the model will be punished by the misclassification of certain classes. Thus, assigning higher weights to minority classes would force the model to pay special attention to these classes. There is a lot flexibility on how to assign weights to each class. After experimenting with an inverse frequency function, we found that giving minority classes too much weight brings the micro score down to non-permissible levels. That is because the proportion of the most rare cancer types in the dataset is extremely low, which leads to excessively high weights for the rarest classes. As a result, the model focuses on learning features for these rare classes and ignores the majority classes, which highly impacts the micro F1 score.

For our class weight implementation, we used a variation of the inverse class frequency where minority classes are assigned non-excessive, larger weights. we set the class weights WC_c following Equations 3.1 and 3.2.

$$weight_c = \log(\frac{|Y|}{|y_c|}) \tag{3.1}$$

$$\begin{cases} WC_c = weight_c & weight_c > 1 \\ WC_c = 1 & otherwise \end{cases} \tag{3.2}$$

In the equations above, $|Y|$ is the total number of samples in the dataset, and $|y_c|$ is the number of samples belonging to class c. Using this rule gives a weight of 1 to the majority classes and a class weight close to 14 for the most rare cases.

This approach gives higher importance to the rarest cancer types without ignoring the majority classes.

3.4.3 Two-phase Learning

We implemented a version of two-phase learning originally introduced by Lee et al. [59]. In their paper, the authors first train the model with a class-normalized dataset which has a thresholded class distribution. Due to extreme imbalance and the large frequency of the top classes, we implemented a variation of this method in which the top 50 classes are completely left out during the first phase of training. The model is then fine-tuned with the entire dataset during the second phase. We tried a standard version of two-phase learning without class weights, and we also tried another version in which class weights are introduced (as described in 3.4.2) during both learning phases.

3.4.4 Undersampling

In the simplest form, undersampling methods discard a portion of the majority class to balance the dataset. In problems with moderate levels of class imbalance, one can simply discard majority class samples until reaching equal number of samples with the minority classes. For our specific problem, discarding the top classes based on the frequency of the rarest cancer types is not possible because these rare classes appear at extremely low proportions (see Section 3.4.7 and B.1). Alternatively, we discarded a number of documents from the top classes using a threshold based on certain percentiles (50^{th}, 90^{th}, and 95^{th}) of class frequency. The specific implementation procedure is described as follows:

- Find number of documents belonging to the class in the respective percentile (50^{th}, 90^{th}, and 95^{th}). We call this value the undersampling threshold α.

- Discard documents from the dataset so that there are at most α documents in each class. No documents are discarded for classes with fewer than α samples.
- Train model with this smaller, more balanced dataset.

3.4.5 Class-Specialized Ensemble

Our novel method was inspired by MoE and two-phase learning with class weights. We wanted to create an ensemble of models were each TextCNN would specialize in different group of classes. Thus, we first ordered the classes by frequency based on the training and validation datasets. Then we created groups of 50 (histology) and 28 (subsite) classes based on their frequency order. The reasoning behind forming frequency-based groups is that the imbalance between the individual groups will be reduced, as opposed to creating groups of classes selected randomly. We decided to use group sizes of 50 and 28 because that will keep the ensemble relatively small (12 models). However, one could easily experiment with having larger ensembles which specialize in smaller groups of classes.

During the first learning phase, we let individual members of the ensemble learn features that are key for their assigned class group. Then, each member was fine-tuned with the entire dataset. For example, during the first training phase of the histology task, we trained one TextCNN with the top 50 classes (classes 0-49) and another TextCNN with the second group of 50 classes (classes 50-99), and so on. During the second learning phase (fine-tuning), we trained each of the models with the entire dataset. Figure 3.1 shows a general overview of the steps we took to train the class-specialized ensemble.

To aggregate the individual predictions of the ensemble and generate the final prediction, we use a simple multilayer perceptron (MLP) model. This MLP model is trained with input vectors that are created by concatenating the softmax vectors from each of the 12 models in the ensemble and the respective y label associated with the documents. Thus, for the case of histology where there are 645 classes,

the input to the MLP is a vector of size 7740 (number of classes multiplied by the number of models). The architecture consists of two dense layers with 4000 and 3000 neurons, respectively. We also included a dropout layer between each of the dense layers (dropout rate = 0.5). The number of layers, neurons, and hyperparameters were found experimentally. The MLP network has ∼47 million trainable parameters.

3.4.6 Ensemble

Since our proposed model involves an ensemble of models which naturally presents an advantage against individual models, we implemented two traditional ensemble learning techniques. We used an ensemble of 12 models to be consistent with our class-specialized method and perform a fair comparison.

The first ensemble technique implemented is majority voting. Here, the final prediction is the class that is predicted the most often across the 12-model ensemble (ties are resolved by randomly selecting one of the classes with the most votes). The other technique is softmax averaging. This method consists of taking the average of the softmax vectors across the ensemble. For example, for the ensemble trained in the histology task, we simply take the average of 12 vectors (ensemble size) of size 645 (number of classes) and then predict the class with the highest softmax value in this average vector.

We note that our selection of ensemble methods can easily be applied by other machine learning practitioners, it is highly parallelizable, and it is computationally cheap compared to other ensemble methods.

3.4.7 Dataset

The dataset consists of cancer pathology reports from the Louisiana Tumor Registry (LTR), Kentucky Cancer Registry (KCR), Utah Cancer Registry (UCR), New Jersey State Cancer Registry (NJSCR), Seattle Cancer Registry (SCR), New Mexico Cancer Registry (NMCR), and California Cancer Registry (CCR). The total number of

pathology reports from these seven registries is 2,059,758 documents. Table 4.2 shows the size of each of the individual datasets associated with the seven registries. We use numerical values instead of the actual registry names to preserve anonymity. Even though there are other tasks associated with our dataset (site, laterality, and behavior), in this study we focus on the histology and subsite tasks because these are the top priority for NCI; our labels are based on the ICD-O-3 system from the World Health Organization Classification of Tumors [86]. Additionally, histology and subsite have the largest numbers of classes and highest level of class imbalance, making them good targets for our robustness study.

There are 645 histology classes, and the dataset presents extreme cases of class imbalance. For example, 22.0% of the reports belong to the top class (adenocarcinoma in situ/NOS) and 19.0% belong to the second most popular class (duct carcinoma). The top 10 classes constitute 62.8% of the dataset. The least prevalent 635 classes constitute only 37.2% of the data. Some of the cancer types (31 classes) are exceptionally rare and only appear once in the entire dataset. Although removing these classes could make sense from a modeling perspective, these cancer types may still be encountered at deployment time, and they are still part of the classification problem. Therefore, all the classes were considered during training.

Identifying the subsite of a cancer pathology reports is a task with 327 classes. The level of imbalance found in this task is still high but slightly lower than what we observed in the histology task. Here, only 8.9% of the reports belong to the top class (compared to 22.0%), and the top 10 classes constitute 49.5% of the documents (compared to 62.8%). Just as in the histology task, there are cancer subsites in the dataset which are extremely rare. For example, 16 cancer subsites appear less than ten times in the dataset.

Researchers in the class imbalance field often use metrics to quantify the levels of imbalance in the datasaet. For example, one common metric is $\rho = \frac{max_i(|C_i|)}{min_i(|C_i|)}$. Where $max_i(|C_i|)$ and $min_i(|C_i|)$ represents the number of samples in the top class and the bottom class, respectively. Computing this value for histology leads to $\rho = 452,363$.

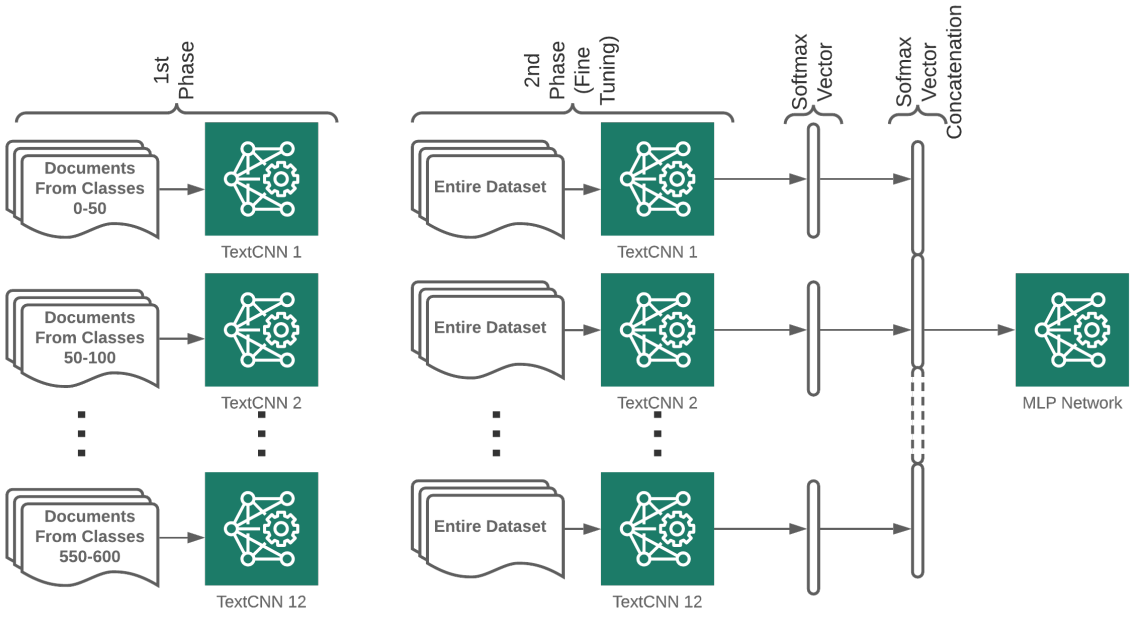


Figure 3.1: Training pipeline of the proposed model. The MLP network takes as input a vector of concatenated softmax vectors and their respective Y label.

Table 3.1: Number of pathology reports in each individual dataset.

Registry	R1	R2	R3	R4	R5	R6	R7
Dataset Size	85,789	577,094	137,135	92,481	441,732	360,375	365,152

We note that this value is substantially larger than what one usually finds in previous work, which further demonstrates the extreme levels of imbalance in our problem.

3.4.8 Experimental Setup

For our experiments, we took a leave-one-out approach. In each run, we first define which of the seven registries will be the OOD dataset. This dataset is left-out of the training process. The other remaining six registries are then combined and shuffled. The combined dataset represents what a machine learning practitioner may be given to train a model in a lab setting and the left-out registry represents what one may find when deploying the model in the real world. After training the model with the combined dataset, we recorded performance metrics for both the test set from the combined dataset and the left-out (OOD) registry. In order to consider every possible combination case, we repeat this process seven times for each of the two tasks. Thus, every registry is used as the OOD dataset once. This experimental setup leads to a total of 14 individual results (7 possible dataset combinations and two tasks).

We used standard training practices to prevent serious overfitting issues. We performed a 80/10/10 train-validation-test split on the combined dataset. At the end of each epoch, we monitored the validation loss. We let the models train until the validation loss stopped decreasing for five consecutive epochs. Once training stops, we recovered the best set of weights based on the validation loss. To further prevent overfitting, our CNN model uses 50% dropout on the dense layer (Section 3.4.1).

The parameters and software used in this study are similar to previous work involving pathology report classification [20, 2, 1]. We used Keras 2.3 with the Adam optimizer, a batch size of 128, and a learning rate of 1e-4.

We set the document length size to 1500 words, meaning that longer documents are truncated and shorter documents are zero-padded. The word embeddings consist of vectors of size 300 which were randomly initialized; previous studied showed that

random embeddings are as effective as other pre-trained word embeddings when applied to our dataset [78].

All experiments were run on individual NVIDIA V100 GPUs. The ensemble models were trained in parallel, with their output combined to form the final predictions.

3.4.9 Evaluation Metrics

We evaluated the performance of the models by computing the micro F1 score (Equations 3.3-3.5) in the test dataset and in the OOD dataset. We note that for our problem, micro F1 score is equivalent to accuracy.

When working with highly imbalanced datasets, using micro F1 scores can be misleading. That is because the majority classes will drive a large portion of this score, and information about the model performance on the rare classes is lost. In order to better understand the performance of the model in minority classes, we also calculated macro F1 scores (Equation 3.6). The macro F1 score is a common evaluation metric used in problems with class imbalance because it averages the model's accuracy on individual classes independently of their frequency in the datasets. In other words, this metric gives equal importance to every class.

$$\text{Precision} = \frac{True\ Positive}{True\ Positives + False\ Positives} \tag{3.3}$$

$$\text{Recall} = \frac{True\ Positives}{True\ Positives + False\ Negatives} \tag{3.4}$$

$$\text{Micro F1} = 2 * \frac{(Precision) \times (Recall)}{Precision + Recall} \tag{3.5}$$

$$\text{Macro F1} = \frac{1}{|C|} \sum_{C_i}^{C} F1(C_i) \tag{3.6}$$

In Equation 3.6, $F1(C_i)$ is the accuracy score for class i, and $|C|$ represents the total number of classes in the dataset.

3.4.10 Performance on Rare Cancer Types

We performed an additional analysis of the model performances in minority classes. This in-depth study focused on the histology task because it has a larger number of classes and more extreme levels of class imbalance. For this analysis, we sorted the classes by frequency in the dataset and then created groups of 50 classes so that the first group contains the top 50 classes and the last group consists of the 50 least common classes. Then, we used the models that were trained in the entire datasets to predict on each of the specific groups. The motivation for this analysis is to gain more insight into the models' performance on minority classes beyond a single macro F1 score.

3.5 Results

3.5.1 Class distribution in OOD datasets

Given the large number of classes in the two tasks considered, we hypothesize that the class distribution in the training and OOD datasets will differ considerably for the rare cancer types. In order to test this hypothesis, we plotted the distribution of classes as percentages of their respective datasets in Figure 3.2, using registry 6 as the OOD dataset. We note that for the top classes (Figure 3.2a), the distribution is similar in both datasets. However, there are substantial differences in the distribution of the least common cancer types (Figure 3.2b).

The implications of these distributional differences are one of the focus of this study. Deep learning models are known to be biased towards the top classes since they drive the loss function. Hence, the distribution of classes will affect the features it learns and which classes receive higher priority. The distribution of minority classes

differs greatly across registries leading to a large underperformance of the model in OOD datasets in terms of macro scores.

3.5.2 F1 Scores

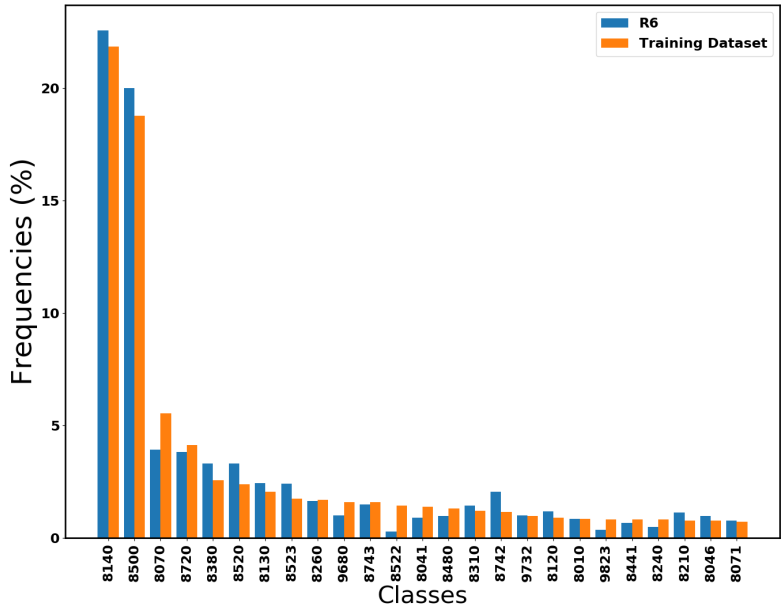
Our experimental setup yields seven individual sets of results (one for each registry left-out) for each of our two tasks. Here, we present the average scores across the seven registries for histology (Table 3.2) and subsite (Table 3.3). The pattern found in this table is representative of the outcomes found in the individual registry results, but the scale may differ slightly (see Appendix B.1 for individual registry results).

We found that training a baseline CNN leads to decent micro scores but low macro scores. This was an indication of the model's bias against the top classes: it learned features that were important to classify common cancer types and reduce the loss, but it ignored patterns that were relevant for the more rare cancer types.

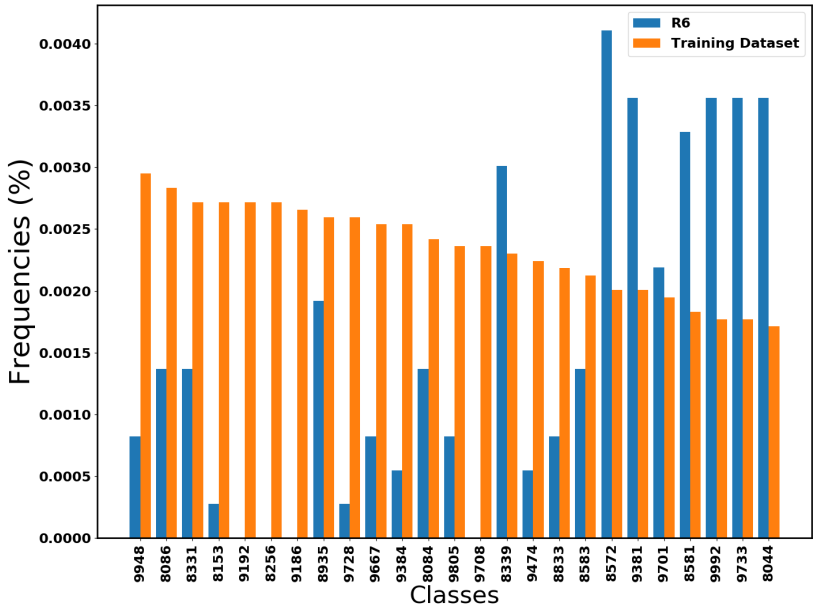
Adding class weights to the model helped the minority classes but it also hurt the performance on the majority classes (in the case of the histology task). We note that this technique is prone to produce a trade-off between micro and macro scores, since giving special attention to minority classes will reduce the performance on top classes.

