UNIVERSITY OF TRIESTE

MASTER THESIS

Meta-Learning for Medical Image Segmentation

Author:
Rabindra Khadka

Internal Supervisor:
Felice Andrea Pellegrino

External Supervisors:
Pål Halvorsen
Michael Riegler
Steven Hicks
Vajira Thambawita
Debesh Jha

A thesis submitted in fulfillment of the requirements
for the degree of MS Data Science
in collaboration with
SimulaMet,
Department of Holistic Systems, Norway

Declaration of Authorship

I, Rabindra Khadka , declare that this thesis, titled Meta-Learning for Medical Image Segmentation, and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.

Signed:

Date:

Abstract

Human beings learn by building upon the prior experiences. We can learn a task quickly just by observing or experiencing few instances of the task. The quest of building an artificial system that mimics human, needs to be able to learn from prior tasks with few example points. In this thesis, we adopt the learning process based on this concept of learning to learn from prior experience with few shots in medical settings. The meta-learning , unlike the conventional learning from scratch, is based on learning from a number of task that are comprised of N-classes and K-shot instances.

With the aim of adopting the meta-learning process in medical setting specially for polyp segmentation, previous works on the field of meta-learning and polyp segmentation were studied. The terminologies and problem setting for optimization based meta-learning processes were studied and applied to our task of polyp segmentation. The data pipeline for task generation process was constructed to sample tasks from four different sources of polyp images. Unlike prior approaches on polyp segmentation in which models were trained on a large number of data points from one particular data source, this work is based on using the prior learning and adapts to tasks comprised of few images from different sources. The implicit model agnostic meta-learning (iMAML) algorithm was adopted for meta-learn the segmentation problem.

The results show that for the polyp segmentation problem, models meta-learned on tasks with a few shot instances produced better result than the model trained by merging datasets from different sources. However, it was not able to match the results from a model that was trained on a single data source. The results also indicate that by increasing the number of instances and diversity of data class or source in a task, the meta-learner can exhibit better generalization capability. Finally, the thesis is concluded with some remarks and queries that came up during the experiments. The future research directions and the relevancy of meta-learning to improve the learning and generalization process in medical domain is also discussed.

Acknowledgements

I am profoundly grateful for my supervisors in SimulaMet; *Prof. Pal Halvorsen, Michael Riegler, Debesh Jha, Steven Hicks, Vajira Thambawita* for their continuous support, guidance throughout the period; encouraging me to participate in *Medico challenge* and providing me the flexibility to work on novel methods. I would like to also thank for each week's Friday session where I could discuss my problems and have fruitful conclusions. I am also grateful to my university professor *Felice Andrea Pellegrino* for inspiring me towards the field of computer vision. It was amazing time throughout my learning period. I would also like to thank all my class mates for being so co-operative and open.

I acknowledge that the research has benefited from the Experimental Infrastructure for Exploration of Exascale Computing (eX3), which is financially supported by the Research Council of Norway under contract 270053.

Finally, I would like to dedicate this thesis work to my parents Gopal Khadka and Junu Khadka for always being there for me with full love and support.

Contents

List of Figures

List of Tables

Acronyms

AGI Artificial General Intelligence. 6

AI Artificial Intelligence. 1, 41

FOMAML First Order MAML. xiii, 11, 17

ICPR International Conference on Pattern Recognition. 1

iMAML implicit MAML. 2–4, 12, 33, 36, 37, 39

LSTM Long Short Term Memory. 11

MAML Model Agnostic Meta-Learning. 4, 11, 39

ML Machine Learning. 1, 2

Chapter 1

Introduction

1.1 Background

AI refers to the ability of machines to mimic human intelligence and actions. The field of AI has grown significantly in past decades, but displaying creativity and learning capacity like human beings is challenging. Humans possess a great capacity of learning quickly and building on the top of what we have learned before. Humans leverage prior knowledge, concepts and abstract information gained over time to handle new tasks and adapt in new environments. The question widely remains how the complex learning process of humans can be imitated and rivaled by machines.

ML community have mastered the full stack end to end learning; and most of the tasks follow this learning path from scratch as it does not require specific domain knowledge to solve the task, for instances, winning game of 'GO' without domain knowledge (Silver et al., 2017), Ciresan et al. used neural network to win the ICPR contest for cancer detection without any domain knowledge (Cireşan et al., 2013). If we observe carefully how humans learn and build into their prior knowledge, it should not be necessary to design systems that learn from scratch. For instance, a kid can learn to play football without learning explicitly to pick up the ball. Taking this perception into designing neural networks and learning functions, it bring us to the subject of *Meta-Learning* which points out into the direction of achieving general human level intelligence without the requirement of huge numbers of data points. Meta-learning is the idea of learning to learn from various tasks that are composed of instances from different classes.

Training models only with a single dataset containing different types of data examples do not increase the model's adaptability capacity. So, basically, the approach should be transfer of knowledge. Meta-Learning is the learning to learn (Thrun Sebastian, 2012) process which focuses on transferring knowledge from different prior tasks when tackling a new task. It studies how the learner commonly called as bias or prior, effects generalization across different tasks (Vilalta and Drissi, 2002). One of the established field of transferring knowledge is known as $transfer\ learning$ which is a form of meta-learning that leverages prior data while learning new data points. The fine-tuning of pre-trained model on new target dataset has been commercially successful. It has been effectively used for image classification (Donahue et al., 2014), text classification (Raina, Ng, and Koller, 2006) and most

recently transfer learning from the muscles signal when classifying brain-waves signal (Bird et al., 2020). Pre-training is effective but would struggle as the source data domain of the pre-trained network widely differs target data domain and also when there are few training examples (Yosinski et al., 2014). So, learning with few examples (few shot learning) and increasing generalization capacity in various domains are crucial in mimicking human level intelligence.

In this thesis, the idea of transferring of knowledge in an optimal way from prior learning or previously seen tasks and then combining the knowledge with few shot instances has been used to tackle segmentation problem in medical setting. There are different kinds of approach to perform meta-learning. The algorithm adopted for this thesis work is known as iMAML (Aravind Rajeswaran, 2019) which is the latest member in the class of optimization based meta-learning approach.

1.2 Problem Statement

Labeled data is scarce, and it is more so in the medical domain. Therefore, *few shot learning* is getting popular within the ML community as it just requires few training points. Few shot learning methods with meta-learning process have shown promising results in classification task (Chelsea Finn and Levine, 2017), (Nichol, Achiam, and Schulman, 2018), but these methods, for a difficult task like predicting dense output for semantic segmentation, have not been explored much.

To explore the potential opportunity as discussed above, the hypothesis outlined was; *"Meta-Learning with the iMAML algorithm, can be applied to perform medical image segmentation."*

From the hypothesis, the following sub-questions arise:

- Does gradient base meta-learning with implicit gradient iMAML extend to semantic segmentation which is characterized by dense prediction and skewed distributions?
- What is the performance of iMAML in the medical image segmentation context with few shot training examples?
- Can we identify possible advantages and disadvantages?

1.3 Scope and Limitation

The scope of this thesis work is to construct a meta-learning pipeline, train the meta-learner using the compute and memory-efficient iMAML algorithm and evaluate its performance in a medical setting for image segmentation under a few shot scenario. The data come from colonoscopy testing and contain polyps (Jha et al., 2020a). The limitation associated with the datasets is that they do not contain large enough data images to sample the tasks effectively. Moreover, the datasets used do not provide enough diversity between classes

and remains to be tested in future with diverse set of classes. Tasks with few instances are generated from different data sources for training and testing the meta-learner. The UNet architecture (Ronneberger, 2015) guided with an attention mechanism is adapted as the meta-learner. A range of hyper-parameters for the iMAML algorithm was tested by following the iMAML paper (Aravind Rajeswaran, 2019). One of the limitations experienced during empirical study was the lack of enough diversity among the pools of dataset. The other limitation of the iMAML algorithm was the trade-off between compute time and accurate gradient using conjugate gradient. Since no previous work on polyp segmentation has adopted the few shot approach for training their model, a pre-trained UNet on the brain MRI dataset was fine-tuned with the merged polyp datasets to create the baseline [section 4.7].

1.4 Novelty Approach and Relevance

The above mentioned particular approach was chosen to perform polyp segmentation as generating labeled polyp images is an arduous task, and they are rare. Furthermore, there exists data shift such as population shift, acquisition shift and annotation shift [chapter 3] while gathering polyp images which is a hindrance to model's generalization capacity during deployment. The methodology adapted in this thesis will investigate the feasibility of meta-learning with iMAML in the medical image segmentation setting with N way k shot tasks where the efficacy of the method is still unexplored. To the author's best knowledge, this approach has not been adapted till date in polyp segmentation; also the iMAML algorithm has not been tested for the image segmentation problem, especially in medical domain under few shot setting.