Using two-phase learning without weights effectively improved the test macro scores in both tasks by a significant amount ($\sim$7% and $\sim$4%). It also improved the micro test scores ($\sim$1% for both tasks). This method also introduced a small decrease in OOD micro. In terms of OOD macro scores, the improvement was small ($\sim$1%) for histology, and there was no improvement for subsite.

Two-phase learning with class weights pushed the macro scores further but resulted in mediocre micro scores that were below the baseline CNN. We also note that the increase in macro scores between the test and OOD datasets was highly disproportional when comparing it with the baseline CNN: for histology, the test macro increased by $\sim$10% and the OOD macro increased by $\sim$2% . For subsite, the



(a) Classes 0-24 (top 25 major classes)



(b) Classes 350-374 (a subset of the minority classes)

Figure 3.2: Differences in class distribution between the training data and registry 6 (see Section 3.4.7) for the histology task. The specific class names associated with the encoded labels can be found in the SEER website [86].

test macro increased by ∼9% and the OOD macro increased by only 0.2%. In both cases, the test macros were the highest scores across the board. The disproportional increase in macro scores between test and OOD is a potential sign of overfitting in some of the minority classes. Two-phase learning is potentially more susceptible to overfitting the minority classes since these documents are introduced during both training phases.

Ensemble methods outperformed other models in terms of micro test and micro OOD. They also provided high macro scores when compared to single models. Between the two standard ensemble methods that we implemented (majority-voting and softmax-average), the scores were similar for subsite. For histology, taking the average of the softmax vectors across the ensemble showed slightly better performance.

Our class-specialized ensemble method produced the highest OOD macro score across the board, outperforming the baseline CNN results by ∼4% for both tasks. It also produced higher test macro scores (between ∼3% and ∼4%) than the standard ensemble methods. In terms of micro scores, the class-specialized model outperformed the baseline CNN but performed slightly worse than the other ensemble models.

We also compared the performance gap between test and OOD scores for all models (Table 3.4). Higher values in this table indicates lack of robustness. We observed that two-phase learning led to the highest performance gap. The implication of this result is that methods like two-phase learning can be highly misleading when trained in a lab setting without access to an OOD dataset because lab results may not correlate with real world performance. The table also shows the baseline CNN with class weights resulted in the smallest performance gap.

During our experiments, we found that undersampling is not an effective technique for the classification of cancer pathology reports. Discarding documents from the majority classes diminishes the model micro F1 scores to non-permissible levels with no significant improvement in macro F1 scores. The results table with different undersampling thresholds are included in Appendix B.2 (Section 3.4.4, Table B.1)

We note that in task such as ours, with extreme class imbalance and a large number of classes, it is common to observe relatively low macro F1 scores. Classifiers are not able to correctly identify features that are relevant for classes that are extremely rare. This is exacerbated by the lengthiness of cancer pathology reports – on average each report is approximately 700 words in length, so it is difficult to distinguish which words are relevant to a particular class when there are very few samples. In our previous work, we demonstrated that low macro scores persist across different deep learning architectures ([2]).

3.5.3 Classification Performance in Minority Classes

Figure 3.3 shows the micro F1 scores obtained when predicting in different class groups ordered by frequency. In the case of the test dataset, we observed that two-phase learning with class weights clearly outperforms other methods for all the groups, except for the first one (the top 50 classes). However, for the OOD dataset, the differences in performance becomes smaller and class-specialized ensemble outperforms other methods for some of the groups. We note that the top 50 classes drive most of the micro F1 score, and we observe that models which excel in this group often show lower performance in the rest of the groups.

Figure 3.2a (also see 3.4.7 for exact percentage values) shows that the top two classes were especially common in the dataset. These two classes have a lot of influence during the training phase because they drive most of the loss function. Models which focus on learning features that are important for top classes will show degraded performance on the rest of the classes while obtaining high F1 micro scores. Therefore, we also analyzed the performance of the models when these two top classes are left out at testing time. The motivation of this experiment was to understand how much bias is involved in the learning process. We were interested in observing which model captures the most characteristics relevant for the non-top classes. The results for this experiment complement the macro F1 score further, and provide a

Table 3.2: Histology Results. Overall micro and macro scores for the test and the out-of-distribution data (unseen registry). Scores were calculated by taking the average of the individual results for each of the seven registries.

Model	Test Micro	Test Macro	OOD Micro	OOD Macro
CNN	0.8007	0.4089	0.7749	0.3552
CNN w/ Class Weights	0.7885	0.4104	0.7704	0.3677
Two-Phase	0.8071	0.4815	0.7723	0.3624
Two-Phase w/ Class Weights	0.7942	**0.5169**	0.7631	0.3771
Ensemble (Maj.Vot.)	0.8096	0.4373	0.7866	0.3781
Ensemble (Softmax. Avg.)	**0.8119**	0.4458	**0.7876**	0.3841
Class-Specialized Ensemble	0.8085	0.4809	0.7778	**0.4003**

Table 3.3: Subsite Results. Overall micro and macro scores for the test and the out-of-distribution data (unseen registry). Scores were calculated by taking the average of the individual results for each of the seven registries.

Model	Test Micro	Test Macro	OOD Micro	OOD Macro
CNN	0.7133	0.4005	0.6717	0.3269
CNN w/ Class Weights	0.7077	0.4232	0.6701	0.3371
Two-Phase	0.7230	0.4476	0.6671	0.3232
Two-Phase w/ Class Weights	0.7090	**0.4873**	0.6543	0.3290
Ensemble (Maj.Vot.)	0.7319	0.4320	**0.6902**	0.3462
Ensemble (Softmax. Avg.)	**0.7371**	0.4397	0.6896	0.3492
Class-Specialized Ensemble	0.7253	0.4785	0.6746	**0.3658**

Table 3.4: Absolute differences in micro and macro scores between the corresponding test scores and the OOD score. Bold values represent the largest differences (lack of robustness) while underlined values represent the smallest differences.

	Histology		Subsite	
Model	Test-OOD Mic	Test-OOD Mac	Test-OOD Mic	Test-OOD Mac
CNN	2.57	5.36	4.16	7.35
CNN w/ Class Weights	<u>1.81</u>	<u>4.28</u>	<u>3.76</u>	8.61
Two-Phase	**3.48**	11.92	**5.59**	12.44
Two-Phase w/ Class Weights	3.12	**13.98**	5.48	**15.83**
Ensemble (Maj.Vot.)	2.30	5.91	4.16	<u>8.58</u>
Ensemble (Softmax. Avg.)	2.43	6.17	4.75	9.06
Class-Specialized Ensemble	3.06	8.10	5.08	11.26

broader insight about models performance on minority classes and their differences in performance with respect to test and OOD datasets.

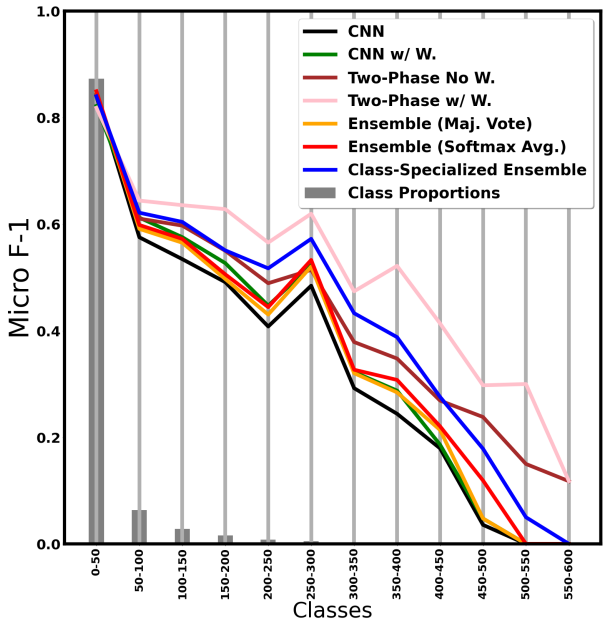
Table 3.5 shows the micro and macro F1 scores obtained when predicting in a subset of the datasets which excludes the top two classes. We found that under this experimental setting, the class-specialized ensemble outperforms other methods in terms of test micro scores and OOD macro scores. Moreover, consistent with previous results, two-phase learning obtains the highest test macro.

3.6 Discussion

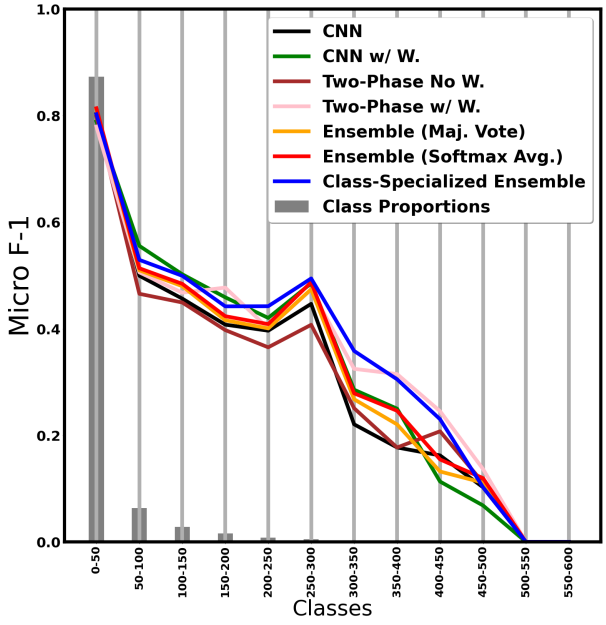
This is the first study which quantifies the performance of the TextCNN for cancer pathology report classification on OOD datasets. In the histology task, we observed drops in performance of up to 3.48% and 13.98% for micro and macro scores, respectively. For the subsite task, the observed values were 5.59% (micro) and 15.83% (macro). These scores demonstrated that the baseline CNN model is not robust under natural distribution shifts when classifying cancer reports.

We note that our methods have different effects in the test and OOD datasets. While we tried to increase the performance of the model on the OOD dataset, we often found that what works well in the OOD dataset also works well (or better) in the test dataset. Thus, there was a consistent gap between the test and OOD dataset. The implication of this result is that improving performance in a closed environment can be highly misleading. That is because improvements in the test dataset do not necessarily correlate with improved performance at deployment time on new data. Other authors have found that larger and diverse datasets can help with robustness [100]. Our models were trained with a large amount of data from different registries across the country, yet we still identified serious drops in performance when predicting on unseen registries.

Through a data profiling analysis, we detected large class distribution differences of the non-common cancer types. Some of the minority classes which appeared in



(a) Test Dataset (registries 1,2,3,4,5,6) combined.



(b) OOD Dataset (Registry 7)

Figure 3.3: Class group performance for the histology task using registry 7 as the OOD dataset. Classes are ordered by frequency which is shown by the gray bars.

low proportions in the training dataset can often appear much more frequently on the unseen registries. We argue that these class distribution differences are one of the sources of performance drop when deploying the model, especially in terms of macro score. Thus, we explored techniques developed in prior studies to deal with imbalanced datasets and improve the performance on the rare cancer types. While some common methods such as ROS and RUS are not adequate for our specific problem, we showed that other techniques such as two-phase learning and class weights are efficient.

Our experiments showed that the CNN with class weight effectively improves the macro score and provides the lowest difference when comparing the test and OOD scores. The downside of this method was that the micro scores were relatively lower. Using two-phase learning pushed the macro scores further without degrading the micro score performance. In fact, this method provided the highest test macro scores among all experiments. As one would expect when combining multiple models, ensemble methods had the highest micro scores across the board and improved the OOD macro. Finally, our novel class-specialized ensemble method, inspired by the mixture-of-experts model, obtained the highest OOD macro score while maintaining competitive test macro and micro scores.

For two-phase learning, the increase in macro scores was highly disproportional when comparing the test and OOD scores; that is, the increase in OOD macro is relatively low. We hypothesize that the performance of this technique is highly dependent on the class distribution of minority class documents. Two-phase learning is able to capture more features that are relevant for the classification of non-majority classes. Because of the large differences in class distribution with respect to the OOD dataset, the OOD macro score did not experience a meaningful improvement. Another possibility is that two-phase learning is overfitting on the minority documents, which leads to low generalization power.

Our methods comparison provided insight that can help other machine learning practitioners deal with extreme class imbalance and issues caused by natural distribution shifts. Based on our results, we provide the following four recommendations: 1) if micro score or accuracy is the main concern independently of the model's performance in minority classes, then using traditional ensembles is appropriate, 2) if classification of minority classes is a priority, we recommend using our class-specialized ensemble implementation, 3) if one is primarily concerned with minimizing differences in performance between the test and OOD datasets, then simply using the CNN with class weights may be sufficient, and 4) if one cannot afford the computational cost of ensemble methods, then two-phase learning is a simple and effective option. However, the high test macro scores can be misleading in terms of the robustness and generalization of the model.

We acknowledge that our study lacks a formal analysis of the statistical significance of our results. The main reason for this design choice is that we used large datasets ($\sim$2 million documents) which made the experiments computationally expensive (a single model trained on six registries and tested on the seventh unseen registry takes $\sim$8hs). We considered statistical tests such as the McNemar's test, which are common in circumstances where training multiple models is expensive. However, this test computes a p-value based on the differences in class predictions between two models, and we observed that for the tasks considered in this research (with large number of classes) the p-value is too close to 0. Nevertheless, the main results of this paper (Table 3.2, 3.3) are the average of seven individual registries which exhibit the same pattern (micro and macro scores for each individual registry are included in Appendix B.1). Therefore, we feel confident of the validity of our results.

3.7 Conclusion

The issue of performance drop at deployment time is complex, and the source of the problem is likely due to multiple factors. In this study, we showed that natural

distribution shifts degrade the performance of a TextCNN for the task of classifying cancer pathology reports. We particularly focused on improving the performance of the model on rare cancer types, increasing the macro scores of the model. We presented a novel version of ensemble learning in which each model learns features that are relevant for a specific group of classes. Our class-specialized ensemble model outperformed other techniques implemented in this paper in terms of OOD macro scores while obtaining competitive test macro and micro scores. Our results helped formulate a series of suggestions for other machine learning practitioners working with highly imbalanced datasets and robustness issues.

Our methods are computationally intensive due to ensembles of models operating on millions of pathology reports from seven different states. Although computation at this scale is feasible for supercomputing centers, the average medical center may not have the same volume of data and therefore require the same computational capabilities. Additionally, the majority of resources are needed for training and not for applying the models. And if the trained models are more robust to begin with, less frequent training and deployments are needed. This is of particular importance for models integrated in clinical workflows where minimization of system downtimes and clinical disruptions is a target.

The results presented in this paper form the basis for future research in model robustness on out-of-distribution clinical text. We showed that the class distribution of rare cancers varies widely across different registries, which translates into diminished performance on minority classes and low macro scores. We hypothesize that distinct vocabulary patterns that are unique to individual registries can also contribute to the disparity in performance. In addition, we showed that ensemble methods outperform single models even when testing on OOD datasets. A natural question that follows is whether a model distilled from the ensemble can maintain the performance advantage over baseline models. This would enable us to obtain similar robustness levels while enjoying the low-resource advantages of a single model.

Table 3.5: Histology Results. Accuracy results when testing in all but the two most frequent classes. The top two classes represent 40.95% of the dataset.

Model	Test Micro	Test Macro	OOD Micro	OOD Macro
CNN	0.6985	0.4090	0.6663	0.3559
CNN w/ Class Weights	0.7118	0.4144	**0.6899**	0.3720
Two-Phase	0.7077	0.4816	0.6590	0.3623
Two-Phase w/ Class Weights	0.6885	**0.5203**	0.6445	0.3797
Ensemble (Maj.Vot.)	0.7077	0.4365	0.6770	0.3779
Ensemble (Softmax. Avg.)	0.7119	0.4453	0.6792	0.3842
Class-Specialized Ensemble	**0.7251**	0.4890	0.6831	**0.3994**

Chapter 4

Using Ensembles and Distillation to Optimize the Deployment of Deep Learning Models for the Classification of Electronic Cancer Pathology Reports

Disclosure Statement

A version of this chapter was originally published in the Journal of the American Medical Informatics Association - Open (JAMIA Open):

Kevin De Angeli, Shang Gao, Andrew Blanchard, Eric B Durbin, Xiao-Cheng Wu, Antoinette Stroup, Jennifer Doherty, Stephen M Schwartz, Charles Wiggins, Linda Coyle, Lynne Penberthy, Georgia Tourassi, Hong-Jun Yoon, Using ensembles and distillation to optimize the deployment of deep learning models for the classification of electronic cancer pathology reports, JAMIA Open, Volume 5, Issue 3, October 2022, ooac075, https://doi.org/10.1093/jamiaopen/ooac075

Authors' contributions - KD: investigation, methodology, software, visualization, writing. SG, AB, HY: conceptualization, investigation, methodology, writing, supervision. ED, XW, AS, JD, SS, CW, LC, LP: data curation, writing. GT: funding acquisition, supervision, wiring

No revisions to this chapter have been made since the original publication.