1.5 Main Contributions

A novel approach was adopted to tackle the problem of polyp segmentation. The core contribution is the extension of the optimization-based meta-learning iMAML algorithm to segmentation settings in the medical domain. The idea applied was to build upon the prior knowledge by initializing the meta-learner with pre-trained weights. The subset of this work on the concept of knowledge transfer applied in conjunction with attention mechanism was submitted for review to *MediaEval'*20 as one of the challenge papers. The empirical results obtained from the polyp segmentation task under a few-shot supervised setting were compared with the results from prior approaches. The datasets of polyp images from various sources were taken and the custom data pipeline for generating tasks from different data sources was constructed. The experimental evaluations demonstrated that meta-learning from few-shot instances of polyp images generalizes well to unseen tasks in comparison to the model's performance that was trained over merged datasets directly. A decent dice score value of 75.54 % was obtained under

3-way 10-shot task set up which supports the stated hypothesis that meta-larning algortihm iMAML can be applied for image segmentation in medical domain. The results showed an improving trend when the number of shot and tasks were increased. It was shown that the difficult problem like image segmentation with few shot instances can be solved by leveraging the optimization based meta-learning algorithm. One of the observed advantages of the algorithm is that it allows the meta-learner to learn from different tasks under few shot setting with the flexibility of arbitrarily increasing the number of tasks. The collected experimental findings points to the possible direction of improving performance of the meta-learner while comparing with the state of art results for polyp segmentation.

1.6 Thesis Outline

The outline of this thesis are as follows:

- In Chapter 2, we formulate the meta-learning problem, highlight different meta-learning paradigms, discuss the challenges of optimization-based meta-learning MAML and one of the proposed solution called iMAML.
- In Chapter 3, we look over various problems related to data; especially within the medical domain. We also have an overview of the dataset used to carry out the experiments.
- In Chapter 4, we discuss the methods and techniques adapted to implement the segmentation task.
- In Chapter 5, we observe experimental results, discuss and interpret the results .
- Finally, in Chapter 6, we conclude the thesis by discussing some of the noted open challenges, especially within medical image segmentation using meta-learning under few shot setting and about possible future road maps.

Chapter 2

Foundations and Theory

2.1 Artificial Intelligence

The term *artificial intelligence* was coined by a team of researchers including Newell and Simon (Russell, 2003). Artificial intelligence (AI) is the simulation of human intelligence with the help of computers that are programmed to mimic human thinking and actions. AI is also defined as the field that studies intelligent agents (Russell, 2003). Typically, the agent in the system becomes aware of the environment and takes action that maximizes the chance of success. The agent learns from experience, is flexible with a shift in environment, goals; and makes reasonable choice conditioned on limitations (Mackworth, 2017).

In the early phase, AI was based on logic-based or *symbolism* (symbolists), for instance, if it looks like a duck, swims like a duck, and quacks like a duck, then it is a duck logically. The *second* approach of AI is Bayesian inference for instance if it looks like a duck, swims like a duck, and quacks like a duck then the probability of it being a duck can be adjusted. The *third* approach is popular and also known as analogizers such as SVM (Vapnik, 1995), K-Nearest Neighbor algorithms (Altman, 1992). For example, based on previous records of how ducks look like, the way they swim, quack and walk, the current animal can be classified as a duck. The fourth approach is the connectionist way where models are created based on how brain neurons are connected. The connections refer to the strength of signals and the number of signals. They learn by comparing the output with the desired target and alter the connections accordingly (Selmer Bringsjord, 2018).

Currently, most of the tractable AI work can be categorized as *narrow AI* which can effectively show intelligence action in only one specialized area such as medical diagnosis, recommendation, or autonomous driving. However, the artificial intelligence field originally wanted to develop a system that can be applied to a variety of complex problems known as *artificial general intelligence (AGI)* (Pennachin and Goertzel, 1992). Recent developments in the field of AI have pointed towards the direction of AGI. For instance, *Deepmind* developed a generalized intelligence system that could learn many types of Atari games (Mnih, 2015). One of the fields that are currently popular in the AI community is meta-learning which produces a flexible AI model that can learn from the various task with few data points. This way of learning to learn has shown some promising results and rightly points towards

the direction of attaining AGI.

2.1.1 Machine Learning

Machine learning (ML) is a subset of AI that allows computer programs to learn from experience. It learns from data concerning some task which can be difficult to solve by writing programs. The performance of a machine learning algorithm must be measured at the end. This is achieved through various metrics such as *accuracy* which is measured as the ratio of correct output to the total examples. *Error rate* also gives the same information which is the ratio of incorrect output to the total examples. The performance is measured with the never seen *test set* after training the ML model with *train set* so that the model's generalization capacity can be observed.

There are various other metrics that particularly fit specific task nature. For instance, segmentation of objects in an image requires metrics such as intersection over union or dice score to measure the performance of the model. So, sometimes it will not be straightforward to choose a suitable metric to measure the performance of the model.

Machine Learning algorithms can be broadly divided into *supervised* and *unsupervised* algorithms. *Supervised learning algorithm* takes the dataset which is comprised of both the input (X) and its label or target Y, then learns to predict Y from X. The performance check is done by computing the loss function which gives the cost or difference between the model's prediction ($\hat{Y}$) and the true label (Y).

Unsupervised learning algorithm takes in a dataset with many features then learn the properties of the structure in the dataset on its own. For instance, while clustering unlabeled data, the dataset is divide into clusters of similar examples. This type of learning is also used in density estimation for finding probability density function. (Goodfellow, Bengio, and Courville, 2016)

2.1.2 Deep learning

Deep Learning is a subset of machine learning algorithm that uses multiple layers which gradually extracts higher level features from input. With the development in the field of deep learning, now artificial neural networks mainly characterize deep learning. The goal of the network is to approximate some function f, for e.g $y = f(X)$ takes input x and maps to a domain category y. The network then approximates $y = f(x, \theta)$ with some parameter θ. The networks are known as neural as they are connected like neurons in our brain. Each layer is comprised of neurons that operate in parallel fashion and follow the principle "neurons those are wired together, fire together".

One of the advantages of training a neural network is that due to its non-linearity, it causes most of the loss function to become non-convex. This is why the neural network is trained by iteratively using a gradient-based optimizer. This way of training brings the cost function to a low value rather than a global convergence. So, the stochastic gradient descent is sensitive to the

values of initial parameters and it is a good practice to initialize the neural network.

The cost function is similar to that of parametric models. Generally, cross-entropy is taken between training input and the model's output for the cost function. The cost function is combined with regularization terms such as the weight decay approach commonly known as $l2$ *regularization*, $L1$ *regularization*. Other forms of regularization can be applied while training a neural network, to name some of them, such as *early stopping*, *parameter sharing*, *drop out*, *data augmentation*. (Goodfellow, Bengio, and Courville, 2016)

2.1.3 Capacity and Generalization

The core objective of machine learning is to increase the generalization capacity by performing well on unseen data points. During the training of machine learning, training error is computed based on the training set and the objective would be to reduce the training error which is the optimization problem. Further, during the test phase, generalization error is computed which is obtained by computing test error based on the test set. The quest in machine learning is to lower the test error by observing the training error. This is supported by the assumption that training and test points are identically distributed drawn from the same probability distribution. The average test error is greater than or equal to the average training error. These two error factors will determine how good a machine learning model will perform.

$Underfitting$ is the scenario when the model does not obtain a sufficiently low training error. $Overfitting$ occurs when the gap between train and test error is too big. The underfitting and overfitting can be tuned by changing the model's *capacity*. Capacity refers to the number of functions a machine learning model can select as a possible solution to a given task. So, the model with low capacity will tend to underfit and the model with high capacity can overfit and do not perform well on the test set. So, capacity should be proportional to the complexity of the task and the difficulty of training data. For instance the linear regression algorithm's ($\hat{y} = b + wx$) capacity is increased by including polynomials ($\hat{y} = b + w_1 x + w_2 x^2$). Here, by adding x^2, a quadratic model can be learned.

Capacity is also defined by the choice of functions from a particular set of functions which is known as the *representational capacity* of the model. Vapnik-Chervonenkis (VC) dimension (Vapnik, 1995) a popular theory helps to measure the model's capacity. It is defined in the context of binary classifier where it is the number of the largest set of x points that the algorithm can shatter. The theory suggests that the difference between training and generalization error is upper bounded by some quantity that increases as the model capacity expands but will decrease as the number of training points increases. This theory reflects well in machine learning algorithms but practically not imposed in deep learning algorithm as the capacity of deep learning models is hard to estimate because of the constrained posed by optimization algorithm. (Goodfellow, Bengio, and Courville, 2016)

2.1.4 Regularization

Regularization is the method to add information to prevent overfitting or solve an ill-posed problem. The regularization term adds penalty or cost to the optimization function and aims to reduce the generalization error but not the training error. Learning by a model is not only influenced by the number of the set of functions in some hypothesis space but also by being able to choose the kinds of functions that can be used to output solutions. This choosing of functions from hypothesis space can be done with the help of the regularization technique. A regularization term can be added to the loss function as in equation 2.1.

$$J(w) = \sum_{i=1}^{n} L(f(x_i), y_i) + \lambda w^T w \tag{2.1}$$

where L is the loss function that quantifies the cost of prediction, λ is the regularization parameter that allows controlling how small the weight W can be made, and $w^T w$ being the regularizer. If chosen λ value to be larger, the weights become smaller. So, when the cost $(J(w)$ is minimized, it chooses weight such that it trades off between fitting training data and having small weight i.e the scenario of the solution with small slope or fewer features being weighted. There is no particular best method to choose for regularization but depends upon the nature of the task. (Goodfellow, Bengio, and Courville, 2016)

2.1.5 Convolutional Neural Network

Convolutional neural networks(CNN) (LeCun, 1998) are a type of neural network that uses convolution operation instead of matrix multiplication in at least one of the layers. Mathematically , convolution is represented by

$$s(t) = \int x(a)w(t-a)da \tag{2.2}$$

and the convolutional operation is denoted by asterisk(*) as $s(t) = (x * w)(t)$. In the context of CNN, x is the input and the w is referred to as kernel. The output of the convolution is then called as $feature\ map$. In practice, x is a multi-dimensional array of data and the kernel will be multidimensional arrays of parameters.

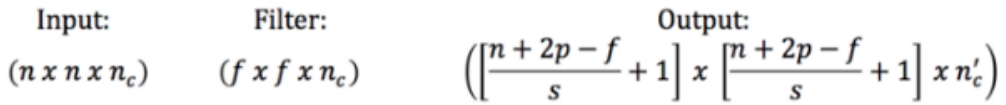

FIGURE 2.1: The general output size after applying $n_c \times f \times f$ filters where n_c is the number of channels, p is number of padding, s number of strides and f is filter or kernel size.

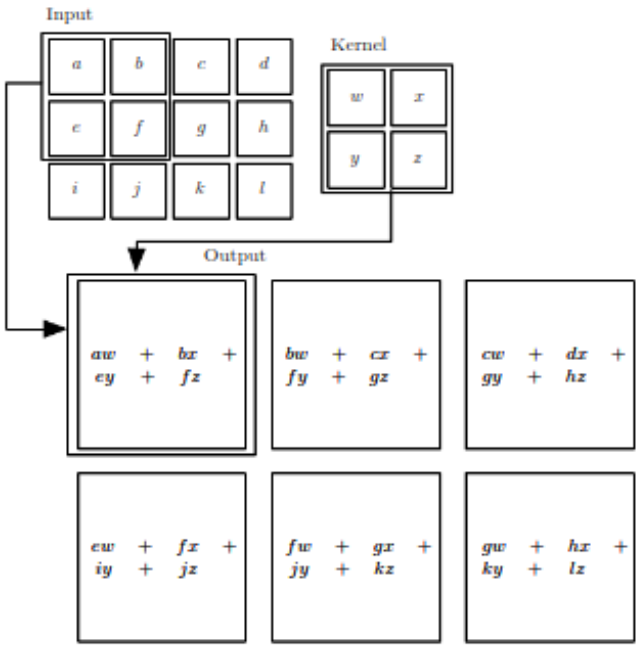

FIGURE 2.2: Showing Kernel operation over input image and its output. (Goodfellow, Bengio, and Courville, 2016)

CNN has three significant features namely *sparse interactions, parameter sharing, and equivariant respresentation*.

Sparse interaction facilitates kernels to recognize meaningful features such as edges from a large input's pixel space. This makes CNN memory efficient and fast.

Parameter sharing is the idea that a feature detector(e.g vertical edge detector) useful in one location of an image is also going to be useful in another location of the image. In CNN, each element of the kernel is applied to all the locations of elements in the input. It helps to reduce memory usage by storing only k parameters which is much smaller than the input dimension.

Equivariance to translation is enjoyed by CNN because of the parameter sharing property. This means if the input is shifted by some function f then the kernel is invariant to that shift. For example, if an object is moved in an input image, it will be moved with the same amount in the output. However, convolution is not equivariant to other transformations such as scale and rotation of an image.

Typically convolutional network contains three stages. The *first layer* performs multi convolutions in parallel that outputs a set of linear activations. These are passed to *second stage* where non-linear functions such as rectified linear activation function(RELU) are applied which acts as a detector. Then the pooling layer is applied which replaces the output at a location with summary statistics of the neighborhood elements. Pooling helps to keep the representation invariant to small translations at the input. It also preserves the location of a feature in the image.

Convolution network has the advantage point that it can process inputs of varying dimensions. For example, a dataset with an image of different width

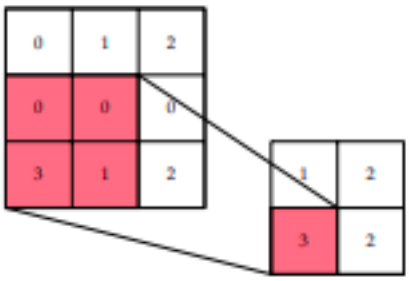

FIGURE 2.3: Schema of max-pooling operation where the maximum number in the window is selected.

and height are taken by CNN layer and kernel can be applied for a different number of times corresponding to the input's height and weight. It can also take in 1-D single channel data like audio waveform, 2-D images, and 3-D volumetric data such volumes from medical imaging; CT scans, brain MRI.

CNN is a good representation of neuroscientific principles that deep learning has adopted. It is also known as the best representation of biologically inspired intelligence. In the study of how mammalian vision system functions (Lienhard and Wiesel, 1964), the activity of individual neurons were observed in cats and found that the neurons in the early vision system responded strongly to certain patterns of light but not to others. This research became the guiding path to the development of neural networks.

2.1.6 AI privacy in Healthcare

With the advent of artificial intelligence (AI), the field of healthcare is experiencing paradigm shift with the integration of machine learning models for actions like making clinical decisions, diagnosing diseases, personalized medicine and others (Dilsizian, 2014; Amato F, 2013). Going by the phrase, *"with power comes responsibility"*, AI usage should be bounded by ethical policies as it also poses threat to patient's privacy, safety and preference.

Some of the major ethical questions regarding the application of AI in healthcare comes from privacy ethics. Health data, unlike other data is highly sensitive and is under regulation. Therefore , it is not easily available for usage. Another reason the data in healthcare is not highly shareable is because of the high cost involved in data generation process which carries significant commercial value too. There have been solutions offered such as data anonymisation but study have proven that it does not do enough to protect privacy (L. Rocher, 2019). There is another effective concept known as federated learning (B. McMahan and Arcas, 2017) which tackles the problem of privacy by allowing to learn from data without having to see the data by collaborating with data owner. In this concept, the data do not move and only the model is moved to different data locations that can learn from the data in each location. This method holds great potential to help address the privacy issue in medical field while applying machine learning models to improve efficiency in healthcare service.

2.2 Literature Review

Few shot learning methods can be broadly classified into two categories. One method is to take the approach of data augmentation and the other would be the task-based meta-learning method. In the current research context, meta-learning for few-shot learning can be broadly divided into three sub classes. One of the subclasses is the *non-parametric* based method where models learn to compare between training and test data examples based on distance metric, examples are Siamese network (Koch, Zemel, and Salakhutdinov, 2015), Matching networks (Vinyals et al., 2016). The second subclass can be *black box* method which can use the concept of memory like work done by Rave et al. (Sachin Ravi, 2017) where they used LSTM based meta learner model to learn the exact optimization algorithm to train another artificial learner with few shot examples. The other black box approach called SNAIL by (Nikhil Mishra, 2018) is one of the robust examples of black-box meta-learning where instead of yielding all the task-specific parameters ϕ_i, it outputs a low dimensional vector which is then used along with meta-parameters θ for predictions. The third subclass can be the *optimization* based method that looks for the best parameter that can adapt across all the task, example for it is MAML algorithm by (Chelsea Finn and Levine, 2017) which learns a good initialization for base-learner for solving new tasks. In this method, the learner uses gradient-based learning and can generalize well even when it receives out of sample data. It generalizes better than memory-based meta-learning using LSTM. However, MAML has some limitations such as avoiding overfitting it uses a shallow network and is unable to use deep powerful architecture. It also requires many similar tasks for meta-learning which can be costly (Qianru Sun, 2019). Similarly, the other gradient-based method is called Reptile which is based on first-order gradient information and is closely related with FOMAML which ignores second-order gradient and not so accurate like MAML (Nichol, Achiam, and Schulman, 2018).

Some of the prior approaches using gradient-based meta-learning for image segmentation in few shot scenario includes like the work by (Sean M. Hendryx, 2019) where they directly applied the first-order MAML (Chelsea Finn and Levine, 2017) on FSS-1000 dataset (Tianhan Wei and Tang, 2018), Qi Dou et .al (Qi Dou, 2019) applied MAML algorithm and the idea is to operate in semantic feature space with the aim of domain generalization. They showed consistent improvement also with brain MRI images. The key contribution of the paper is two complementary loss function which regularizes the semantic structure in feature space by using episodic learning process of MAML and it tries to learn semantically invariant features across training domains. Some of the other approaches besides gradient-based meta-learning in few shot scenario for image segmentation includes the use of attention and multi-context guiding network consisting of the query, support, and feature fusion branch (Tao Hu, 2019), Wang et. al (Kaixin Wang, 2019) used k-way semantic segmentation which performs segmentation through matching each pixel to the learned prototypes for each semantic class.