4.1 Abstract

Objective: We aim to reduce overfitting and model overconfidence by distilling the knowledge of an ensemble of deep learning models into a single model for the classification of cancer pathology reports.

Materials and Methods: We consider the text classification problem that involves five individual tasks. The baseline model consists of a multitask convolutional neural network (MtCNN), and the implemented ensemble (teacher) consists of 1,000 MtCNNs. We performed knowledge transfer by training a single model (student) with soft labels derived through the aggregation of ensemble predictions. We evaluate performance based on accuracy and abstention rates by using softmax thresholding.

Results: The student model outperforms the baseline MtCNN in terms of abstention rates and accuracy, thereby allowing the model to be used with a larger volume of

documents when deployed. The highest boost was observed for subsite and histology, for which the student model classified an additional 1.81% reports for subsite and 3.33% reports for histology.

Discussion: Ensemble predictions provide a useful strategy for quantifying the uncertainty inherent in labeled data and thereby enable the construction of soft labels with estimated probabilities for multiple classes for a given document. Training models with the derived soft labels reduce model confidence in difficult-to-classify documents, thereby leading to a reduction in the number of highly confident wrong predictions.

Conclusions: Ensemble model distillation is a simple tool to reduce model overconfidence in problems with extreme class imbalance and noisy datasets. These methods can facilitate the deployment of deep learning models in high-risk domains with low computational resources where minimizing inference time is required.

4.2 Background and Significance

The American Cancer Society (ACS) estimates 1.9 million new cancer cases will be diagnosed in 2022 [95]. Because cancer is a reportable disease, states rely on population-based registries to maintain a database of cancer pathology reports. Information contained in these documents is key to identifying new reportable cancers and their characteristics across the country.

Electronic pathology reports are stored as unstructured text, and current information extraction relies almost completely on manual processing by trained personnel, which is expensive, time consuming, and prone to error. In the last few years, researchers have reported promising results when training deep learning (DL) models to automate the information-extraction process for pathology reports [36, 38, 2, 35].

Data noise is a serious issue when training DL models for classifying cancer pathology reports. Documents often describe multiple specimens and biopsies that involve different organs analyzed for diagnosis. Manual annotators read the results of

each biopsy and assign a specific cancer site label for the entire report. Although this is a standard way to annotate data, this process leads to a large volume of data noise because large portions of pathology reports focus on the analysis of specimens that are associated with a different site and are not relevant to the context of their ground-truth label. Pathology reports also include information (e.g., names and addresses) that contributes to additional noise. Training neural networks with noisy data can yield models that learn spurious correlations and shortcuts [114, 41].

Label noise presents additional challenges. Annotators are tasked with selecting a class out of hundreds of options. Tasks such as *cancer subsite* and *histology determination* involve the identification of specific classes that often share similarities (e.g., "overlapping lesion of other and unspecified parts of mouth," "mouth not-specified"). Human annotation errors will naturally occur when working with a large number of similar classes and documents in which multiple specimens associated with different classes are reviewed. Additionally, errors can derive from data processing. For example, labels are defined at the cancer/tumor/case (CTC) level. CTC is a data entity that encapsulates all diagnostic, staging, and treatment for a reportable neoplasm. Consequently, pathology reports created during diagnosis are assigned labels based on the CTC—even if these documents analyze specimens associated with different labels.

Extreme class imbalance combined with data noise can lead to serious overfitting issues. The cancer subsite and histology coding tasks consist of 326 and 639 classes, respectively. Some of the classes in these tasks are extremely common. For example, in the subsite task for breast cancer, *upper-outer quadrant of breast* constitutes 8.9% of the data, whereas for the top class in histology, *adenocarcinoma, NOS* corresponds to 21.7% of the data. On the other side of the spectrum, there are cancer types that rarely appear. There are 16 classes in subsite and 127 classes in histology with less than ten instances. When few samples are available during training, DL models tend to memorize specific patterns that do not generalize well [63, 64]. Thus, overfitting is a major challenge when classifying cancer pathology reports.

During classification, it is often desirable to keep only the predictions produced with high confidence. For example, cancer registries from Kentucky, Utah, New Jersey, Washington, and New Mexico (Section 4.4.1) currently require machine learning models to achieve 97% accuracy on a standard test dataset before deployment. The high accuracy imposed on models is necessary to limit potentially costly mistakes in processing healthcare records. Like other high-risk fields such as self-driving cars and medical diagnosis, the goal is to minimize error and maximize coverage. Previous investigators have referred to this research area as *selective classification* [40, 116, 26], *prediction with a reject option* [15], and *model abstention* [102].

The literature on selective classification for traditional machine learning is extensive, with one of the first papers published in the 1950s [16, 15, 47]. However, few papers have discussed model abstention in the context of DL [40]. Previous work in this area focused mostly on deriving an optimal softmax threshold given a certain cost/risk constraint. Because softmax layers are common in DL architectures, the softmax thresholding framework is a simple and convenient rejection rule that can be applied to most models, including pretrained networks.

Model overconfidence deteriorates the efficiency of abstention mechanisms that are based on softmax thresholding. Previous researchers have hypothesized about the source of model overconfidence. They pointed out that using one-hot (hard) labels during training leads to overconfidence because it encourages the model to produce predictions with 100% confidence [103, 99]. From an overfitting perspective, overconfidence is the result of overfitting the negative log-likelihood loss, which encourages the model to produce outputs with low entropy [28]. Additionally, hard labels do not allow for degrees of truth, and assigning 100% confidence to noisy documents that mention specimens associated with numerous classes may not accurately represent the input. Model overconfidence leads to a larger volume of highly confident but wrong predictions, and that has a direct negative impact in abstention mechanisms that are based on softmax thresholding.

One could train models with soft labels to reduce overconfidence and provide accurate input representation. However, creating accurate soft labels imposes several challenges. Manual creation of soft labels would involve assigning probabilities to each class, which can be subject to the annotator's interpretation and is often not feasible owing to time constraints.

A simple way to derive soft labels is label smoothing. Given some constant, α, with $\alpha \in (0, 1)$, this method assigns a $1 - \alpha$ to the ground-truth class and splits α equally among the rest of the classes (i.e., 0 becomes $\frac{\alpha}{K-1}$, where K is the number of classes). However, label smoothing does not introduce any information about the underlying class hierarchies or knowledge related to the quality of the input. Additionally, although label smoothing can potentially reduce model overconfidence, that does not imply improved abstention performance when choosing a softmax threshold. In fact, we hypothesize that this method is likely to deteriorate abstention rates because it is prone to shorten the distribution of model predictions (i.e., from $[0, 1]$ to $[0, \alpha]$), thereby making it more difficult to find a softmax threshold that separates between right and wrong predictions.

Ensemble learning is a simple solution to reduce overfitting. The benefits of ensembles in the context of overfitting have been quantified extensively by previous researchers [12, 17]. Highly parallelizable ensemble methods are especially attractive because they can be implemented and tested quickly. However, cancer registries across the country have limited computing resources, thereby making ensemble methods unfeasible in the deployment/inference phase. Ensembles also require additional testing time because a single prediction often requires the output of every single model in the ensemble. Thus, ensembles remain a computationally expensive technique, and that limits their utility and prevents their deployment in numerous environments.

Model distillation is a promising, low-resource solution that leverages the benefits of ensembles without using hard labels. The idea behind model distillation is to transfer and compress the knowledge of a larger model (teacher) into a smaller (student) network. In the context of ensemble model distillation, researchers have

attempted to transfer the combined knowledge of a group of models into a single, low-resource network [50]. Thus, they aim to maintain the high performance of the ensemble while enjoying the computational flexibility of a single model. One intuitive way to perform ensemble model distillation is to train a student model with soft labels obtained through the aggregation of the ensemble predictions. That is, training the student model using ensemble predictions instead of the annotated labels. This method permits automatic derivation of soft labels that contain information about the variability within the ensemble and avoids the use of hard labels.

Previous work explored model abstention for ensemble learning [107, 31]. Existing work focused on deriving rejection boundaries based on the statistics of the ensemble predictions. The downside of these studies is that they focused on simple binary problems, and more complex classification tasks, such as the ones we have described for electronic pathology report information extraction, are not considered. To the best of our knowledge, the effect of ensemble model distillation in the context of selective classification remains an understudied research area.

4.3 Objective

The objective of this study was to investigate the feasibility of ensemble model distillation as a low-resource alternative for the deployment of DL models for cancer pathology report classification. We hypothesized that ensemble model distillation would allow us to enjoy both the overfitting reduction benefits of the ensemble and a reduction of model overconfidence caused by hard labels. Performance was quantified as accuracy and abstention rates by using softmax thresholding tuned to yield 97% accuracy. We provided additional analysis of the benefits of ensemble model distillation on data and label noise. These findings may provide solutions to other machine learning researchers working in high-risk domains with limited computational resources where low-error rates are required.

4.4 Materials and Methods

4.4.1 Dataset

Classifying electronic cancer pathology reports consists of five individual tasks. That is, each pathology report must be labeled with a specific site, subsite, laterality, histology, and behavior. The number of classes in each task is shown in Table 4.1.

For this study, we used datasets from the Louisiana Tumor Registry (LTR), Kentucky Cancer Registry (KCR), Utah Cancer Registry (UCR), New Jersey State Cancer Registry (NJSCR), Seattle Cancer Registry (SCR), and New Mexico Tumor Registry (NMTR). The sizes of the six individual datasets are listed in Table 4.2. To satisfy de-identification requirements, we used integers instead of the actual names to represent each of the datasets.

The dataset exhibits extreme class imbalance. For example, the top two histology classes (i.e., *adenocarcinoma, NOS* and *ductal carcinoma*) constitute 41.0% of the dataset. In the subsite task, the top two classes (i.e., *upper-outer quadrant of breast* and *prostate gland*) correspond to 17.3% of the data. It is not uncommon to see fewer than ten instances for rare cancer types.

4.4.2 Experimental Setup

We aimed to develop experiments that would simulate real-world deployment. To achieve this, we implemented a leave-one-registry-out approach in which we first combined five registries for training and validation. Once the model was trained, we deployed the model by predicting the left-out (out-of-distribution) dataset. We expected the left-out dataset to contain natural variations not observed during training.

This experimental setup simulates real-world deployment and allows us to evaluate the generalizability of the classifier. In a previous study [21], we quantified the performance disparity between a test dataset, which was taken from the same

distribution as the training and validation data, and a completely unseen (left-out) registry. In that study, we observed that R4 exhibited the largest performance drop. Therefore, for this study, we used R4 as our left-out dataset. Leaving out R4 and combining the rest of the registries leads to a total of 1,525,545 pathology reports for training and validation and 441,732 (size of R4) documents for testing.

DL models were trained using early stopping as a standard overfitting prevention practice. We set the patience parameter to 5, so if the validation loss does not decrease for 5 consecutive epochs, then training stops, and the model recovers the best set of weights. Figure 4.1 shows an overview of our training pipeline.

4.4.3 Multitask TextCNN

The base model for our experiments is the TextCNN. We use this specific DL model because: 1) in our previous publication involving pathology report classification, we showed that the TextCNN model performs about the same or better than transformer based models [35], 2) we can train ensembles of MtCNNs in parallel since it is a computationally cheap model (in terms of memory and speed), and 3) the low computational requirements of the MtCNN makes it accessible to cancer registries across the country, allowing for rapid deployment.

We used a specific version of the TextCNN known as the multitask CNN (MtCNN), which has been implemented in numerous previous studies involving cancer pathology report classification [20, 1, 2, 36, 21, 8]. The MtCNN simultaneously outputs predictions for all five tasks.. The input to the MtCNN first passes through an embedding layer, in which each word token is mapped to a 300-dimensional word-embedding vector. The resulting matrix passes through three parallel convolutional layers with filter sizes of 3, 4, and 5 consecutive words; each of the convolutional layers contains 300 filters. The output of the convolutional layers is then concatenated and sent to a global max pooling over time layer. Finally, the resulting vector goes through

Table 4.1: Number of classes in each task.

Task	Site	Subsite	Laterality	Histology	Behavior
Classes	70	326	7	639	4

Table 4.2: Size of individual registries.

Registry	R1	R2	R3	R4	R5	R6
e-Path Reports	85,789	577,094	137,135	441,732	360,375	365,152

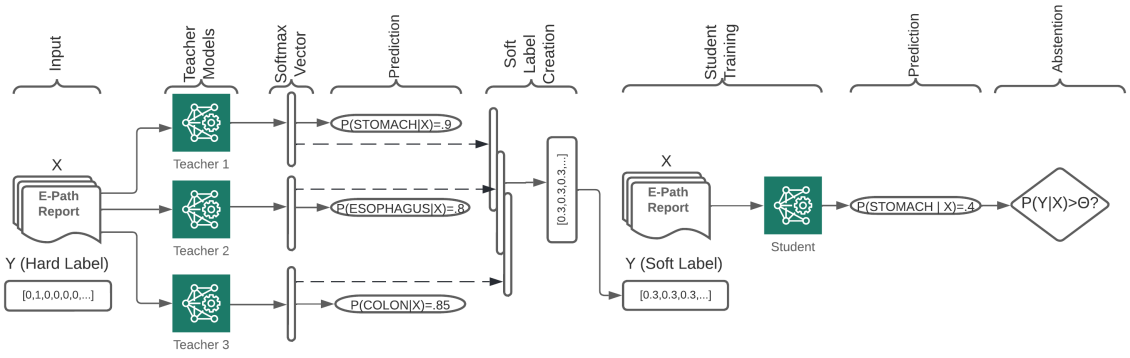


Figure 4.1: Overview of our training pipeline with a hypothetical example in which three different models classify a pathology report as stomach, esophagus, and colon. Our actual implementation consists of 1,000 teacher models.

five parallel dense layers (one for each of the five classification tasks), and the five predictions are produced.

4.4.4 Ensemble Learning

Ensemble learning is a machine learning method that utilizes multiple models to obtain better predictions. Several ensemble learning algorithms are available, including bootstrap aggregation (bagging) [10], boosting [32], and a mixture of experts [53].

Previous studies have noted that training deep neural networks with more data led to better performance, and bagging can hurt performance because models only see ∼63% of the data [58, 60]. These studies showed that using the entire training dataset is more efficient than bagging approaches, which sample the training dataset with replacement. Therefore, in this study, we trained 1,000 MtCNN models using the entire training dataset but different random initialization seeds. Our method is highly parallelizable and simple to implement.

The ensemble inference was derived by normalizing the summation of the outputs from the multiple models: $D(\mathbf{x}) = \sum_{i=1}^{T} d_i(\mathbf{x})$, where $\mathbf{x}$ is some document input, d_i is the prediction vector for model i in the ensemble, and T is ensemble size ($T = 1,000$). Then we applied the softmax function to the ensemble output D to infer the ensemble decision.

Notably, to obtain a prediction for a given document, x, one must first use all T models for the prediction and then aggregate their predictions to obtain the final output. This can be time consuming and computationally demanding. Therefore, ensemble learning is often a nonviable method for several real-world applications.

4.4.5 Distillation

In a typical supervised learning setting, a neural network is trained with data in the form (x, y), where in our case, x is a pathology report, and y is the associated label

depending on the task (e.g., cancer site, subsite). This type of learning uses *hard labels*, which means y is a one-hot encoded vector that contains binary information: either y belongs to a certain class or it does not (i.e., [0,0,1]). Alternatively, one could train a model with data in the form of $(x, \vec{y})$, where $\vec{y}$ could be interpreted as the probability that x belongs to each class (i.e., [0.1,0.1,0.8]). This paradigm is known as *soft labeling*.

To distill the knowledge of the ensemble, we trained a single MtCNN using the ensemble predictions (aggregated vectors) as the class labels for the respective documents. Thus, the student model was trained with the same training documents, X, but with the soft labels derived from the ensemble instead of the original hard labels. Notably, our distillation implementation uses the categorical cross-entropy loss function: $CE(\mathbf{y}, \hat{\mathbf{y}}) = \sum_{k=1}^{K} y * \log \hat{y}$ where y is the soft label, $\hat{y}$ if the softmax prediction vector, and K is the number of classes. The following list describes the steps we took to distill the knowledge of the ensemble model:

- Train 1,000 MtCNN models.
- Extract the softmax prediction vector of the 1,000 MtCNNs for each document in the training set.
- Aggregate the prediction vectors by summing and normalizing. This will form a new set of $\vec{y}$ labels, where the labels are vectors (soft labels).
- Train a single MtCNN using the original pathology reports, X, but with the soft labels $\vec{y}$ instead of the original ground-truth labels.

By using the soft labels of the ensemble as the truth labels, we hypothesized that the model would identify the same features that the ensemble used to produce the classification probabilities.

4.4.6 Selective Classification with Softmax Thresholding

In numerous application such as ours, low error rates are tolerated. In these problems, abstention mechanisms are implemented so that only models with highly confident predictions are used. In this study, we seek to maintain an error rate below 3% based on the consensus of state cancer registry partners.