2.2.1 Existing Approaches for Polyp Segmentation

Polyps are protruding tissue that can grow abnormally in the gastrointestinal tract and result in colorectal cancer (Ann Pietrangelo, 2018). Acknowledging the importance of early detection of polyps while screening, the area has been an active topic of research. Traditionally, several manual techniques to extract polyp features such as color, shape, appearances have been used to train a classifier to identify polyps from its background. With the advent of deep learning models, the polyp segmentation problem has been approached by learning polyp and its mask. Some of the work in this domain using AI includes a region-based approach using CNN for polyp detection in images and videos (DY. Shin and Balasingham, 2018), they also applied GAN (I. Goodfellow and Bengio, 2014) to generate polyp images for improving their model performance. To improve real-time detection and localization, YOLO-v2 (J. Redmon and Farhadi, 2016) was also applied in the work by (Lee, 2020) which showed a good real-time performance. Recently there has been some enhancement to the polyp segmentation approach. One of the notable work was Pranet (Deng-Ping et al., 2020) which used a reverse attention mechanism to model the boundaries of polyps. The model was complex and claims to have achieved improved state of art results on various polyp datasets. Another notable work was ResUNet++ (Jha et al., 2019) which had residual blocks, squeeze and excitation blocks, Atrous Spatial Pyramidal Pooling (ASPP), and attention blocks. The results showed robust performance with various datasets. They also proposed DoubleUNet (D. Jha and Johansen, 2020) with two UNet stacked on top of each other that achieved state-of-art performance on CVC-Clinic and ETIS-Larib polyp datasets. There are also work to improve segmentation precision, for example, work based on boundary-aware network (BA-Net)for polyp segmentation by (R. Wang and Li, 2020). It was based on an encoder-decoder network that was able to capture high-level features while preserving the spatial information.

All of these above mentioned approaches require a large number of training examples and do not show a good generalization ability when faced with a new testing environment. In contrast to these prior works, our work is based on the concept of few-shot learning that uses a gradient-based meta-learning algorithm called *Implicit model agnostic meta learning* iMAML (Aravind Rajeswaran, 2019), which is a memory and compute efficient algorithm in comparison to other optimization-based meta-learning algorithms, particularly .

2.3 General Problem Formulation for Meta-Learning

In this section, we draw a overview picture of meta-learning and formalize the problem. Meta-learning is characterized by a meta-learner which gets trained episodically on a set of tasks so that the model can adapt to new and unseen tasks by using only a few training examples.

Meta-learning can be viewed in two ways namely deterministic and probabilistic view point. The probabilistic view is based on the Bayesian inference

approach. The deterministic viewpoint simply takes a set of training data D_train, test data and meta parameters θ as input to a function to produce a target label.

The first paper that discussed the formulation of a possible meta-learning set up was (Sachin Ravi, 2017). The problem was framed as maximizing the likelihood of model parameters ϕ given a task and other meta training data.

$$argmax\ logp(\phi|D, D_{meta-train}) \tag{2.3}$$

where $D_{meta-train}$ is a set of data-sets for different chosen tasks D_1,D_2,..,D_n. They then introduced a set of meta parameters θ which carries information about various tasks for solving a new task where $\theta = p(\theta|D_{meta-train})$. Then the likelihood of the parameters of the original data given the meta-training data can be expressed as in equation 2.2 which is the integral over all the meta-parameters θ.

$$logp(\phi|D, D_{meta-train}) = log \int_{\theta} p(\phi|D, \theta p(\phi|D_{meta-train}) d\theta \tag{2.4}$$

The above equation 2.2 can be further approximated using point estimate for the parameters.

$$\approx\ log\ p(\phi|D, \theta^*) + log\ p(\theta^*|D_{meta_train}) \tag{2.5}$$

where; $p(\phi|D, \theta^*)$ is the *adaptation phase* that takes into account the task specific parameters *phi* or a new task given its access to data (D) and meta parameters θ, $p(\theta^*|D_{meta_train})$ is the meta training phase that takes into account a set of meta parameters θ given that it has seen the meta-training data D_{meta_train}. This way, the meta-learning domain can be modular with two phase process namely *adaptation phase* and *meta-training phase*.

2.4 Loss Optimization

The meta-training data is comprised of pairs of training and test set for each task. So, there will be k feature-label pairs in the training set D_i^{train} and m feature-label pairs in the test set D_i^{test}.

$$D_{meta_train} = D_i^{train} = (x_1^i, y_1^i), ...(x_k^i, y_k^i);\ \ D_i^{test} = (x_1^i, y_1^i), ...(x_m^i, y_m^i) \tag{2.6}$$

In the adaptation phase, a function f_θ takes the training set D^{train} as input and returns ϕ^*, a set of task related parameters. So,in nutshell a set of meta parameters θ is learned so that a good ϕ_i= $f_\theta(D_i^{train}$ can be estimated that does well against the test set D_i^{test}.

In the meta-learning phase, the probability of task parameters ϕ is maximized given the test data points D_i^{test} such that a good set of meta-parameters θ^* can be estimated as suggested in the equation below.

$$\theta^* = max_\theta \sum_{i=1}^{n} log\ p(\phi_i|D_i^{test}) \tag{2.7}$$

So, the loss is optimized by the learner by finding a set of task parameters that does well against the test points and the gradient of the loss is with respect to the parameters θ. This process is further discussed below where we describe and formulate the optimization-based meta-learning method.

2.5 Optimization Based Meta-learning

We follow the path of optimization-based meta-learning as formulated by Finn et.al (Chelsea Finn and Levine, 2017) to tackle our task of meta-learning segmentation.

In section 2.1 above, the adaptation phase was described by the distribution of task-specific parameters ϕ_i as $p(\phi_i|D_i^{train}, \theta)$. In the optimization-based method, this distribution of ϕ is treated as an optimization procedure so that the process of getting task-specific parameters ϕ_i can be optimized. The meta-learning process can be formalized as below. The first part indicates the maximizing the likelihood of training data given task-specific parameters and the second part indicates the maximization of task-specific parameters given the meta-parameters.

$$max\ log\ p(D_i^{train}|\phi_i) + log\ p(\phi_i|\theta) \tag{2.8}$$

The meta parameters θ are pre-trained and fine tuned while testing which can be represented by the equation below which uses gradient descent for optimization with learning rate α.

$$\phi \leftarrow \theta - \alpha \nabla_\theta L(\theta, D^{train}) \tag{2.9}$$

In the above equation 2.7, the parameter θ can be extracted from pretrained network trained with large dataset such as image net or other large dataset sources and this becomes the reason when the meta-learning method becomes ineffective with small amount of training data. This problem was tackled by $Model\ Agnostic\ Meta-learning\ (MAML)$ by Finn et.al (Chelsea Finn and Levine, 2017) where they modified the above loss function in equation 2.7; which then only takes into account the best task specific parameters ϕ with respect to test data points across all the tasks such that the test loss of the task will be minimized; $L(\phi, D_i^{test})$.

$$min_\theta \sum_{task_i} L(\theta - \alpha \nabla_\theta L(\theta, D_i^{train}, D_i^{test}) \tag{2.10}$$

So, in nutshell, there is a bi-level meta-learning setting where a space of algorithms is considered that compute task-specific parameters with the help of meta-parameters θ and training dataset from the task which can be written as $\phi_i = Alg(\theta, D_i^{train})$ for task T_i. The key idea of meta-learning is to

learn meta-parameters θ that can yield effective task-specific parameters after adaptation which is shown in equation 2.9 (Aravind Rajeswaran, 2019).

$$\overbrace{\theta^*_{ML} :\equiv argmin_\theta\, F(\theta)}^{outer-level}, where F(\theta) = \frac{1}{M}\sum_{i=1}^{M} L(\overbrace{Alg(\theta, D_i^{train})}^{inner-level}, D_i^{test}) \quad (2.11)$$

This is viewed as a bi-level optimization problem as computing the task-specific parameters $Alg(\theta, Dtrain_i$ is interpreted as solving an inner level optimization problem that can be solved explicitly or implicitly. In the MAML paper (Chelsea Finn and Levine, 2017), the inner level optimization uses one or multiple steps of gradient descent with the parameters initialized by θ. For instance, equation 2.7 is considered as one step of gradient descent. Then for solving the outer level with gradient descent methods, differentiation is applied to Alg, and in MAML back-propagation is used through k steps of gradient descent. MAML authors (Chelsea Finn and Levine, 2017) have experimented MAML algorithm widely to test its effectiveness and universality which shows that it can approximate any function; it also covers well as the black box algorithms.

2.6 Challenges of MAML

One of the noticeable challenges with the MAML algorithm is that it needs a very deep neural architecture so that it gets a good inner level gradient update. This challenge of searching for a deep good architecture was tackled by Kim et.al (Jaehong Kim, 2018) by proposing automating machine learning pipelines named auto-meta. They automatically found the architecture with deep and thin layers which achieved the state-of-art results on a five-shot five-way mini ImageNet classification problem.

The unreliability with bi-level optimization ideas in MAML also brings the stability issue with it. There are various optimization methods proposed to tackle this problem which has proven to converge faster with improved accuracy. For instance, Li et al proposed Meta-SGD (Zhenguo Li, 2017), an SGD like trainable meta-learner that can initialize parameters and adapt any differentiable learner in one step of the iteration. It can also learn the learning rate and update the direction of the learner all in a single end to end meta-learning process. There have also been other work such as MAML++ by Antoniou et al (Antreas Antoniou, 2019) where they proposed methods to lower the generalization error and hyperparameter sensitivity to improve the MAML stability during training.

Thirdly, MAML also suffers due to computationally expensive back-propagation which increases with the number of inner gradient steps. It uses the second derivative when back-propagating the meta-gradient through the gradient operator in the meta objective function in equation 2.9. One of the approaches proposed to tackle this problem is by Finn et. al (Chelsea Finn and Levine,

2017) where they dropped the back-propagation and used first-order approximation by approximating $\frac{d\phi}{d\theta}$ as an identity function. The other approach was proposed by Nichol et.al (Nichol, Achiam, and Schulman, 2018) named as REPTILE where they repeatedly sampled the task, trained, and moved the initialization towards the trained weights. These first-order meta-learning methods which avoided the second-order derivatives also performed well and gave results close enough to the result from the original MAML. The other method is called ; proposed by Rajeswaran et.al (Aravind Rajeswaran, 2019) in which they devised a theorem to compute the meta gradient $\frac{d\phi}{d\theta}$ implicitly which depends only on the solution to the inner level optimization but not the number of inner gradient steps.

2.7 Implicit MAML (iMAML)

Optimization-based meta-learning, though being an effective way to solve a few shot learning problems, has some caveats. One of the challenges is that it becomes computationally expensive while scaling up with tasks containing a large number of examples as it needs to differentiate through the inner level. This problem was addressed by Aravind Rajeswaran, 2019 with the concept of implicit differentiation.

The algorithm learns a set of parameters with which an optimization algorithm can be initialized and regularized to this parameter vector which in return will have a good generalization capacity across different tasks. In other words, the meta gradient(outer level) will only depend upon the output of the inner optimization and not the number of optimization steps or paths taken by the inner level optimization problem.

The core part of is the proximal regularization term in the equation 2.12 below.

$$Alg^*(\theta, D_i^{train}) = \underset{\phi' \in \phi}{argmin}\ L(\phi', D_i^{train}) + \frac{\lambda}{2}||\phi' - \theta||^2 \tag{2.12}$$

In the algorithm, only a few gradient steps are used which acts as early stopping and can be taken as regularization. Based on this idea of early stopping as regularization, adds a regularization term that allows the algorithm to learn more without over-fitting. The regularization term will keep the model parameters ϕ_i close to the meta parameter θ and maintains strong dependence. The regularization parameter λ can be treated as scalar vector and acts as the learning rate similar to α in MAML which controls the influence of θ concerning different training task D_i^{train}.

So, substituting the input of the loss function in the inner level of equation 2.9 by equation 2.10, it can be written as :

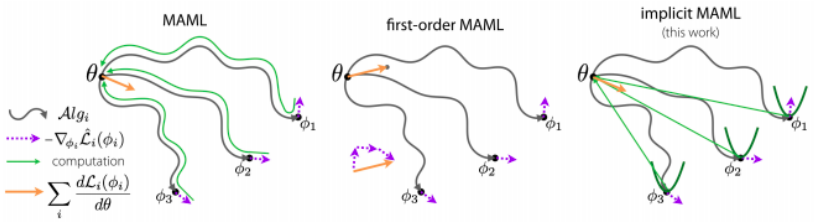

FIGURE 2.4: In MAML algorithm, the meta gradient is computed by going through the optimization path highlighted in green. The FOMAML algorithm computes the meta gradient by approximating $\frac{d\phi}{d\theta}$. In , the meta gradient is obtained without differentiating through SGD steps simply by considering the curvature in the loss space.

$$\overbrace{\theta^*_{ML} :\equiv argmin_\theta\ F(\theta)}^{outer-level}, where F(\theta) = \frac{1}{M}\sum_{i=1}^{M} L(\overbrace{Alg(\theta, D_i^{train})}^{inner-level}, D_i^{test}),\ and$$

$$Alg_i^*(\theta) := \underset{\phi' \in \theta}{argmin}\ G_i(\phi, \theta),\ where\ G_i(\phi^i, \theta) =\ L_i(\phi^{'}) + \frac{\lambda}{2}||\phi^{'} - \theta||^2 \quad (2.13)$$

The goal is to get the best meta parameter θ by minimizing function $F(\theta)$ where $F(\theta)$ is the average validation loss on the parameters found by inner optimization procedure across different tasks. The inner optimization algorithm Alg^* minimizes the function $G_i(\phi^i, \theta)$ which is the training loss of the task parameter plus the regularization term.

The bi-level meta learning problem in equation 2.13 can be solved using an iterative gradient based algorithm of the form $\theta \leftarrow \theta - \eta\, d\theta F(\theta)$.This can be expanded by chain rule as

$$\theta \leftarrow \theta - \eta \frac{1}{M}\sum_{i=1}^{M} \frac{d\ Alg_i^*(\theta)}{d\theta}\ \nabla_\phi\ L_i(Alg_i^*(\theta)) \quad (2.14)$$

The equation 2.12 is the entire gradient descent optimization procedure which describes how the initial meta parameter θ be changed concerning the end parameter ϕ. The main idea of the paper was to compute $\frac{d\ Alg_i^*(\theta)}{d\theta}$ which is a Jacobian matrix given by the equation 2.15.

$$\frac{d\ Alg_i^*(\theta)}{d\theta} = (I + \frac{1}{\lambda}\nabla^2_\phi \hat{L}_i(\phi))^{-1} \quad (2.15)$$

The above equation 2.15 is the inverse of the Jacobian matrix that contains the identity function, learning parameter, and hessian matrix of the training

loss across tasks. The Hessian matrix represents the curvature in the landscape of the loss as seen in figure 2.4. The main point to be noted in this equation is that there is no any SGD process to be performed through being part of the SGD process.

2.8 Summary

Generally learning algorithms can perform very well in one domain but not in another test domain. Meta-Learning is the process that will allow the learning algorithms to learn a new task by leveraging prior knowledge obtained by training on a few shot examples from N classes. There are several approaches to solve the meta-learning task. Based on properties like expressive power, stability, positive inductive bias, and uncertainty awareness, an optimization-based meta-learning algorithm has been chosen for this thesis work. The algorithm has been implemented for meta-learning polyp segmentation tasks due to superior memory and computational efficiency over the original algorithm supported by the fact that unlike , does not have to back propagate through the optimization path.

Chapter 3

Data

3.1 Data challenges in medical domain

Advances in machine learning have led to great improvement in predictive accuracy for many applications like identification of cancer at early stages, identifying potential clinical trials, real-time monitoring of the patient, outbreak prediction, and many more. Having this great potential of AI usage in the medical setting, there remain various challenges to translate the ideas and success to broader clinical practice. The key challenges can be broadly divided into (i). *datascarcity* due to low availability of good quality training data needed for ML algorithms (ii). *data mismatch* which represents the model failing to translate its lab performance to real-world clinical environment (Daniel C. Castro, 2019). *Data mismatch* problem has been further categorized as (a). *selectionbias* (b). $population\ shift$ and (c). $prevalence\ shift$. In the work of Daniel C. Castro, 2019, they have highlighted the causal relationships found between the specification of the input images and the targets, and its significance to tackle data challenges in the medical domain.

The scarcity of labeled data in the medical domain is very noticeable as it is costly to get annotated data by experts which can also involve lab testing. Some of the techniques that provide alternative paths to overcome data scarcity challenges are using abundantly available unlabelled data to semi-supervised learning process (Chapelle O., 2009) and data augmentation. Recently researches have also shown techniques that can learn optimum transformations using unlabelled data. This line of research nicely complements semi-supervised learning to realize its potential and also improves the application of the data augmentation process (Chaitanya, 2019).

The other challenge is the **data mismatch** during training and test time/deployment time which hugely affects the generalization capacity of ML models, particularly in a medical setting. The mismatch between data distributions can be described by **population/dataset shift** and **selection bias**. **Dataset mismatch due to dataset shift** can occur due to some outlier or exogenous process such as highly variant data acquisition processes, a dissimilar bunch of cases. The causality diagram in figure 3.1 captures the overall view of the dataset shift. D is the domain variable which indicates the train or test domain, X is the input images, Y is the target variable and Z is the variable that represents the unobserved true anatomy. So, depending on

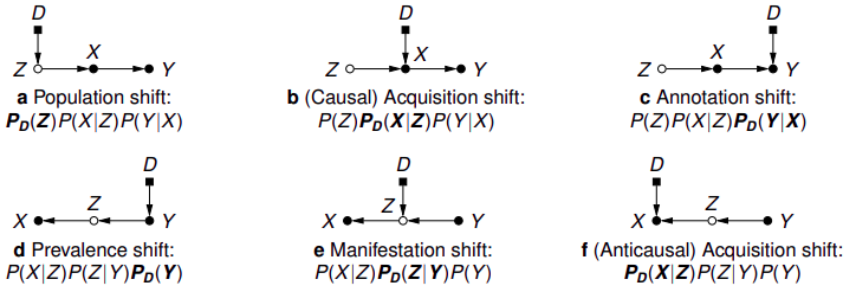

FIGURE 3.1: Representing different data shift conditions in causal and anti-causal direction. The different disentangling of joint distribution between X,Y,Z across train or test domain(D) give rise to different dataset shift (Daniel C. Castro, 2019)

the casual direction for making prediction and the change of joint distribution between variables in different domains, data shift is classified into *population shift*, *annotation shift*, *prevalence shift*, *acquisition shift* and *manifestation shift* (Nikhil Mishra, 2018). These shifts in general machine learning practice are known as *covariate shift*, *concept shift*, *target shift*, *domain shift* and *conditional shift* respectively (Chaitanya, 2019). The instances for various kinds of data shift has been tabulated in table 3.2

Type	Direction	Change	Examples of differences
Population shift	causal	$P_D(Z)$	ages, sexes, diets, habits, ethnicities, genetics
Annotation shift	causal	$P_D(Y\|X)$	annotation policy, annotator experience
Prevalence shift	anticausal	$P_D(Y)$	case-control balance, target selection
Manifestation shift	anticausal	$P_D(Z\|Y)$	anatomical manifestation of the target disease or trait
Acquisition shift	either	$P_D(X\|Z)$	scanner, resolution, contrast, modality, protocol

FIGURE 3.2: Categories of data shift (Daniel C. Castro, 2019)

Population shift takes into account the change of demographics of the population under study which means the distribution of subjects' anatomy during the training phase is different than the test phase $(P_{tr}(Z)! = P_{te}(Z)$. Generally, the population shift does not have a bigger impact on model prediction i.e the model transports well when applied to a new population (Judea Pearl, 2014). The training distribution should have covered a range of variation in the test distribution for the model to perform well in this data shift scenario, however, the model performance is not guaranteed to translate well for different possible variations in test distribution.