To accomplish this, we implemented a straightforward version of model abstention that can be used with any trained model. Our goal was to satisfy the requirements by finding a softmax threshold that will yield 97% accuracy. The specific procedure is described as follows:

- Create a list of potential thresholds from 0 to 1 with a step size of 0.001 (i.e., [0.001, 0.002, ..., 0.999]).
- Using the validation dataset, find the smallest threshold so that when predictions are filtered out with a softmax below this threshold, the accuracy is $\geq$0.97.
- Use this threshold as a rejection rule. At test time, discard predictions that have a confidence below the selected threshold.
- Compute the percentage of documents that remain in the dataset after abstention and measure accuracy of that subset of the data.

Although the abstention rate refers to the proportion of documents left out during testing, in this study, we report results in terms of *retention proportions*: $RP = \frac{\hat{X}}{X}$ where $\hat{X}$ is the number of predictions made with confidence higher than the rejection threshold, and X is the size of the dataset. Also note that RP=(100%-abstention rate). Ideally, one will retain a large percentage of the data (high coverage) and obtain an accuracy of $\sim$0.97 on these documents.

4.4.7 Statistical Significance

We wanted to analyze if the student model performance was statistically better than a standard MtCNN model trained with hard labels. To account for natural variation, we used the 1,000 MtCNN models that we trained for the ensemble to derive 95% confidence intervals.

4.4.8 Model Overconfidence

We analyzed model overconfidence by focusing on wrong predictions. We first determined the distribution of prediction confidence (softmaxes) for all wrong predictions. In this analysis, we compared the baseline MtCNN with the student model to understand the extent to which the student model can reduce the number of wrong predictions with high confidence.

In our second analysis of model overconfidence, we analyzed the number of highly confident but wrong predictions for different ensemble sizes. In particular, we quantified the number of wrong predictions made with a confidence of >0.97. Our choice of a 0.97 threshold is based on the consensus of the cancer registries' error tolerance. This analysis was performed by averaging multiple samples obtained by bootstrapping MtCNNs from the pool of 1,000 models and then creating ensembles of the respective sizes.

4.4.9 Data and Label Noise Analysis

We hypothesized that data and label noise were both issues when training models for classifying cancer pathology reports. We investigated the effects of model distillation on data and label noise by examining the distribution of ensemble predictions and inspecting individual reports.

We first verified that the ensemble predictions could effectively fix noisy labels. To that end, we inspected documents in which a wrong prediction was made with 100% agreement between the 1,000 models.

Our analysis of data noise was based on the assumption that the amount of noise contained in the input will manifest itself in the distribution of votes across the ensemble. Traditional majority voting uses Equation 4.1 to infer the ensemble predictions [25], where $d_{t,j} = 1$ if model t of the ensemble T predicts class j and $d_{t,j} = 0$ otherwise. Here, we were interested in cases where the number of votes were split almost equally between two or three classes, meaning that there is not a clear winner. For cases in which half of the ensemble predicts class y_1, and the other half predicts y_2, we expected to observe a report that contained lexical patterns common to both classes.

$$\max_{1 \leq j \leq k} \sum_{t=1}^{T} d_{t,j} \tag{4.1}$$

The distribution of ensemble votes have a direct impact on the derived soft labels. For example, given a pathology report about the gum, half of the ensemble may predict *upper gum*, and the other half may predict *lower gum*. Naturally, the resulting soft label for such input is expected to represent such a division (i.e., [0.5,0.5,0, ...] = [*lower gum*, *upper gum*, ...]). Thus, we wanted to visualize what aspect of the input drives this type of predictive pattern and how that relates to the derived soft label.

4.5 Results

4.5.1 Selective Classification

Table 4.3 presents the results in terms of retention proportions (i.e., the percentage of documents that would be classified when deploying the models). As expected, when combining predictions from multiple classifiers, the ensemble model yielded the best overall performance. We also observed that the student model outperformed the baseline MtCNN for all tasks except for behavior (the task containing only 4 classes). We note that the major boost in performance was observed for subsite

and histology, for which the absolute increase in coverage was ∼1.81% and ∼3.33%, respectively. These are the two most difficult tasks because they are characterized by a large number of classes and severe class imbalance.

Using the percentages from Table 4.3 and the size of the test dataset (R4: 441,732), one can translate these values into the number of pathology reports. For example, in subsite, the MtCNN and the student model classify 152,485 and 160,481 of the 441,732 reports, respectively. This indicates that the student model can be used to classify an additional 7,996 documents. A similar calculation for the histology task showed that the student model can predict an additional 14,710 pathology reports (i.e., $120,151 - 105,441 = 14,710$).

Table 4.4 lists the accuracy scores obtained among the non-abstained documents. Although the individual softmax thresholds were tuned with the validation dataset to yield a 97% accuracy, we still observed accuracy below our target performance in all tasks except for behavior. This type of drop was expected owing to the natural distribution shifts when applying models to new registries, and in practice, this drop can be easily mitigated by using a higher target accuracy. Notably, the baseline MtCNN exhibits performance well below the 97% target (see subsite and histology in Table 4.4). The student model alleviates the performance drop, but it still fails to reach the target.

4.5.2 Wrong Prediction Confidence

When implementing a softmax–threshold abstention mechanism, minimizing the number of highly confident but wrong predictions is essential. That is because having too many highly confident wrong predictions pushes the softmax threshold toward 1, thereby leading to higher abstention percentages (i.e., less coverage). We compared the prediction confidence distributions of the wrong predictions for histology (Figure 4.2) and subsite (Figure 4.3). For both tasks, the student model generated fewer wrong predictions with softmaxes above 90%, and this difference

Table 4.3: Retention proportions results. The numbers shown represent the percentage of document remaining after abstention (higher percentages means more coverage). Intervals represent 95% confidence intervals.

Model	Site	Subsite	Laterality	Histology	Behavior
MtCNN	90.62	34.52	87.83	23.87	99.46
	(90.61,90.63)	(34.48,34.56)	(87.82,87.85)	(23.77,23.97)	(99.45,99.46)
Student	91.10	36.33	88.49	27.20	99.98
Ensemble	92.17	39.00	89.60	34.16	99.42

Table 4.4: Accuracy results. Intervals represent 95% bootstrap confidence intervals.

Model	Site	Subsite	Laterality	Histology	Behavior
MtCNN	96.06	94.43	96.05	95.12	97.69
	(96.06,96.07)	(94.42,94.45)	(96.04,96.05)	(95.10,95.14)	(97.69,97.70)
Student	96.27	94.82	96.19	95.84	97.60
Ensemble	96.19	94.55	96.10	95.78	98.00

was particularly noticeable in the histology task. These plots illustrate the effects of training models with hard and soft labels and their impact on selective classification scores.

Figure 4.4 shows the effects of ensemble sizes on the number of wrong predictions made with a confidence >0.97. We observed that the number of wrong predictions decreased as the ensemble sizes increased, but this trend converges after an ensemble size of approximately 200 models. We also note that most of the improvement occurs with the addition of the first few models. That is, the ensembles with between 2 and 10 models exhibit the highest performance boost.

4.5.3 Data and Label Noise Analysis

We analyzed the effects of ensemble model distillation in terms of label noise by manually reading pathology reports that were classified incorrectly with 100% ensemble agreement. In every document we inspected, we found that those documents were mistakenly annotated. As an example, we de-identified one of the documents that was annotated as *stomach* but was classified as *esophagus* by every model (Figure 4.5). This was a case in which a pathology report discussed the biopsy of only one specimen, and lexical patterns tend to be consistent with the predicted class. Notably, in cases in which there is a 100% agreement within the ensemble, the associated softmax approximates the hard label (e.g., the predicted class contains a value close to 1).

We also analyzed data noise by examining individual pathology reports based on particular ensemble prediction patterns. Intuitively, we expected that when the ensemble votes were split across multiple sites, the pathology report would discuss specimens and biopsies that were associated with each of the predicted sites. As an example, we de-identified a pathology report in which the ensemble votes were split into three equivalent-size groups for the predictions of *stomach*, *esophagus*, and *colon* (Figure 4.6). This is an example where the input contained lexical patterns related

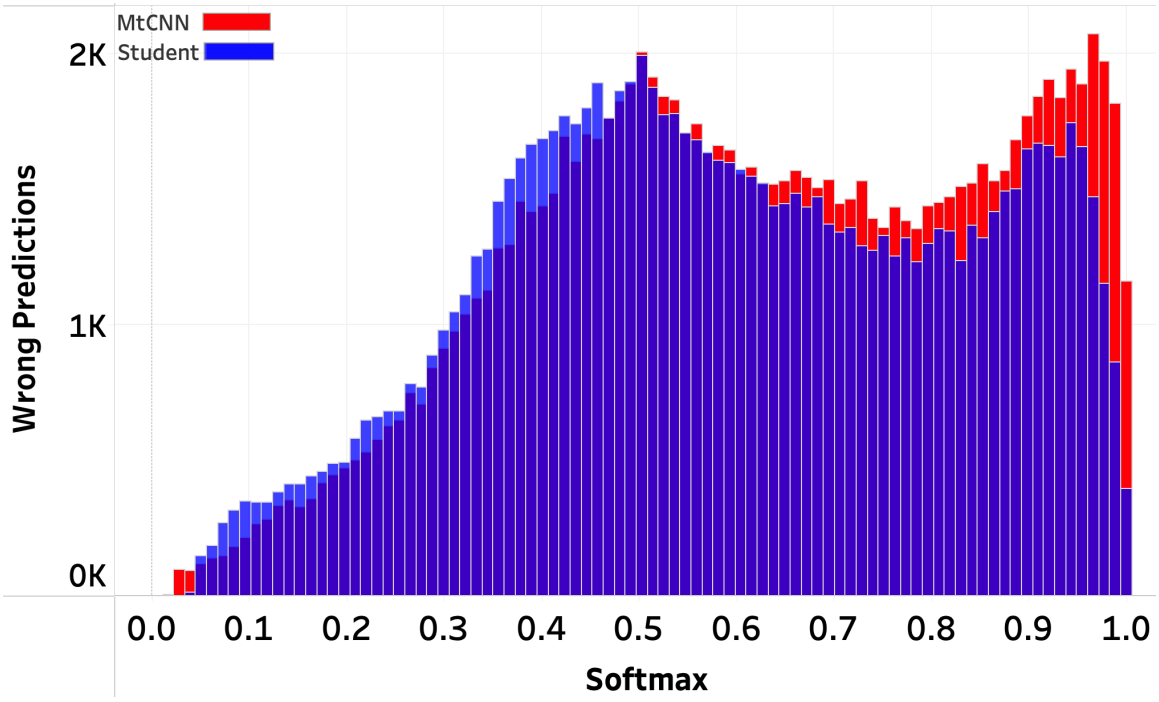


Figure 4.2: Histology Task. Distribution of softmaxes for the wrong predictions.

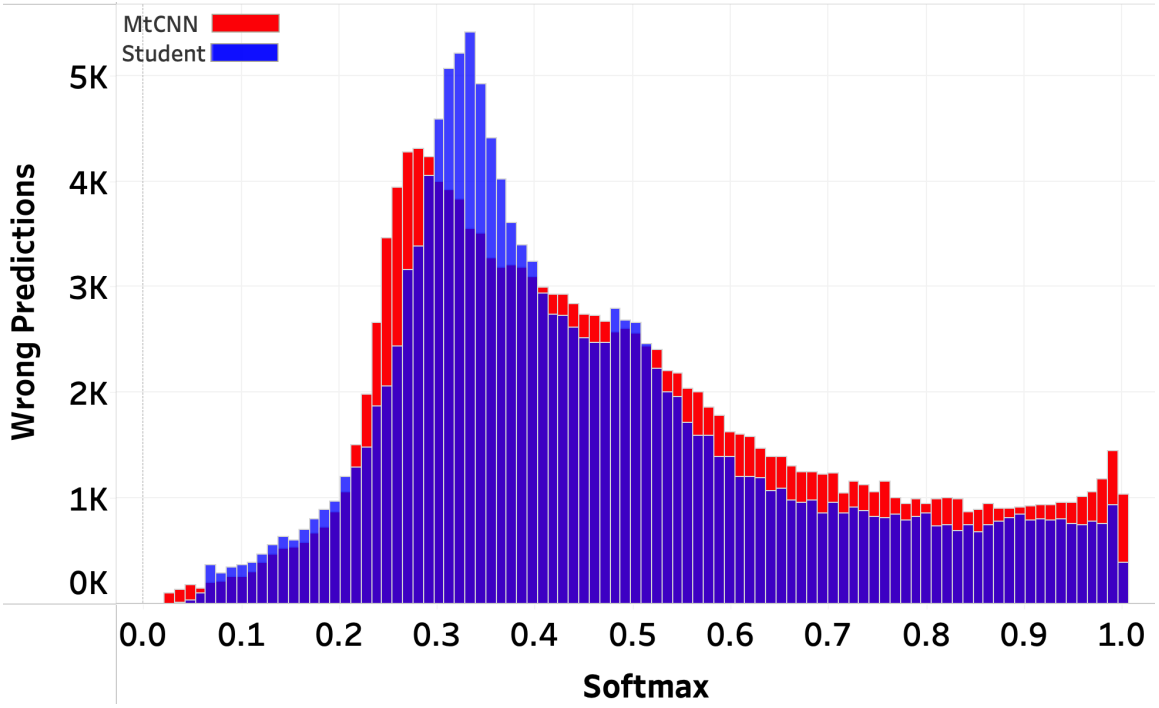


Figure 4.3: Subsite Task. Distribution of softmaxes for the wrong predictions.

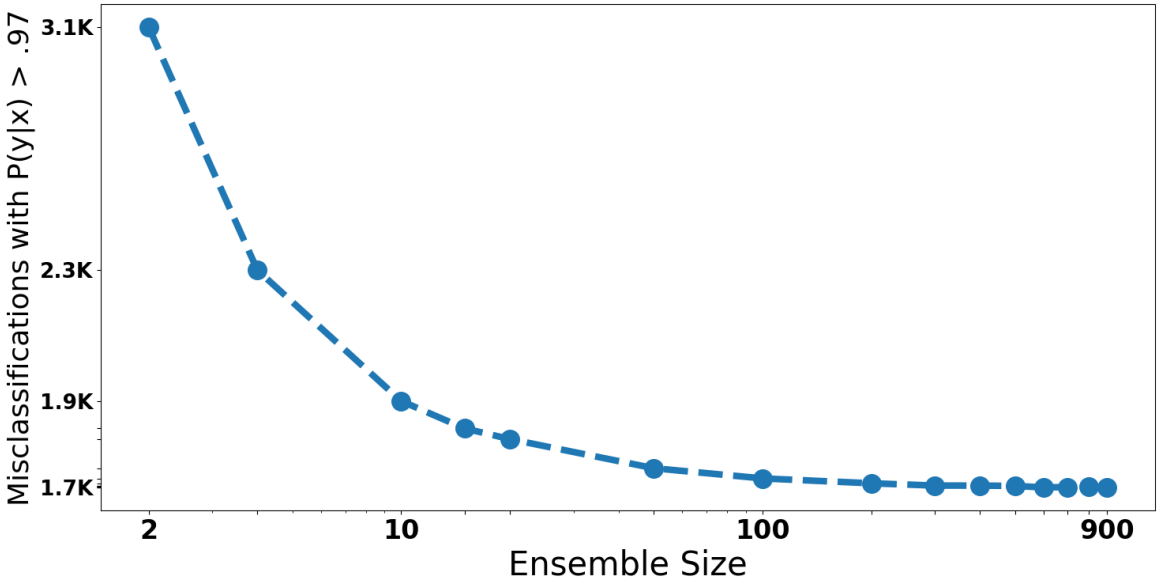


Figure 4.4: Wrong histology predictions made with confidence >0.97.

Ensemble Predictions:
Group 0 Votes: 1000 -- Prediction Class: C15 - ESOPHAGUS
Primary Site (annotated label): C16 - STOMACH

Report

Formal Dx: Distal esophagus multiple biopsies : invasive adenocarcinoma . features consistent with background chronic reflux esophagitis.

Gross Pathology: Single specimen, biopsies of distal esophagus, clinically cancer. This consists of multiple pale to pink mucosal biopsies, aggregate floattoken x floattoken cm, double sponged cassette [...].

Microscopic Description: Multiple biopsies of distal esophagus confirm the clinical impression of malignancy. This is an invasive adenocarcinoma, poorly differentiated. Although in many areas there is distinct gland lumen formation with some evidence of intracytoplasmic and intraluminal mucin substance, the deeper portion is more poorly differentiated showing cohesive malignant glandular epithelial cell infiltrate in a solid pattern. In the malignant glandular lumina, there are apoptotic malignant epithelial cells. While the squamous mucosa shows ulceration, where it is intact, it is hyperplastic showing vascular tufting consistent with background chronic reflux esophagitis. However, there is no definite evidence of barretts esophagus. The tumor undermines benign esophageal mucosa [...].

Nature of Specimens: 'procedure : . specimen (s) : biopsy , cancer of distal esophagus.

Figure 4.5: Incorrectly annotated pathology that was fixed during the distillation process. Some sentences were removed to conserve privacy.

to all three classes (see sections *path comments* and *nature of specimens* in Figure 4.6), and this misled the models' predictions.

4.6 Discussion

This is the first study to quantify the abstention performance of ensemble model distillation by using softmax thresholding for the classification of electronic cancer pathology reports. Our results indicate that soft labels derived through ensemble model distillation can effectively improve abstention performance. Therefore, in real-world settings where inference resources are limited, our proposed distillation method yielded considerable improvements while maintaining the computational cost of a single model.

We measured performance by calculating the percentage of documents kept in the dataset after abstention and the accuracy of those documents. We observed that the student model outperformed the baseline MtCNN under both metrics and for four of the five tasks. The most substantial improvements occurred in the two most difficult tasks (subsite and histology), which contained the highest number of classes (326 for subsite and 639 for histology). Thus, we showed that ensemble model distillation can help increase the DL models' coverage when deploying models in the real world, thereby allowing practitioners to use AI systems in a larger volume of documents and reduce the cost and time associated with manual annotation.

Although the performance increase obtained with the student model may seem small, these improvement become substantial at a large scale. For the subsite and histology tasks, the student model increases coverage by $\sim$1.81% and $\sim$3.33%, respectively. These values represent 7,996 and 14,710 pathology reports using our current test dataset. However, given the ACS predicts that 1.9 million new cancer cases could be diagnosed in 2022 [95], and it takes approximately 2 minutes to read each report, these percentages could represent 10,000+ pathology reports and 1,000s of hours of manual annotation saved.

Analyzing the distribution of ensemble votes and their resulting soft labels yielded additional insight into data noise in pathology reports. During cancer diagnosis, it is common to analyze multiple samples from tissues outside of the diseased organ. The result is multiple pathology reports from numerous biopsies. We argue that data noise is a serious issue in these reports because they contain lexical patterns commonly associated with classes outside of their assigned ground-truth label. When analyzing the ensemble vote distribution, we found that these particularly confusing pathology reports can be identified by inspecting documents in which the ensemble votes are split into equivalent-sized groups (Figure 4.6). In these cases, assigning a soft label allows for degrees of truth to indicate that the word patterns and context of the input do not belong exclusively to one class but to a group of classes. Conversely, when inspecting documents with high ensemble agreement (i.e., cases in which soft labels approximate hard labels), we found clear pathology reports that focus on one specific class (Figure 4.5). The importance of this ensemble agreement and soft-label relationship is that we can guide the student model to make highly confident predictions for documents with little noise while lowering confidence for documents that involve multiple sites.

An additional benefit of ensemble model distillation is alleviating label noise. When predicting with 1,000 models, one can naturally expect that a 100% agreement in predictions is likely to correspond with correct predictions. When inspecting pathology reports that were wrongly classified with 100% ensemble agreement, we found that these reports were actually incorrectly annotated (Figure 4.5). Consequently, ensemble model distillation is a useful tool to reduce label noise issues and is particularly beneficial in domains with a high number of highly related classes. This result is consistent with previous work that used ensembles for label correction [122, 92].

Notably, developing efficient abstention mechanisms is still an open area of research. In this study, we implemented softmax thresholding, which is a common abstention mechanism compatible with most DL models. We used the validation

dataset to identify a threshold that would yield a 97% accuracy. However, we observed that when using this threshold in the holdout dataset, the resulting accuracy can be as low as 94.42% (see subsite in Table 4.4). We hypothesize that this performance disparity is amplified because of our leave-one-registry-out experimental setup, and it may be a sign of further overfitting. The discrepancy observed between the validation and test dataset highlights the need for future research that focuses on more efficient abstention techniques.

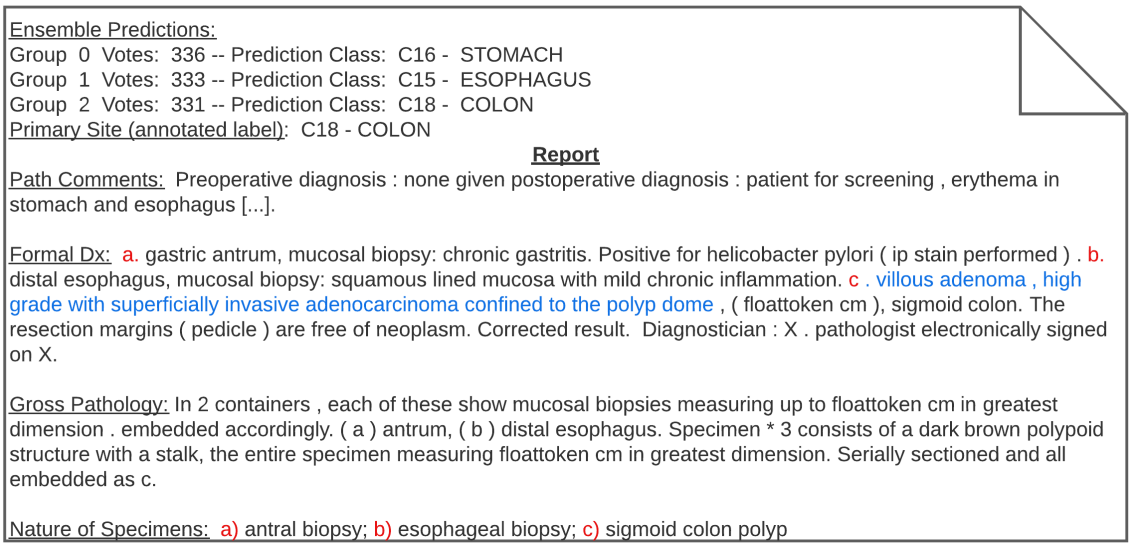
Our results yielded insight into the effect of ensemble size on model overconfidence. In this study, we implemented an ensemble of 1,000 models. However, our results indicated that even an ensemble with between 4 and 100 models can reduce the number of wrong predictions made with high confidence. This highlights the accessibility of our methods because DL practitioners who have limited computational resources can still benefit from the overconfidence reduction obtained through ensemble model distillation.

The methods presented in this study are simple and highly parallelizable. The literature on model ensembles and distillation is extensive and includes a large variety of implementations. One can easily experiment by combining distillation with a particular ensemble technique such as boosting [84] or bagging [11]. In this paper, we focused on a simple implementation that involves training multiple versions of our current classifier with different initializations. Previous work has shown that the training models with distinct random seeds can infuse diversity in terms of the inductive biases learned by the networks [19]. We hope that the simplicity of our method can provide a convenient solution for other DL practitioners who intend to reduce the number of highly confident but wrong predictions for the deployment of DL models in high-risk domains.

4.7 Conclusion

Extreme class imbalance together with data and label noise leads to serious overfitting issues when training DL models for the classification of electronic cancer pathology reports. Ensemble methods are a simple solution to alleviate these issues, but these methods are computationally expensive and unsuitable for deployment by cancer registries across the country. Thus, this study quantified the use of ensemble model distillation as a low-resource alternative. The soft labels derived through model aggregation contain information about the variability in ensemble predictions. We showed that training a student model with the derived soft labels can reduce the number of highly confident but wrong predictions, thereby leading to a boost in abstention rates when using softmax thresholding. The implemented methods provide a simple and highly parallelizable solution for researchers working in high-risk domains. Our ensemble model distillation code is available on Github.*

*https://github.com/kevindeangeli/EnsembleDistillation/



Ensemble Predictions:
Group 0 Votes: 336 -- Prediction Class: C16 - STOMACH
Group 1 Votes: 333 -- Prediction Class: C15 - ESOPHAGUS
Group 2 Votes: 331 -- Prediction Class: C18 - COLON
Primary Site (annotated label): C18 - COLON

Report

Path Comments: Preoperative diagnosis : none given postoperative diagnosis : patient for screening , erythema in stomach and esophagus [...].

Formal Dx: a. gastric antrum, mucosal biopsy: chronic gastritis. Positive for helicobacter pylori (ip stain performed) . b. distal esophagus, mucosal biopsy: squamous lined mucosa with mild chronic inflammation. c . villous adenoma , high grade with superficially invasive adenocarcinoma confined to the polyp dome , (floattoken cm), sigmoid colon. The resection margins (pedicle) are free of neoplasm. Corrected result. Diagnostician : X . pathologist electronically signed on X.

Gross Pathology: In 2 containers , each of these show mucosal biopsies measuring up to floattoken cm in greatest dimension . embedded accordingly. (a) antrum, (b) distal esophagus. Specimen * 3 consists of a dark brown polypoid structure with a stalk, the entire specimen measuring floattoken cm in greatest dimension. Serially sectioned and all embedded as c.

Nature of Specimens: a) antral biopsy; b) esophageal biopsy; c) sigmoid colon polyp

Figure 4.6: Pathology report in which the ensemble prediction votes were split into three equivalent groups. This report includes results of three analyzed specimens related to the stomach, esophagus, and colon. Some sentences were removed to ensure privacy.

Chapter 5

Conclusion

5.1 Summary of Findings

Overfitting is a common issue in domains with high class imbalance because ML models tend to memorize samples from under-represented classes, leading to poor generalization [65]. This is a serious issue in cancer pathology report classification where, for example, two of the 639 histology classes compose $\sim 40\%$ of the dataset.

One way to alleviate overfitting of the minority classes is by obtaining more documents (training samples) from these rare cancer types. However, as previously mentioned, obtaining labels for cancer pathology reports is a costly and time-consuming task. In addition, annotators typically select a random subset of the pool of unlabeled data to assign labels, and it is impossible to tell if this subset contains any minority classes at all. As more and more data is annotated, the class imbalance remains relatively constant even if the number of documents belonging to rare classes increase. Thus, in Chapter 1 we introduced the following research question:

> *How can we intelligently select a distinct subset of a larger pool of unlabeled data so that manual annotator efforts are maximized? In other words, can we focus annotation efforts on pathology reports that will maximize the information gain and reduce overfitting? And what are the implications of selecting a particular subset of data in terms of class imbalance and rare classes?*

In Chapter 2 we implemented 11 active learning algorithms and showed that these algorithms can help us identify informative documents from a large pool of unlabeled data. Our experiments showed that by using active learning techniques, we can achieve peak performance with only half of the available data. Furthermore, one of the most valuable finding of this study is that active learning can be highly valuable in domains with extreme class imbalance since these algorithms effectively sample documents from minority classes, increasing their representation in the dataset and reducing class imbalance.

Overall, Chapter 2 presents a potential set of tools that trained annotators could use to sort and prioritize which documents should be labelled first. The significance of our results is that we can potentially reduce expensive annotators costs by integrating AL techniques into the data annotation workflow. Based on our study, we formulated a comparison table of the 11 algorithms in terms of micro/macro scores obtained and the computational cost of each technique. We hope our table can serve as a guide to other ML practitioners working on problems with large amount of labels and extreme class imbalance.

One potential drawback is that some of the documents selected by the active learning algorithms may be considered *informative* when in reality they make contain a lot of noise. In other words, a DL model may find documents that are hard to classify not because it does not know enough about that class, but because the information contained in the pathology report is not representative of any class. This potential problem is hard to quantify since it involves manually reading documents filled with technical, domain-specific terms. Nevertheless, we expect that even if AL techniques sample some noisy documents, the proportion is likely so be small since the accuracy and macro scores are still relatively high.

We note that we observed a boost in performance with a relatively small dataset. Late iterations of AL have a small subset of the data available to select documents, and the pool of pathology reports may not be highly informative. Thus, most of the boost in performance occurs during the early iterations of the experiments. This would be different if AL techniques were deployed in the real world because new unlabeled cancer pathology reports are stored every day, and the pool of unlabeled data would continue to increase instead of shrinking.

While active learning showed promising results as a potential tool to reduce overfitting by sampling documents from minority documents, we were still missing a proper way to quantify the differences in performance between lab testing and real-world deployment as reported by NCI. Thus, the second research question and topic that we proposed in Chapter 1 was:

> Can we design experiments that are more reflective of the real-world deployment so that we can quantify the performance drop reported by registries? We want to analyze how the distribution of rare cancer classes varies between in-lab training and real world deployment. We consider the use of ensemble learning to tackle overfitting and mixture-of-experts methods to boost rare classes performance.

In chapter 3, we focused on quantifying the performance of the MtCNN in out-of-distribution datasets. To that end, we developed experiments following a leave-one-out-approach. Previous studies involving the classification of cancer pathology reports were developed by combining the individual datasets from all six registries into one main dataset and performing a 80/10/10 split after random shuffle. Instead of combining all the individual datasets, here we combined five of them that we used for training and validation, and use the left-out dataset as an out-of-distribution dataset for testing. We then analyzed the differences between the training and testing datasets and found that different prevalence of cancers leads to large variations of class distribution between the training/validation and testing datasets. The implications of this finding is that there exists cancer types which are common within a region of the country and constitute a large percentage of the data, but are considered rare in other areas. Deep learning models are known to be biased towards the top classes since they drive the loss function. Hence, the distribution of classes will affect the features it learns and which classes receive more attention. The distribution of minority classes differs greatly across registries leading to a large underperformance of the model in OOD datasets in terms of macro scores. To alleviate this issue, we experimented with traditional class imbalance techniques to reduce overfitting and boost macro scores. We compared the performance of techniques from two major groups: data-level techniques and algorithm-level methods. One of the main takeaways of this chapter is the efficient generalization power obtained by ensemble methods. However, ensemble models are computationally expensive, which limit their usability when

deployed in the real world by cancer registries. This lead us to our third research question presented in Chapter 1:

> Can we distill the knowledge of an ensemble into a low-resource model that can be used by cancer registries? In addition to the overfitting reduction benefits associated with ensembles, we want to use ensemble learning to derive soft labels that quantify the uncertainty inherent in labeled data. We hope that the derived soft labels will help us boost abstention rates so that the model can be used in a larger subset of data. In this chapter, we tackle the problem of data and label noise with soft labels.

In Chapter 5 we experimented with distilling the knowledge of 1,000 MtCNN models into a single MtCNN. We showed that ensemble predictions provide a useful strategy for quantifying the uncertainty inherent in labeled data, enabling the construction of soft labels with estimated probabilities for multiple classes for a given document. A key takeaway of this work is that the derived soft-labels provide an alternative solution to deal with the extreme levels of noise found in the pathology reports. Documents that are hard to classify (either because they belong to rare classes or because of the noise present) are naturally associated with high ensemble disagreement. As a result, these documents receive soft-labels which tend to be uniform, allowing for degrees of truth that more accurately represent such pathology reports. We argued that this has implications in terms of predictive confidence because during training, the model is not guided to predict with a confident of 1 for these noisy documents. Predictive confidence is directly connected to softmax-based abstention, and we showed that ensemble model distillation can effectively boost abstention performance. Ultimately, we showed that by distilling the knowledge into a single model, cancer registries could deploy a low-resource classifier that will abstain less and allow them to apply the model to a higher percent of documents.

5.2 Future Work

5.2.1 Registry-specific models

As we previously stated, differences in data collection and processing across registries create variations which likely hurt the model performance. A standard pathology report from the raw dataset, before we apply preprocessing steps, consists of different sections such as *textPathComments*, *textPathFormalDx*, *textPathGrossPathology*, and *textPathNatureOfSpecimens*. Depending on the registry that collected a specific report, some document populate all these fields while other pathology reports from a different registry may only populate half of these sections. Thus, differences in the way information is stored create natural variations between the dataset. One natural way to deal with these variations is by training registry-specific models which are trained using only the data from those registries. In such scenario, these models would then only be deployed in their respective registries. Advantages of having registry-specific models include 1) data will be consistent which is likely to help during model training, and 2) the class distribution would be representative of the population, leading to more calibrated prior probabilities.

A negative side of this approach is that if constitutes one step away from the ideal goal of having a single model that works for all cancer pathology reports. In addition, such solution will involve the maintenance and recurrent training of multiple models (one for each registry) which, in the long term, may be expensive and time-consuming.

5.2.2 Class Hierarchy, Cascade Learning, and Domain Knowledge

The dense layer that generates the output of the current MtCNN model treats classes as independent. However, there exists some natural relations between classes. For example, a prediction of Site limits the subset of potential Subsite predictions. That is, predicting *Site = C34: Bronchus and Lungs* and *Site = C50.1: Central portion of*

breast for the same document is a clear contradiction. Currently, although uncommon, there is nothing in the model that prevents these contradictions.

We argue that class relations not only should be taken into consideration to avoid illogical predictions, but these relations can be used to model the problem more efficiently. Manually inspecting and defining rules for all class relationship could be unfeasible. However, integrating a few conditions as domain knowledge could simplify the problem and potentially boost performance. As an example, the official coding manual states that when labeling Laterality one should "Assign code 0 when a. The primary site is not a paired site, b. Primary site is unknown (C809), [...]". The significance of this coding rule is that a prediction for Site could potentially give you the prediction for Laterality and vice versa.

From a modeling perspective, one can attempt to model the problem so that a prediction for one task is taken into consideration when predicting some other class. Future research could investigate redesigning the MtCNN as a *cascade* model where, for example, a prediction for Site is fed into another layer that produces the prediction for Subsite. Alternative, one could attempt to allow the model to decide for what classes it produces a prediction based on the input document and then use those outputs to predict the rest of the tasks.

5.2.3 Alternative Data Pre-processing

The current prepossessing steps involve concatenating all the available sections (see Section 5.2.1) of the pathology into one single document. This practice was adopted because 1) most NLP models were developed to accept one input, and 2) because of the variation across registries, it is never guaranteed that a pathology report will include text in all sections (for example, not all reports include *textPathNatureOfSpecimens*).