Prevalence shift is the case observed in an anti-casual task where the model is trained in balanced data set and tested in a highly imbalanced population. This shift can occur due to a change in environmental factors and is frequently observed in studies like epidemiology where test class distribution is known a priori.

In the case of *annotationshift* the class definition may vary in the train and test domain. This leads to having same data being labelled to different classes in different domain $P_tr(Y|x)! = P_te(Y|X)$. One of the instances where this shift may occur is in projects that involve multiple health institutes. These multiple sites could have their annotation guidelines or personnel with different expertise levels. So, in this scenario models under-perform in a test setting where assumptions and class definitions are different from the training phase. The correction of annotation shift can be time consuming that demands re-annotating and re-calibration of labels.

Manifestation shift happens in cases such as disease manifests differently in the anatomy within the train and test domain. Predictive ML models find it difficult to deal with this kind of data shift. The correction steps require strong parametric assumptions.

Acquisition shift occurs at the time of yielding training images. This shift can happen due to the use of different scanners and imaging guidelines. Some of the ways to mitigate this issue are techniques like intensity normalization, spatial alignment, resampling to a common resolution, and others. There are also some complex methods like synthesizing data and extracting domain invariant features which can eliminate the acquisition shift issue.

The understanding of the characteristics associated with various types of data-set mismatch is useful for diagnosing problems during deployment of the model, identifying the right scenario to use the proposed model, and communicate various data shift issues efficiently.

Selection bias is another process that can create data mismatch. This bias refers to the data gathering process rather than the data generating process which creates bias data. It takes into account how the selection was made from a data population. The causal diagram can be used to depict different kinds of bias as in figure 3.3

FIGURE 3.3: Causal diagrams depicting various types of selection bias:*a.Random b.image dependent c.target dependent d.joint dependent*

Type	Causation	Examples of selection processes	Resulting bias
Random	none	uniform subsampling, randomized trial	none
Image	$X \rightarrow S$	visual phenotype selection (e.g. anatomical traits, lesions)	population shift
		image quality control (QC; e.g. noise, low contrast, artefacts)	acquisition shift
Target	$Y \rightarrow S$	hospital admission, filtering by disease, annotation QC, learning strategies (e.g. class balancing, patch selection)	prevalence shift
Joint	$X \rightarrow S \leftarrow Y$	combination of the above (e.g. curated benchmark dataset)	spurious assoc.

FIGURE 3.4: Sample selection types with examples

Random sample selection is the process where data is sampled randomly with the assumption that the training sample covers the target population. The selection variable S is ignored in this process and this is the ideal case. However, in the real world, there will be some kind of bias injected while gathering data and careful experimental design is required.

Image-dependent sample selection depends upon the image. Images can be selected based upon the anatomical features. It also takes into account the quality of the image like the amount of noise, contrast, artifacts. This can lead to population shift and acquisition shift as tabulated in table 3.4.

Target-dependent selection is where based upon disease labels, annotation, or both; the decision is taken to include the image data. This results in a prevalence shift. The target-dependent bias arises from hospital admissions, criteria adopted in clinical trials, or annotation quality. Also to be noted here is that ML practitioners with their adapted training paths such as picking up only those patches with polyps or class re-balancing method; can give rise to this selection bias.

Jointly-dependent bias arises when the selection variable Z is affected by both image(X) and target(Y) jointly. This leads to a spurious association between data and can greatly affect the generalization capability of the model. Causal reasoning with the help of a causal diagram can help practitioners to scrutinize and communicate about data gathering or generating processes, biases, and data shifts between domains in a precise manner. This can also assist in the careful design of experimental procedures to increase the model's generalization capability.

3.2 Data Shift in polyp segmentation

Polyp images are extracted frames from colonoscopy videos performed on a cohort of patients. Here, Kvasir-SEG (Jha et al., 2020a) dataset is taken for the data shift case study while acquiring polyp images(X). The annotation of the polyps (Y) was performed manually with the help of experts in Norway. Obviously, the manual and visual process depends upon image content, resolution, and contrast. So, the manual annotation of segmentation masks would not affect the images hence it is the case of casual prediction from $X \rightarrow Y$ i.e from *disease* to *segmentation*. The key point to notice is that the models trained on this manually annotated dataset will learn to annotate polyp images rather than predicting the anatomical layout.

Moreover, the Kvasir-SEG dataset was collected using a high-resolution electromagnetic imaging system named Olympus Europe on a certain population sample. So, if the model trained on Kvasir-SEG was deployed in a clinical setting where a different type or version of endoscopy machine is used then it becomes the case of *dataset shift* as the quality of the images can differ (*acquisition shift* along with the size and nature of polyps (*population shift*).

In this particular case, the causal direction can be easily mapped but this can be challenging if the endoscopist or doctors involved in segmentation take into account other additional information that can be more significant

than the information from the images such as pre-conditions in a patient, blood test and other diagnostic information. Therefore, modeling the data generation process and gathering extra metadata can be significant to report biases and identify a casual relationship in a given dataset.

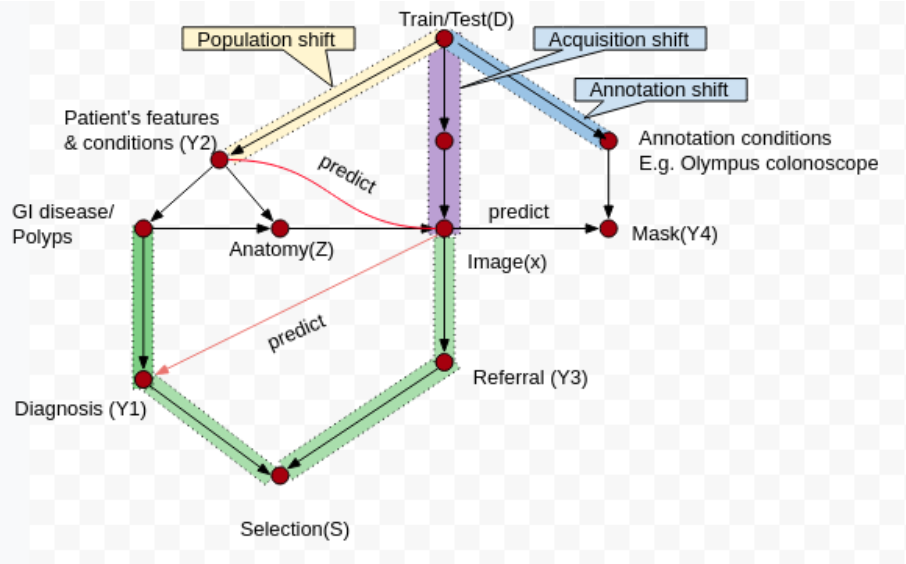

FIGURE 3.5: Causal diagram of Kvasir-SEG dataset showing possible medical work flows. In the dataset, casusal direction from disease to image to mask has been established. Population shift , acquistion shift and possible annotation shift in test domain are mapped in the diagram. (Daniel C. Castro, 2019)

3.3 Polyp Datasets

Polyps are abnormal protruding tissues from the mucous membrane which looks like small bumps or tiny mushroom-like stalks. They are commonly found in the colon region. The polyps are mostly benign but they can grow rapidly and become malignant (Ann Pietrangelo, 2018). Colorectal polyps can be classified into three categories depending upon the sizes of polyps namely diminutive ($\leq$ 5mm), small (6 to 9 mm), and large ($\geq$ 10 mm) (Lee, 2016). The main objective of colonoscopy which is an invasive and time taking procedure is to detect and remove polyps early. However, it has been reported that about 20% polyps are missed or overlooked by doctors or endoscopists' during endoscopy (Kaminski, 2010). Reducing the risk of polyp cancer by assisting doctors in the early detection of polyps is vital where machine learning models can play a significant role.

As discussed in the previous section about the challenges of collection and curation of datasets in a medical setting, the generation and curation of polyp datasets for feeding the machine learning models is a costly and time consuming process. It requires experts and careful consideration of data shift due to population samples, endoscope manufacturers, and endoscopic

procedures. Hence, there is a scarcity of properly annotated data which is required by data-hungry deep machine learning models.

The dataset taken for this thesis project comes from three different sources namely $Kvasir-SEG$ (Jha et al., 2020a), $ETIS-Larib$ (J. Silva and Granado, 2014), $CVC-Clinic$ (J. Bernal, 2015) which contain colonscopic images. The data from $CVC-Clinic$ was divided into two sets $CVC-612$ and $CVC-colonDB$. The description of these dataset has been tabulated below in table 3.1

Dataset	Organ	Source	Findings	Dataset Content	Access
Kvasir-SEG	Large Bowel	White light imaging	Polyp	1000 images with GT	Public
ETIS-Larib	Colonoscopy	White light imaging	Polyp	196 images with GT	Public
CVC-Clinic	Colonoscopy	White light imaging	Polyp	612 images with GT	Public
CVC-Colon DB	Colonoscopy	White light imaging	Polyp	380 images with GT	Public

TABLE 3.1: Different sources of polyp dataset and their characteristics. .

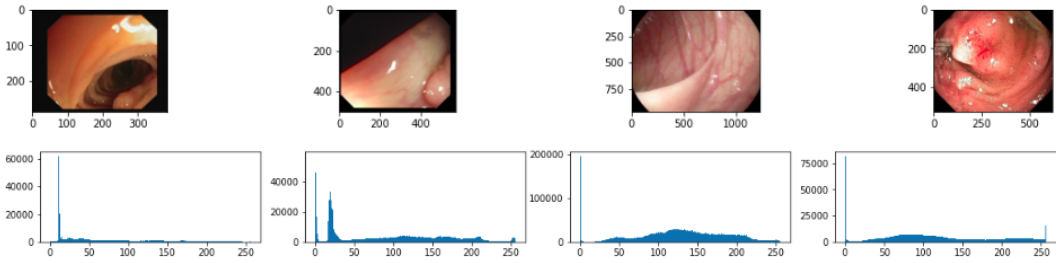

FIGURE 3.6: Noting intensity distribution of a sample image from four different data sources (From L to R : $CVC-612$, $CVC-ColonDB$, $ETIS-LaribPolypDB$, $Kvasir-SEG$)

All the images in datasets mentioned in table 3.1 were manually annotated. The image size in datasets varies from 332×487 to 1920×1072 pixels. The images were acquired through conventional white light imaging and

contain polyps. As data examples were generated using different colonoscopes, there exists variation in pixel intensity distribution as shown in figure 3.6. Hence, the four different sources of datasets can provide enough variations to set up different tasks required to perform meta-learning.

3.4 Summary

Data challenges in the medical domain should be addressed in the right way to have an algorithm that will perform well when deployed in a different setting. Scarcity of labeled and data mismatch during the training and deployment phase impacts the generalization capacity of the model. For implementing the polyp segmentation task, images were taken from four different sources. The bias in the dataset exists due to population shift, acquisition shift, and annotation shift. This can be taken as a good setting to generate various tasks for implementing the meta-learning method.

Chapter 4

Methodology

4.1 Task Generation

As mentioned in chapter 2, the meta-learning model is trained across a set of tasks. The model learns over the distribution of tasks $p(T)$. In an N-way, K-shot setting, the model adapts to learn a new task T_i which consists of K samples of each N classes.

In this work, four different polyp datasets are considered as mentioned in 3. To test the generalization capacity of the model, training tasks are sampled from three out of four datasets and then tested on the testing task sampled from the held-out dataset. A task consists of K shots from each dataset. So, the meta-learner for polyp segmentation gets a task D_i^{train} comprising of $3 \times K$ training points. While generating each task, the sampling of images and their corresponding masks were done randomly without replacement.

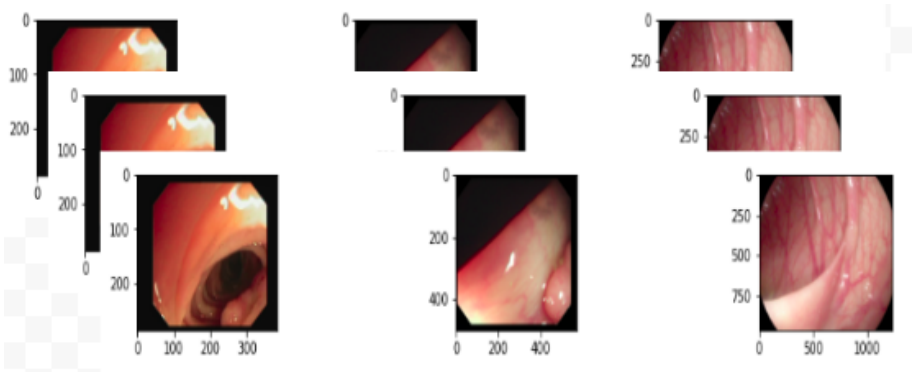

FIGURE 4.1: A sample task D_i^{train} with shots K=3 randomly drawn from three out of four polyp datasets.

4.2 Problem setup

As discussed in the previous section, tasks were generated. Each task was divided into *meta-training training* data and *meta-testing test data*.Then the inner-loop training was set up which takes the meta-parameter θ and runs the inner-loop steps yielding out the final task parameter. The final task parameter is then evaluated with the meta-loss and then tests on *meta-training test data*. The model f learns by observing the change in meta-training test

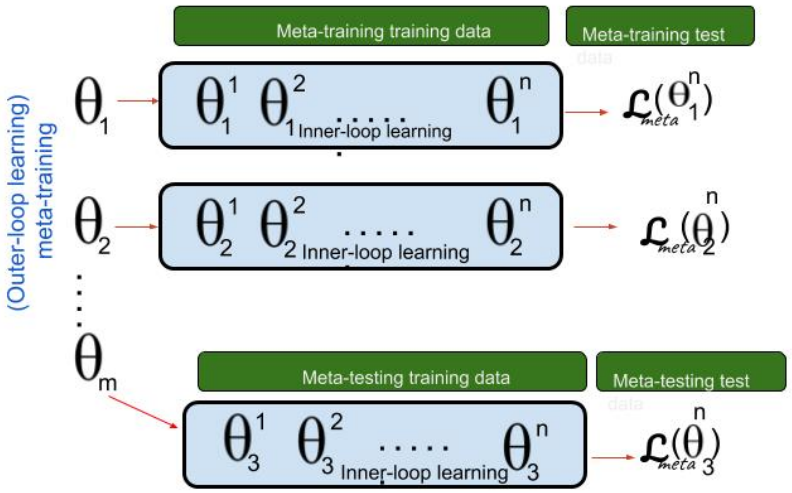

FIGURE 4.2: Showing the schema of meta-learning process highlighting the terms used to refer dataset at different level of the process.

error with respect to the updated task-parameters. So, in nutshell, the test error on *meta-training test data* acts as the training error in meta-learning. After meta-training, the final meta-parameter or the learned algorithm is evaluated for how good is it to learn a new task which is done with *meta-testing training data* and tested with *meta-testing test data*.

For implementing the optimization-based meta-learning, four problem settings were explored while testing on task from held out dataset. First of all, the number of training shots was varied from 5-shot to 10-shot from each dataset. Then the number of tasks was increased randomly from the range of 10 to 20 tasks. The batch size which is the number of tasks in this setting was set in the range from 1 to 2; keeping in mind the hardware limitations as an increase in the batch size overflows the GPU's memory.

4.3 Image Pre-Processing

Images and the corresponding ground truth were normalized in the range of [-1,1] and resized to 256×256. Further, no other pre-processing steps were applied to the images.

4.4 Network Architecture

The U-Net architecture was adapted as the base structure for meta-learning the polyp segmentation. UNet is known for its efficient performance on image segmentation task (Ronneberger, 2015). U-Net has a nice *U* shaped structure with *contraction* and *expansion* path.

The *contraction path* is also known as downsampling which looks like the conventional CNN architecture. It is comprised of 3×3 convolution followed by 2 max pooling. The number of channels is doubled at each downsampling step.

In the *expansion path* which is also known as upsampling, the number of channels is halved at each step. The upsampling is done by 2×2 convolution layer.

The *final layer* is 1×1 convolution which takes the features and maps them to the required number of classes.

The multi-stage cascaded convolutions neural networks help to extract the region of interest and make a dense prediction. Despite UNet having a good representational power, it also redundantly uses compute resources as it repeatedly extracts low-level features. Therefore, to overcome this drawback of UNet, the attention mechanism can be integrated with the UNet architecture. This has led to the improvement of the model's sensitivity to the region of interest and also suppresses features response from irrelevant regions in the image. The soft additive attention has shown better performance than the multiplicative attention (Minh-Thang, Hieu, and Christopher D., 2015).

The additive soft attention mechanism was integrated with the UNet architecture. The schema of the *attention UNet* architecture can be seen Figure 4.3. The key benefit of this attention UNet structure in comparison to multi-stage CNNs is that it does not require the training of multiple models to deal with object localization and thus reduces the number of model parameters. As seen in Figure 4.4, additive attention is applied to obtain the gating coefficient α.

Attention gates are integrated before concatenating operation which allows only relevant activations. The gradients from the background region are given less weight while performing a backward pass which allows the model to focus on relevant regions. The soft attention mechanism (Bahdanau et al., 2014) was first implemented in the field of NLP for the sentence to sentence translation.