However, we argue that there is a possibility that we may be losing information when concatenating all the fields of the path report into one main document. That

is, perhaps the most important word segments for one class are included in one of the sections. By concatenating all the fields, we may be generating noise and losing some structural information that could be exploit by a machine learning model. Future research direction could investigate variations of multi-input neural networks such as the one implemented in [98].

5.2.4 Problem Complexity Reduction

The histology and subsite tasks consists of 639 and 326 classes, respectively. As previously mentioned, some of these classes are extremely rare and a very small percentage of the data belongs to these classes. As a result, ML models may not have enough information to identify patterns to successfully classify these minority classes at test time. Considering the low error rate requirements, we hypothesize that it may be beneficial to focus on the classes which appear frequently enough so that the ML models can extract the features to accurately classify these classes. However, we acknowledge that there are numerous challenges associated with this approach. Firstly, the model will have to be calibrated well enough so that when it encounters one of the left-out classes, it abstains to predict; otherwise, these minority classes would be erroneously classified as one of the more frequent classes. Secondly, it can be challenging to establish a threshold to decide what classes the model knows well enough to include and which classes should be excluded.

Bibliography

[1] M. Alawad, S. Gao, J. Qiu, N. Schaefferkoetter, J. D. Hinkle, H.-J. Yoon, J. B. Christian, X.-C. Wu, E. B. Durbin, J. C. Jeong, I. Hands, D. Rust, and G. Tourassi. Deep transfer learning across cancer registries for information extraction from pathology reports. In *2019 IEEE EMBS International Conference on Biomedical Health Informatics (BHI)*, pages 1–4, 2019. 62, 69, 92

[2] M. Alawad, S. Gao, J. X. Qiu, H. J. Yoon, J. Blair Christian, L. Penberthy, B. Mumphrey, X.-C. Wu, L. Coyle, and G. Tourassi. Automatic extraction of cancer registry reportable information from free-text pathology reports using multitask convolutional neural networks. *Journal of the American Medical Informatics Association*, 27(1):89–98, 11 2019. 3, 4, 62, 69, 75, 86, 92

[3] M. Alawad, S. Gao, J. X. Qiu, H. J. Yoon, J. Blair Christian, L. Penberthy, B. Mumphrey, X.-C. Wu, L. Coyle, and G. Tourassi. Automatic extraction of cancer registry reportable information from free-text pathology reports using multitask convolutional neural networks. *Journal of the American Medical Informatics Association*, 27(1):89–98, 11 2019. 14, 15, 26, 27, 28

[4] A. Ali, S. M. Shamsuddin, and A. Ralescu. Classification with class imbalance problem: A review. 7:176–204, 01 2015. 6, 55

[5] S. Argamon-Engelson and I. Dagan. Committee-based sample selection for probabilistic classifiers. *CoRR*, abs/1106.0220, 2011. 23

[6] C. Bellinger, R. Corizzo, and N. Japkowicz. Remix: Calibrated resampling for class imbalance in deep learning. *CoRR*, abs/2012.02312, 2020. 55

[7] R. Blagus and L. Lusa. Smote for high-dimensional class-imbalanced data. *BMC bioinformatics*, 14:106, 03 2013. 58

[8] A. Blanchard, S. Gao, H.-J. Yoon, B. Christian, E. B. Durbin, X.-C. Wu, A. Stroup, J. Doherty, S. Schwartz, C. Wiggins, L. Coyle, L. Penberthy, and

G. D. Tourassi. A keyword-enhanced approach to handle class imbalance in clinical text classification. *IEEE Journal of Biomedical and Health Informatics*, pages 1–1, 2022. 92

[9] L. Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996. 61

[10] L. Breiman. Bagging predictors. *Machine Learning*, 24:123–140, 2004. 94

[11] L. Breiman. Bagging predictors. *Machine Learning*, 24:123–140, 2004. 106

[12] W. Bridewell, N. B. Asadi, P. Langley, and L. Todorovski. Reducing overfitting in process model induction. In *Proceedings of the 22nd International Conference on Machine Learning*, ICML '05, page 81–88, New York, NY, USA, 2005. Association for Computing Machinery. 89

[13] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, Jun 2002. 58

[14] Y. Chen, S. Mani, and H. Xu. Applying active learning to assertion classification of concepts in clinical text. *Journal of biomedical informatics*, 45:265–72, 11 2011. 18

[15] C. Chow. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16(1):41–46, 1970. 88

[16] C. K. Chow. An optimum character recognition system using decision functions. *IRE Transactions on Electronic Computers*, EC-6(4):247–254, 1957. 88

[17] P. Cunningham. Overfitting and diversity in classification ensembles based on feature selection. 2000. 89

[18] A. D'Amour, K. Heller, D. Moldovan, B. Adlam, B. Alipanahi, A. Beutel, C. Chen, J. Deaton, J. Eisenstein, M. D. Hoffman, F. Hormozdiari, N. Houlsby,

S. Hou, G. Jerfel, A. Karthikesalingam, M. Lucic, Y. Ma, C. McLean, D. Mincu, A. Mitani, A. Montanari, Z. Nado, V. Natarajan, C. Nielson, T. F. Osborne, R. Raman, K. Ramasamy, R. Sayres, J. Schrouff, M. Seneviratne, S. Sequeira, H. Suresh, V. Veitch, M. Vladymyrov, X. Wang, K. Webster, S. Yadlowsky, T. Yun, X. Zhai, and D. Sculley. Underspecification presents challenges for credibility in modern machine learning, 2020. 54, 60

[19] A. D'Amour, K. A. Heller, D. Moldovan, B. Adlam, B. Alipanahi, A. Beutel, C. Chen, J. Deaton, J. Eisenstein, M. D. Hoffman, F. Hormozdiari, N. Houlsby, S. Hou, G. Jerfel, A. Karthikesalingam, M. Lucic, Y. Ma, C. Y. McLean, D. Mincu, A. Mitani, A. Montanari, Z. Nado, V. Natarajan, C. Nielson, T. F. Osborne, R. Raman, K. Ramasamy, R. Sayres, J. Schrouff, M. Seneviratne, S. Sequeira, H. Suresh, V. Veitch, M. Vladymyrov, X. Wang, K. Webster, S. Yadlowsky, T. Yun, X. Zhai, and D. Sculley. Underspecification presents challenges for credibility in modern machine learning. *CoRR*, abs/2011.03395, 2020. 106

[20] K. De Angeli, S. Gao, M. Alawad, H.-J. Yoon, N. Schaefferkoetter, X.-C. Wu, E. B. Durbin, J. Doherty, A. Stroup, L. Coyle, L. Penberthy, and G. Tourassi. Deep active learning for classifying cancer pathology reports. *BMC Bioinformatics*, 22(1):113, Mar. 2021. 54, 62, 69, 92

[21] K. De Angeli, S. Gao, I. Danciu, E. B. Durbin, X.-C. Wu, A. Stroup, J. Doherty, S. Schwartz, C. Wiggins, M. Damesyn, L. Coyle, L. Penberthy, G. D. Tourassi, and H.-J. Yoon. Class imbalance in out-of-distribution datasets: Improving the robustness of the textcnn for the classification of rare cancer types. *Journal of Biomedical Informatics*, 125:103957, 2022. 91, 92

[22] A. Desai, A. R. Khaki, and N. M. Kuderer. Use of Real-World Electronic Health Records to Estimate Risk, Risk Factors, and Disparities for COVID-19 in Patients With Cancer. *JAMA Oncology*, 7(2):227–229, 02 2021. 2

[23] T. J. DiCiccio and B. Efron. Bootstrap confidence intervals. *Statistical Science*, 11(3):189–212, 1996. 31

[24] J. Djolonga, J. Yung, M. Tschannen, R. Romijnders, L. Beyer, A. Kolesnikov, J. Puigcerver, M. Minderer, A. D'Amour, D. Moldovan, S. Gelly, N. Houlsby, X. Zhai, and M. Lucic. On robustness and transferability of convolutional neural networks, 2021. 60

[25] A. Dogan and D. Birant. A weighted majority voting ensemble approach for classification. In *2019 4th International Conference on Computer Science and Engineering (UBMK)*, pages 1–6, 2019. 98

[26] R. El-Yaniv and Y. Wiener. On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, 11(53):1605–1641, 2010. 88

[27] R. S. Evans. Electronic health records: Then, now, and in the future. *IMIA Yearbook*, 25, 05 2016. 2

[28] S. Feng, E. Wallace, A. Grissom II, M. Iyyer, P. Rodriguez, and J. Boyd-Graber. Pathologies of neural models make interpretations difficult. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3719–3728, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics. 88

[29] A. Fernández, S. García, F. Herrera, and N. V. Chawla. Smote for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary. *J. Artif. Int. Res.*, 61(1):863–905, Jan. 2018. 58

[30] R. Figueroa, Q. Zeng-Treitler, L. Ngo, S. Goryachev, and E. Wiechmann. Active learning for clinical text classification: Is it better than random sampling? *Journal of the American Medical Informatics Association : JAMIA*, 19:809–16, 06 2012. 18

[31] Y. Freund, Y. Mansour, and R. E. Schapire. Generalization bounds for averaged classifiers. *The Annals of Statistics*, 32(4):1698 – 1722, 2004. 90

[32] Y. Freund and R. E. Schapire. A short introduction to boosting. In *In Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, pages 1401–1406. Morgan Kaufmann, 1999. 94

[33] Y. Gal, R. Islam, and Z. Ghahramani. Deep bayesian active learning with image data. *CoRR*, abs/1703.02910, 2017. 16

[34] S. Gao, M. Alawad, N. Schaefferkoetter, L. Penberthy, X.-C. Wu, E. B. Durbin, L. M. Coyle, A. Ramanathan, and G. D. Tourassi. Using case-level context to classify cancer pathology reports. *PLOS ONE*, 15(5):1–21, 2020. 15, 26

[35] S. Gao, M. Alawad, M. T. Young, J. Gounley, N. Schaefferkoetter, H.-J. Yoon, X.-C. Wu, E. B. Durbin, J. Doherty, A. Stroup, L. Coyle, and G. D. Tourassi. Limitations of transformers on clinical text classification. *IEEE Journal of Biomedical and Health Informatics*, pages 1–1, 2021. 3, 4, 62, 86, 92

[36] S. Gao, J. X. Qiu, M. Alawad, J. D. Hinkle, N. Schaefferkoetter, H.-J. Yoon, B. Christian, P. A. Fearn, L. Penberthy, X.-C. Wu, L. Coyle, G. Tourassi, and A. Ramanathan. Classifying cancer pathology reports with hierarchical self-attention networks. *Artificial Intelligence in Medicine*, 101:101726, 2019. 3, 4, 54, 86, 92

[37] S. Gao, J. X. Qiu, M. Alawad, J. D. Hinkle, N. Schaefferkoetter, H.-J. Yoon, B. Christian, P. A. Fearn, L. Penberthy, X.-C. Wu, L. Coyle, G. Tourassi, and A. Ramanathan. Classifying cancer pathology reports with hierarchical self-attention networks. *Artificial Intelligence in Medicine*, 101:101726, 2019. 14, 15, 16, 26, 27, 28

[38] S. Gao, A. Ramanathan, and G. Tourassi. Hierarchical convolutional attention networks for text classification. In *Proceedings of The Third Workshop on*

Representation Learning for NLP, pages 11–23, Melbourne, Australia, July 2018. Association for Computational Linguistics. 3, 4, 54, 86

[39] S. Gao, M. T. Young, J. X. Qiu, H.-J. Yoon, J. B. Christian, P. A. Fearn, G. D. Tourassi, and A. Ramanthan. Hierarchical attention networks for information extraction from cancer pathology reports. *Journal of the American Medical Informatics Association*, 25(3):321–330, 11 2017. 2, 54

[40] Y. Geifman and R. El-Yaniv. Selective classification for deep neural networks. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4885–4894, Red Hook, NY, USA, 2017. Curran Associates Inc. 88

[41] R. Geirhos, J. Jacobsen, C. Michaelis, R. S. Zemel, W. Brendel, M. Bethge, and F. A. Wichmann. Shortcut learning in deep neural networks. *CoRR*, abs/2004.07780, 2020. 87

[42] R. Geirhos, J.-H. Jacobsen, C. Michaelis, R. Zemel, W. Brendel, M. Bethge, and F. A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, Nov 2020. 60

[43] X. Guo, Y. Yin, C. Dong, G. Yang, and G. Zhou. On the class imbalance problem. In *2008 Fourth International Conference on Natural Computation*, volume 4, pages 192–201, 2008. 6, 55

[44] H. S. C. H., R. Jin, J. Zhu, and M. R. Lyu. Batch mode active learning and its application to medical image classification. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 417–424, New York, NY, USA, 2006. Association for Computing Machinery. 16

[45] H. Han, W.-Y. Wang, and B.-H. Mao. Borderline-smote: A new over-sampling method in imbalanced data sets learning. In D.-S. Huang, X.-P. Zhang, and G.-B. Huang, editors, *Advances in Intelligent Computing*, pages 878–887, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg. 58

[46] B. He, Y. Guan, and R. Dai. Classifying medical relations in clinical text via convolutional neural networks. *Artificial Intelligence in Medicine*, 93:43–49, 2019. Extracting and Processing of Rich Semantics from Medical Texts. 62

[47] M. E. Hellman. The nearest neighbor classification rule with a reject option. *IEEE Trans. Syst. Sci. Cybern.*, 6:179–185, 1970. 88

[48] D. Hendrycks, S. Basart, N. Mu, S. Kadavath, F. Wang, E. Dorundo, R. Desai, T. Zhu, S. Parajuli, M. Guo, D. Song, J. Steinhardt, and J. Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization, 2020. 60

[49] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network, 2015. 62

[50] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network, 2015. 90

[51] M. Hughes, I. Li, S. Kotoulas, and T. Suzumura. Medical text classification using convolutional neural networks. *Studies in Health Technology and Informatics*, 235, 04 2017. 62

[52] P. S. . H. J. Hospitals' use of electronic health records data. *Office of the National Coordinator for Health Information Technology*, 5, 04 2019. 2

[53] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991. 94

[54] J. Johnson and T. Khoshgoftaar. Survey on deep learning with class imbalance. *Journal of Big Data*, 6:27, 03 2019. 6, 55, 58, 59

[55] K. S. Kalyan, A. Rajasekharan, and S. Sangeetha. Ammus : A survey of transformer-based pretrained models in natural language processing, 2021. 4

[56] Kholghi, Mahnoosh, Sitbon, Laurianne, Zuccon, Guido, and A. Nguyen. Active learning: A step towards automating medical concept extraction. *Journal of the American Medical Informatics Association*, 05 2015. 18

[57] M. Kubat and S. Matwin. Addressing the curse of imbalanced training sets: One-sided selection. In *In Proceedings of the Fourteenth International Conference on Machine Learning*, pages 179–186. Morgan Kaufmann, 1997. 57

[58] B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. NIPS'17, page 6405–6416, Red Hook, NY, USA, 2017. Curran Associates Inc. 94

[59] H. Lee, M. Park, and J. Kim. Plankton classification on imbalanced large scale database via convolutional neural networks with transfer learning. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 3713–3717, 2016. 59, 64

[60] S. Lee, S. Purushwalkam, M. Cogswell, D. Crandall, and D. Batra. Why m heads are better than one: Training a diverse ensemble of deep networks, 2015. 94

[61] S. Lee, Y. Xu, A. D'Souza, E. Martin, C. Doktorchik, Z. Zhang, and H. Quan. Unlocking the potential of electronic health records for health research. *International Journal of Population Data Science*, 5, 01 2020. 2

[62] D. D. Lewis and W. A. Gale. A sequential algorithm for training text classifiers, 1994. 19

[63] Z. Li, K. Kamnitsas, and B. Glocker. Overfitting of neural nets under class imbalance: Analysis and improvements for segmentation. In D. Shen, T. Liu, T. M. Peters, L. H. Staib, C. Essert, S. Zhou, P.-T. Yap, and A. Khan, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, pages 402–410, Cham, 2019. Springer International Publishing. 87

[64] Z. Li, K. Kamnitsas, and B. Glocker. Analyzing overfitting under class imbalance in neural networks for image segmentation. *IEEE Transactions on Medical Imaging*, 40(3):1065–1077, 2021. 87

[65] Z. Li, K. Kamnitsas, and B. Glocker. Analyzing overfitting under class imbalance in neural networks for image segmentation. *IEEE Transactions on Medical Imaging*, 40(3):1065–1077, 2021. 110

[66] C. Ling, , C. X. Ling, and C. Li. Data mining for direct marketing: Problems and solutions. In *In Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD-98*, pages 73–79. AAAI Press, 1998. 58

[67] D. Masko and P. Hensman. The impact of imbalanced training data for convolutional neural networks. 2015. 57

[68] J. Miller, K. Krauth, B. Recht, and L. Schmidt. The effect of natural distribution shift on question answering models, 2020. 60

[69] T. Miyato, S.-i. Maeda, M. Koyama, and S. Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1979–1993, 2018. 61

[70] H. N, N. AM, K. M, M. D, B. A, Y. M, R. J, T. Z, M. A, L. DR, C. HS, F. EJ, and C. K. (eds). Seer cancer statistics review, 1975-2017. *National Cancer Institute*, 04 2020. 15

[71] F. Olsson. A literature survey of active machine learning in the context of natural language processing. 05 2009. 17

[72] C. Padurariu and M. E. Breaban. Dealing with data imbalance in text classification. *Procedia Computer Science*, 159:736–745, 2019. Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 23rd International Conference KES2019. 58, 59

[73] F. Pereira, N. Tishby, and L. Lee. Distributional clustering of English words. In *31st Annual Meeting of the Association for Computational Linguistics*, pages 183–190, Columbus, Ohio, USA, June 1993. Association for Computational Linguistics. 23

[74] R. Polikar. Ensemble learning. In *Ensemble machine learning*, pages 1–34. Springer, 2012. 61

[75] S. Pouyanfar, Y. Tao, A. Mohan, H. Tian, A. S. Kaseb, K. Gauen, R. Dailey, S. Aghajanzadeh, Y.-H. Lu, S.-C. Chen, and M.-L. Shyu. Dynamic sampling in convolutional neural networks for imbalanced data classification. In *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 112–117, 2018. 57

[76] J. Qiu, H.-J. Yoon, P. A. Fearn, and G. D. Tourassi. Deep learning for automated extraction of primary sites from cancer pathology reports. *IEEE Journal of Biomedical and Health Informatics*, 22(1), 5 2017. 28

[77] J. X. Qiu, H.-J. Yoon, P. A. Fearn, and G. D. Tourassi. Deep learning for automated extraction of primary sites from cancer pathology reports. *IEEE Journal of Biomedical and Health Informatics*, 22(1):244–251, 2018. 3

[78] J. X. Qiu, H.-J. Yoon, P. A. Fearn, and G. D. Tourassi. Deep learning for automated extraction of primary sites from cancer pathology reports. *IEEE Journal of Biomedical and Health Informatics*, 22(1):244–251, 2018. 70

[79] E. Rendón, R. Alejo, C. Castorena, F. J. Isidro-Ortega, and E. E. Granda-Gutiérrez. Data sampling methods to deal with the big data multi-class imbalance problem. *Applied Sciences*, 10(4), 2020. 6, 55, 59

[80] G. Riccardi and D. Hakkani-Tur. Active learning: theory and applications to automatic speech recognition. *IEEE Transactions on Speech and Audio Processing*, 13(4):504–511, 2005. 16

[81] A. Rios and R. Kavuluru. Convolutional neural networks for biomedical text classification: Application in indexing biomedical articles. In *Proceedings of the 6th ACM Conference on Bioinformatics, Computational Biology and Health Informatics*, BCB '15, page 258–267, New York, NY, USA, 2015. Association for Computing Machinery. 62

[82] N. Roy and A. McCallum. Toward optimal active learning through sampling estimation of error reduction. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, page 441–448, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc. 24

[83] G. K. Savova, I. Danciu, F. Alamudun, T. Miller, C. Lin, D. S. Bitterman, G. Tourassi, and J. L. Warner. Use of natural language processing to extract clinical cancer phenotypes from electronic medical records. *Cancer Research*, 79(21):5463–5470, 2019. 62

[84] R. E. Schapire. The boosting approach to machine learning an overview. 2003. 106

[85] A. Schein and L. Ungar. Active learning for logistic regression: an evaluation. *Mach Learn 68*, page 235–265, 05 2007. 19

[86] SEER. Icd-0-3 seer site/histology validation llist, 2020. xv, 67, 73, 147

[87] Settles, Burr, Craven, and Mark. An analysis of active learning strategies for sequence labeling tasks. In *Proceedings of the Conference on Empirical*

Methods in Natural Language Processing, EMNLP '08, page 1070–1079, USA, 2008. Association for Computational Linguistics. 17, 24

[88] B. Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009. 15, 17, 21

[89] B. Settles. From theories to queries: Active learning in practice. In I. Guyon, G. Cawley, G. Dror, V. Lemaire, and A. Statnikov, editors, *Active Learning and Experimental Design workshop In conjunction with AISTATS 2010*, volume 16 of *Proceedings of Machine Learning Research*, pages 1–18, Sardinia, Italy, 16 May 2011. PMLR. 17

[90] H. S. Seung, M. Opper, and H. Sompolinsky. Query by committee. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, page 287–294, New York, NY, USA, 1992. Association for Computing Machinery. 21

[91] C. E. Shannon. A mathematical theory of communication, 1948. 20

[92] H.-C. Shao, H.-C. Wang, W.-T. Su, and C.-W. Lin. Ensemble learning with manifold-based data splitting for noisy label correction. *IEEE Transactions on Multimedia*, 24:1127–1140, 2022. 105

[93] N. Shazeer, A. Mirhoseini, K. Maziarz, A. Davis, Q. Le, G. Hinton, and J. Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*, 2017. 62

[94] Y. Shen, H. Yun, Z. C. Lipton, Y. Kronrod, and A. Anandkumar. Deep active learning for named entity recognition. *CoRR*, abs/1707.05928, 2017. 17

[95] R. L. Siegel, K. D. Miller, H. E. Fuchs, and A. Jemal. Cancer statistics, 2022. *CA: A Cancer Journal for Clinicians*, 72(1):7–33, 2022. 86, 104

[96] K. Stacke, G. Eilertsen, J. Unger, and C. Lundström. Measuring domain shift for deep learning in histopathology. *IEEE Journal of Biomedical and Health Informatics*, 25(2):325–336, 2021. 55, 60

[97] Y. Sun, X. Wang, Z. Liu, J. Miller, A. Efros, and M. Hardt. Test-time training with self-supervision for generalization under distribution shifts. In H. D. III and A. Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 9229–9248. PMLR, 13–18 Jul 2020. 60

[98] Y. Sun, L. Zhu, G. Wang, and F. Zhao. Multi-input convolutional neural network for flower grading. *Journal of Electrical and Computer Engineering*, 2017:1–8, 08 2017. 116

[99] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826, 2016. 88

[100] R. Taori, A. Dave, V. Shankar, N. Carlini, B. Recht, and L. Schmidt. Measuring robustness to natural distribution shifts in image classification, 2020. 60, 78

[101] C. A. Thompson, M. E. Califf, and R. J. Mooney. Active learning for natural language parsing and information extraction. In *Proceedings of the Sixteenth International Conference on Machine Learning (ICML-99)*, pages 406–414, Bled, Slovenia, June 1999. 16

[102] S. Thulasidasan, T. Bhattacharya, J. Bilmes, G. Chennupati, and J. Mohd-Yusof. Combating label noise in deep learning using abstention, 2019. 88

[103] S. Thulasidasan, G. Chennupati, J. Bilmes, T. Bhattacharya, and S. Michalak. On mixup training: Improved calibration and predictive uncertainty for deep neural networks, 2020. 88

[104] L. Torgo, R. P. Ribeiro, B. Pfahringer, and P. Branco. Smote for regression. In L. Correia, L. P. Reis, and J. Cascalho, editors, *Progress in Artificial Intelligence*, pages 378–389, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. 58

[105] D. Tuia, M. Volpi, L. Copa, M. Kanevski, and J. Munoz-Mari. A survey of active learning algorithms for supervised remote sensing image classification. *IEEE Journal of Selected Topics in Signal Processing*, 5(3):606–617, 2011. 17

[106] J. Van Hulse, T. M. Khoshgoftaar, and A. Napolitano. Experimental perspectives on learning from imbalanced data. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, page 935–942, New York, NY, USA, 2007. Association for Computing Machinery. 57

[107] K. R. Varshney. A risk bound for ensemble classification with a reject option. In *2011 IEEE Statistical Signal Processing Workshop (SSP)*, pages 769–772, 2011. 90

[108] K. Wang, D. Zhang, Y. Li, R. Zhang, and L. Lin. Cost-effective active learning for deep image classification. *CoRR*, abs/1701.03551, 2017. 17

[109] M. Wang and X.-S. Hua. Active learning in multimedia annotation and retrieval: A survey. *ACM Trans. Intell. Syst. Technol.*, 2(2), Feb. 2011. 17

[110] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, and A. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, Oct. 2020. Association for Computational Linguistics. 4

[111] J. Wu, X. Li, X. Ao, Y. Meng, F. Wu, and J. Li. Improving robustness and generality of nlp models using disentangled representations, 2020. 60, 61

[112] C. Xiao, E. Choi, and J. Sun. Opportunities and challenges in developing deep learning models using electronic health records data: A systematic review. *Journal of the American Medical Informatics Association*, 25, 06 2018. 28

[113] H. S. Yahia and A. Abdulazeez. Medical text classification based on convolutional neural network: A review. *International Journal of Science and Business*, 5(3):27–41, 2021. 62

[114] Y.-Y. Yang and K. Chaudhuri. Understanding rare spurious correlations in neural networks, 02 2022. 87

[115] L. Yao, C. Mao, and Y. Luo. Clinical text classification with rule-based features and knowledge-guided convolutional neural networks, 2018. 62

[116] M. Y. Yildirim, M. Ozer, and H. Davulcu. Leveraging uncertainty in deep learning for selective classification. *CoRR*, abs/1905.09509, 2019. 88

[117] H.-J. Yoon, J. Gounley, M. T. Young, and G. Tourassi. Information extraction from cancer pathology reports with graph convolution networks for natural language texts. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 4561–4564, 2019. 4

[118] T. Young, D. Hazarika, S. Poria, and E. Cambria. Recent trends in deep learning based natural language processing, 2017. 14

[119] T. Young, D. Hazarika, S. Poria, and E. Cambria. Recent trends in deep learning based natural language processing, 2017. 28

[120] Y. Zhang and B. C. Wallace. Active discriminative word embedding learning. *CoRR*, abs/1606.04212, 2016. 17

[121] Z.-H. Zhou. Ensemble learning. *Encyclopedia of biometrics*, 1:270–273, 2009. 62

[122] X. Zou, Z. Zhang, Z. He, and L. Shi. *Unsupervised Ensemble Learning with Noisy Label Correction*, page 2308–2312. Association for Computing Machinery, New York, NY, USA, 2021. 105

Appendices

A Chapter 2 Supporting Information

A.1 Additional file 1 — Dataset class imbalance plots.

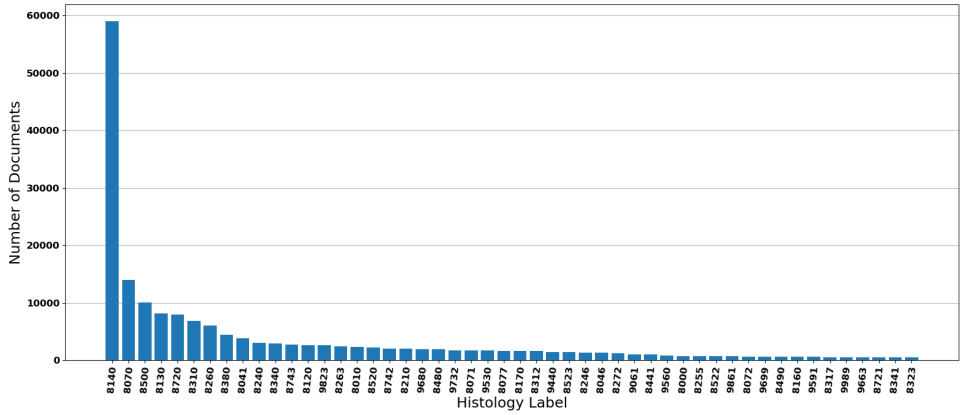


Figure A.1: Class imbalance plots.

A.2 Additional file 2 — Text preprocessing steps.

1. Remove identifier segments (registry ID, patient ID, tumor number, and document ID)

2. Remove XML tags

3. Convert unicode to ASCII
4. Lowercase
5. Replace tabs and line breaks with spaces
6. Replacing all instances of floats with the string “floattoken”
7. Replace all integers higher than 100 with the string “largeinttoken” (to reduce the number of unique tokens associated with numbers)
8. If the same non-alphanumeric character appears consecutively more than once, replace it with a single copy of that character
9. Add a space before and after every non-alphanumeric character
10. Remove words longer than 25 characters to reduce noise (these are generally artifacts from format conversions)
11. Tokenize document
12. Replace any token that appears less than 5 times across the entire corpus with the string “unknowntoken”
13. Add padding or truncate document to 1500 tokens

A.3 Additional file 3 — Bootstrapping procedure for confidence. interval

The 95% confidence intervals presented in this paper were computed as follows:

1. Compute and save the accuracy of every model on the test data set; this is the original accuracy.
2. Create a new dataset by sampling documents with replacement from the test dataset.

3. Find the accuracy of the model in this new dataset.

4. Repeat steps two and three 2,000 times and store all accuracy results.

5. Calculate the 95% confidence interval by finding the 2.5 and 97.5 percentiles.

A.4 Additional file 4 — Performance of cold start ratio sampling, warm start ratio sampling, and random sampling on the histology task (small dataset).

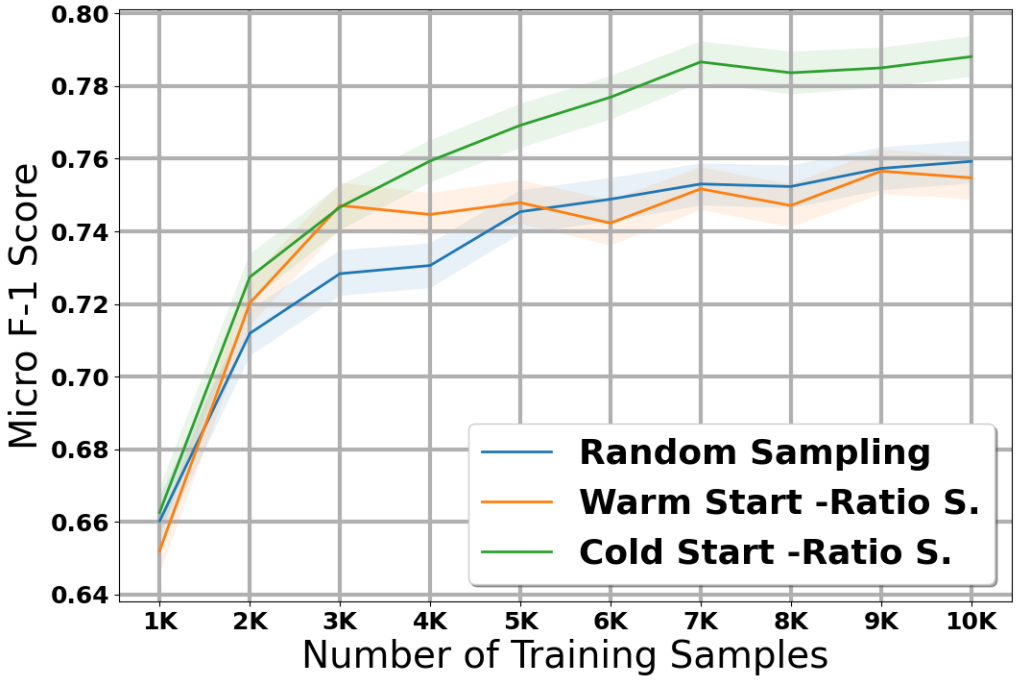


Figure A.2: Cold start ratio sampling vs. warm start ratio sampling.

A.5 Additional file 5 — Training/testing and inference time for the histology experiment (large dataset) using ratio sampling.

Table A.1: Training inference time.

Data Used	Time (seconds)
15K	811.93
30K	1328.5
45K	1255.31
60K	1785.73
75K	1361.69
90K	1835.19
105K	1814.94
120K	1857.69
135K	2551.22
150K	3206.75
162K	3228.64

Table A.2: Testing inference time.

Time Complexity	1 Sample
(Testing)	0.000419609035934 Seconds

A.6 Additional file 6 — Large dataset: micro/macro F-1 scores table - histology task.

This table was included as an external file in excel format (see AF_1_6.xlsx).

A.7 Additional file 7 — Large dataset: micro/macro F-1 scores table - subsite task.

This table was included as an external file in excel format (see AF_1_7.xlsx).

A.8 Additional file 8 — Small dataset: micro/macro F-1 scores table - histology task.

This table was included as an external file in excel format (see AF_1_8.xlsx).

A.9 Additional file 9 — Small dataset: micro/macro F-1 scores table - subsite task.

This table was included as an external file in excel format (see AF_1_9.xlsx).

A.10 Additional file 10 — Large dataset: class imbalance - histology task.

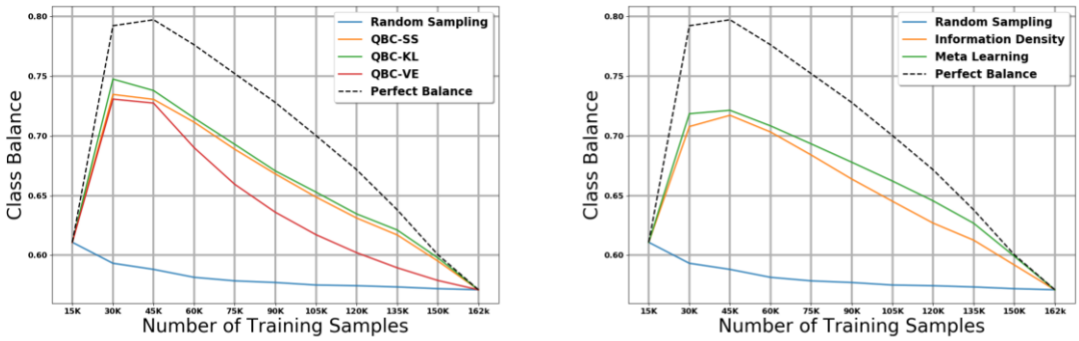


Figure A.3: class imbalance - histology task.

A.11 Additional file 11 — Large dataset: class imbalance - subsite task.

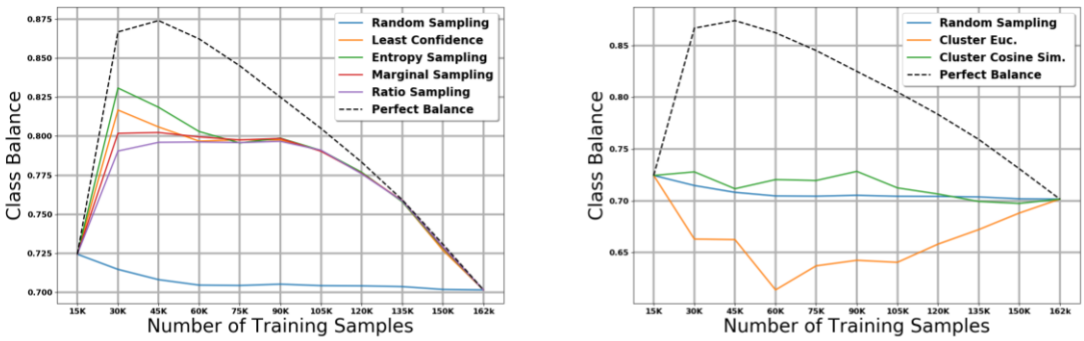


Figure A.4: class imbalance - subsite task.

A.12 Additional file 12 — Small dataset: class imbalance - histology task.

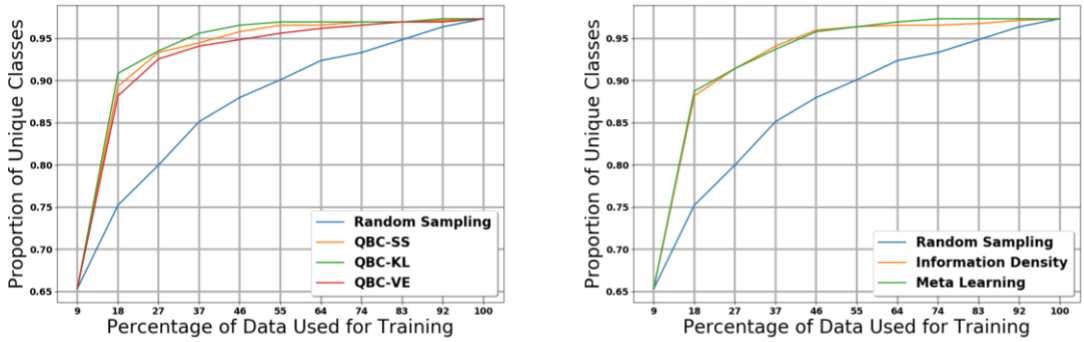


Figure A.5: class imbalance - histology task.

A.13 Additional file 13 — Small dataset: class imbalance - subsite task.

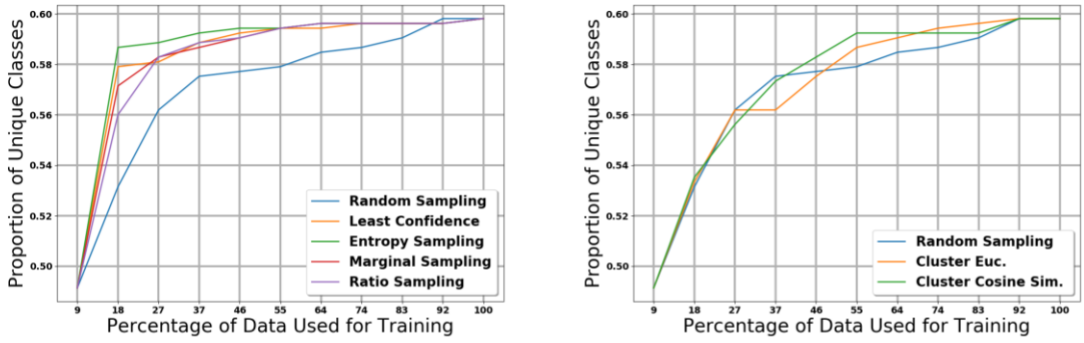


Figure A.6: class imbalance - subsite task.

A.14 Additional file 14 — Large dataset: class proportion plots - histology.

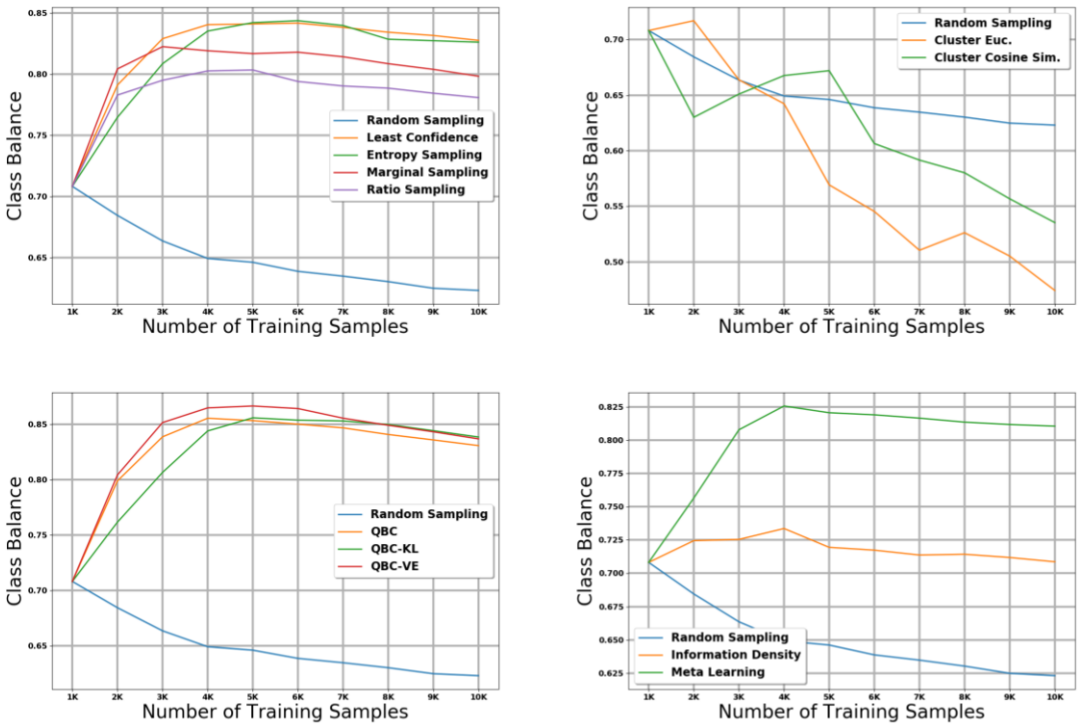


Figure A.7: Large dataset: class proportion plots - histology task.

A.15 Additional file 15 — Large dataset: class proportion plots - subsite task.

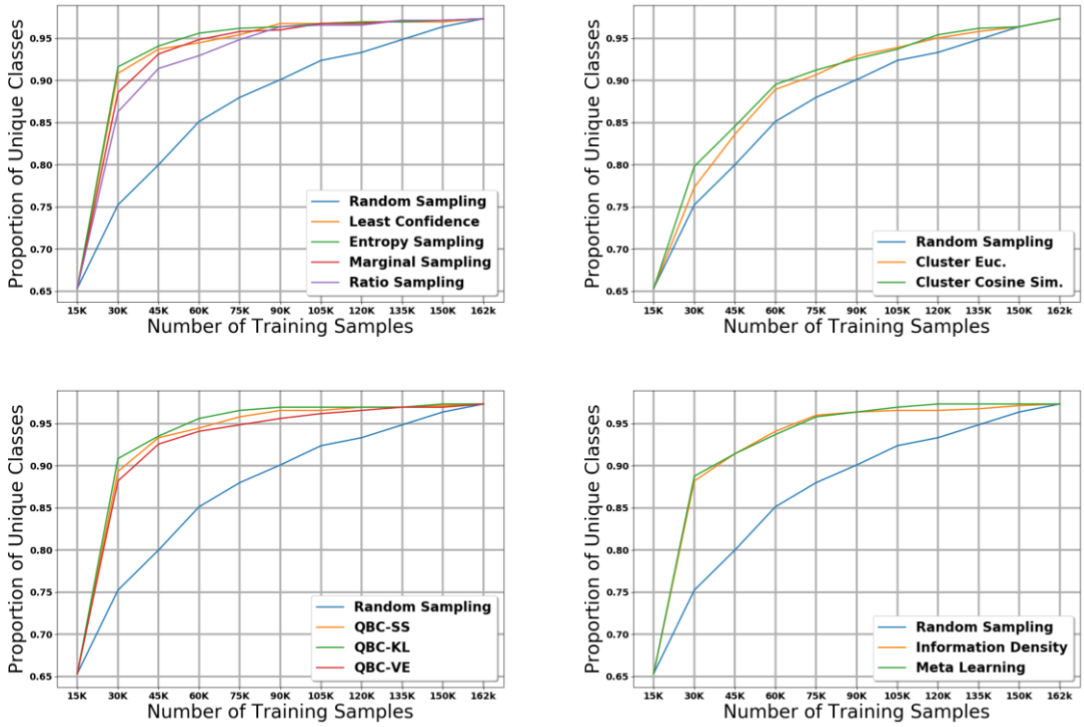


Figure A.8: Large dataset: class proportion plots - subsite task

A.16 Additional file 16 — Small dataset: class proportion plots - histology task.

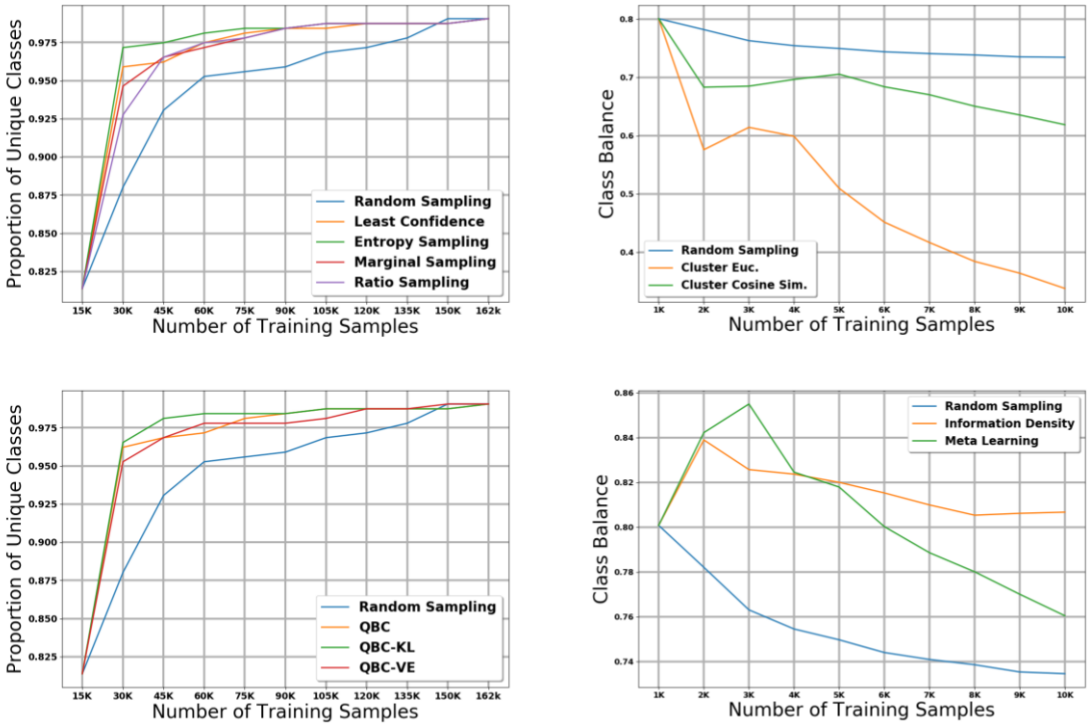


Figure A.9: Small dataset: class proportion plots - histology task

A.17 Additional file 17 — Small dataset: class proportion plots - subsite task.

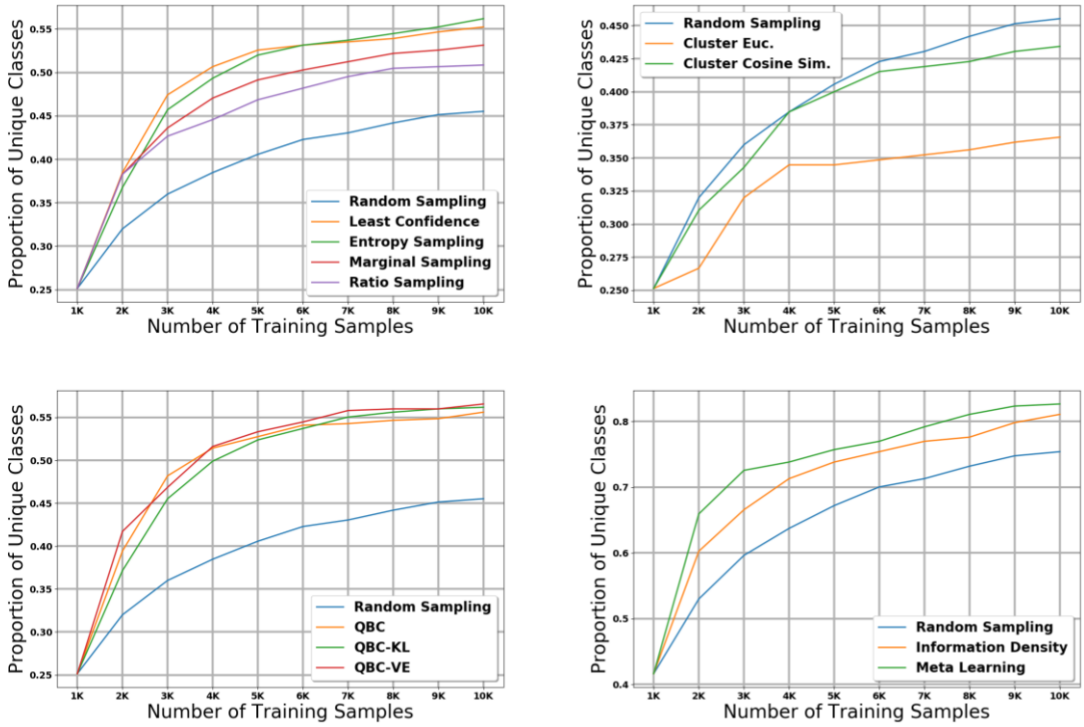


Figure A.10: class proportion plots - subsite task.

A.18 Additional file 18 — Document embeddings generated via TSNE for histology task (small dataset) with and without 10 iterations of active learning. Documents are colored by majority class (number of total samples in dataset above average) and minority class (number of total samples in dataset below average).

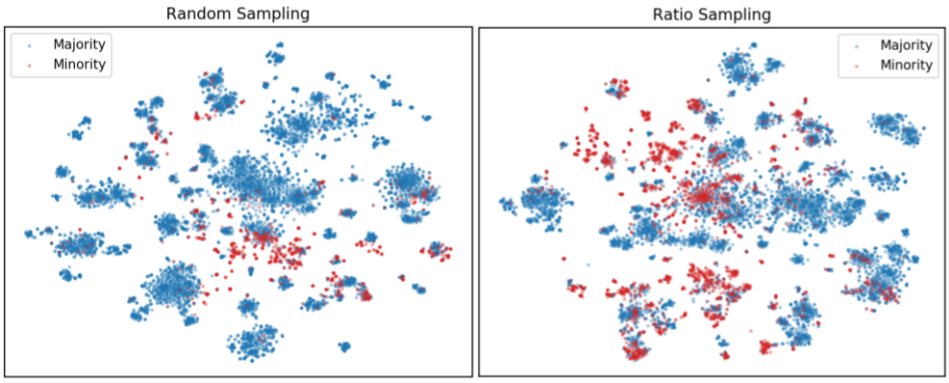


Figure A.11: Document embeddings.

B Chapter 3 Supporting Information

B.1 Supplementary material.

Class frequencies for histology and subsite.

Class frequencies were included as an external file in excel format (see ClassDistributions.xlsx). Full names associated with each class code can be found in [86].

Micro and macro scores for each individual registry.

Score tables were included as an external file in excel format (see IndividualRegistryResults.xlsx)

B.2 Undersampling Results

Table B.1: Undersampling results for the histology task. As in our previous result, this table represents the average of 7 individual results, one for each left-out registry (see 3.4.8). Discarding pathology reports from the top classes diminishes the micro F1 scores to non-permissive levels.

Percentile Threshold	Test Micro	Test Macro	OOD Micro	OOD Macro
50^{th}	0.4528	0.3585	0.4456	0.2979
90^{th}	0.6330	0.4585	0.6222	0.3768
95^{th}	0.6885	0.4539	0.6705	0.3808

Vita

Kevin De Angeli grew up in Buenos Aires, Argentina. He started his career at Del Mar College where he obtained his associate degrees in Mathematics and Computer Science. He then transferred to Texas A&M-Kingsville where he obtained his bachelor's in Mathematics. Through coursework, online courses, and honor projects, Kevin developed a passion for Machine Learning (ML) which led him to pursue his PhD in Data Science and Engineering at the University of Tennessee. He is interested in the implementation and development of efficient, scalable ML solutions that have the potential to improve communities. He also enjoys the interdisciplinary nature of ML, allowing him to interact and work with scientists from diverse backgrounds.