Attention gate has been implemented as shown in figure 4.4 where two input feature maps are first passed through 1×1 convolution block, summed up, and then squeezed through $RELU$. Secondly, 1×1 convolution is performed again and passed through sigmoid activation function which gives output in the range of 0 and 1. This output is resampled by using bilinear interpolation and then multiplied with one of the input features to the attention block. Finally, the upsampled feature maps at the lower level are concatenated with the attention gate.

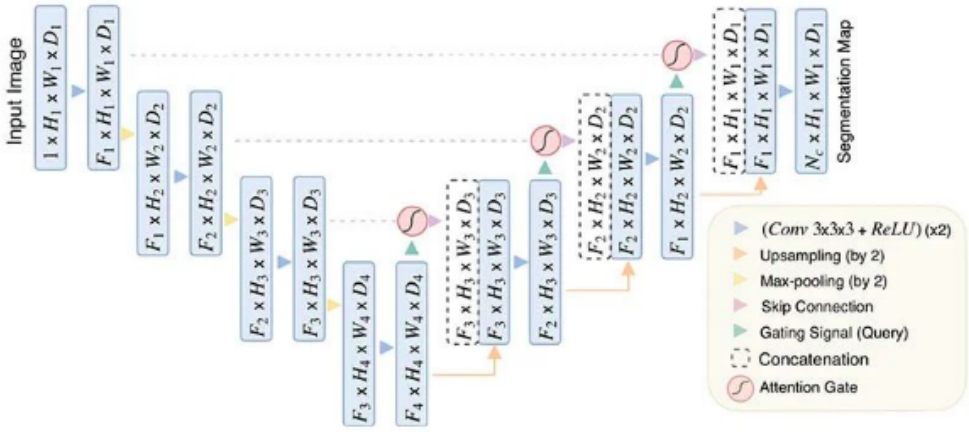

FIGURE 4.3: Schema of Attention with U Net segmentation model. Input image is down sampled by the factor of 2 in the encoding section of the model. Attention gates are connected with skip connection which filters the features. (Oktay et al., 2018)

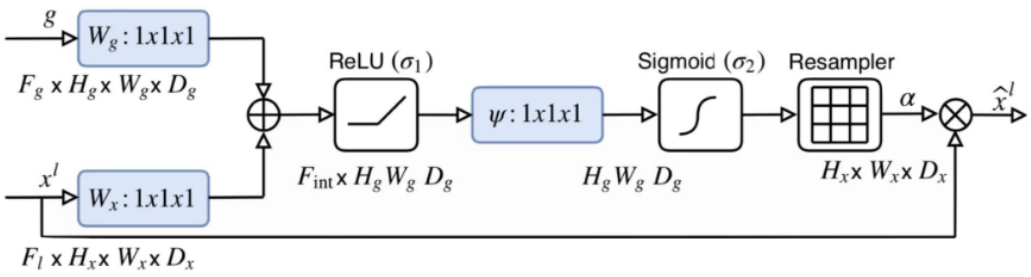

FIGURE 4.4: Schema of Attention block. Two input features are x_l and g are passed through conv. block and then scaled by attention factor α. Resampler is used to make the shape of α equal to feature map x_l. The output of the block is concatenated with the upsampled feature maps at lower level of the model. (Oktay et al., 2018)

4.5 Loss

A compound loss was used during training which comprises of both *log cosh dice loss* and *binary cross entropy loss* ($Loss = L_{BCE} + L_{lc-dce} + \lambda||\theta||_2^2$). L_2 regularization was applied while optimizing the loss.

Cross- entropy measures the difference between two probability distributions. It is generally used as an objective function of image classification and adapts well for image segmentation which is the classification of pixels.

Dice-coefficient is used as the metric for measuring segmentation results. This is also used as an objective function by applying negation. Despite working well in most of the cases, it sometimes fails to reach the optimal value due to its non-convex nature. So, to tackle this problem Lovxz extension was applied for smoothing dice loss (Maxim Berman and Blaschko., 2017). This stands out as $Log - Cosh(DiceLoss)$ where *cosh* makes the objective function tractable and easy to differentiate; and *Log* keeps the result of *cosh* in range. The experiment results with $Log - Cosh\ (DiceLoss)$ have shown better performance in comparison to other losses for learning segmentation task (Jadon, 2020).

$$L_{Dice} = 1 - \frac{2 * \sum y_{true} * y_{pred}}{\sum y_{true} + \sum y_{pred} + \epsilon} \tag{4.1}$$

$$L_{lc-dce} = log(cosh(L_{Dice})) \tag{4.2}$$

$$L_{BCE}(y, \hat{y}) = -(y\ log\ P(\hat{y})\ +\ (1-y)log(1-\hat{y})) \tag{4.3}$$

where, y and $\hat{y}$ are the ground truth value and predicted value respectively.

During the empirical study in this thesis, by changing the loss function from simple dice loss to dice loss that is smoothed by hyperbolic function *cosh*, the model yielded better score values.

4.6 Evaluation Metric

Dice score and intersection over union (IOU) are generally used as the evaluation metric for image segmentation. Dice score gives two times more weight to the area of intersection between the target and the predicted mask in comparison to IOU. They both range from value 0 to 1. The equation 4.4 and equation 4.5 below show the relationship between dice score and intersection over union .

$$Dice = \frac{2 * TP}{2 * TP + FP + FN}, \quad IOU = \frac{TP}{TP + FP + FN} \tag{4.4}$$

$$Dice = \frac{2 * IOU}{IOU + 1} \tag{4.5}$$

where; TP, FP , FN refers to *True Positive, False Positive and False Negative respectively.*

4.7 Baseline

For this thesis, a model trained by transfer learning was used for the baseline as it is a form of meta-learning. A UNet model pre-trained on brain MRI scans (Mateuszbuda, 2015) was taken and the attention mechanism was integrated. It was then fine-tuned with datasets from three different sources merged and tested on an unseen dataset from the held out fourth data source. The compound loss comprising of *BCE* and *Dice Loss* was used as the loss function.

The baseline model was adopted to take on **Medico challenge 2020** (Jha et al., 2020b) hosted by **Mediaeval** (*Mediaeval* 2020). The paper submitted for the challenge describes the baseline method used to solve the polyp segmentation task (Khadka, 2020).

4.8 Programming Framework

Pytorch was used as the deep learning framework to implement the model in this thesis. *Python* 3.6 was used as the scripting language for loading, pre-processing, and post-processing steps while implementing the notion of meta-learning for image segmentation. The library *higher* (*Higher*) was used to implement the inner loop and perform back propagation during meta-training phase.

The model was trained on a single NVIDIA Tesla V100-SXM3 GPU.

4.9 Summary

The methodology is comprised of training a UNet architecture guided with an attention mechanism using . Dice loss with binary cross-entropy was chosen as the loss function. The dice loss was modified for smoothing the curve with *cosh*. The network was initialized using weights obtained by pre-training UNet on the brain's MRI dataset. For comparing the result obtained from the meta-learning approach, UNet architecture with attention was trained on polyp datasets from three out of four different sources and tested on the fourth unseen polyp dataset.

Chapter 5

Experimental Results and Discussion

5.1 Experiment setup

Our experimental evaluation aimed to seek answers to the following questions: (1) How well does the iMAML algorithm respond while performing image segmentation with *few shot* in the medical setting? (2) How will the performance be affected by the increase in the number of training examples in each task? Will the score match the score from a conventional training algorithm? (3) In what way the performance changes when the number of tasks is increased?

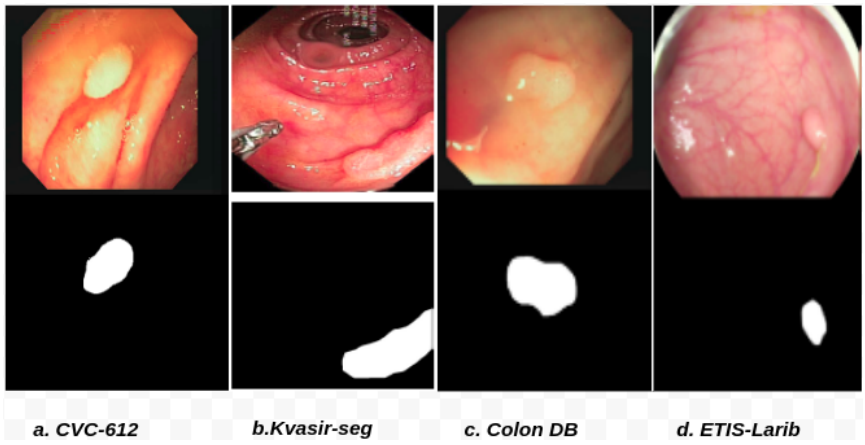

FIGURE 5.1: Sample polyp images with corresponding masks from different data sources used to generate tasks.

To study the above queries and empirical performance of the iMAML approach for segmentation, we took four datasets namely $Kvasir-SEG$, $ETIS-Larib$, $CVC-Clinic$ and $CVC-ColonDB$ from different sources as described in section 3.3. Then, 3-way k-shots tasks were generated for training the algorithm. In our case, the number of classes, i.e., the number of data pools was fixed to three, and "k" was varied. So, each gradient is computed using a batch size of $3 \times K$ examples. For testing, the task comprised in the form 1-way k-shot since only one dataset was held out.

In the first phase, the experimental tasks were setup with 3-way 5-shot and 3-way 10-shot respectively for each run. The number of tasks were fixed

at 10 task for each run of experiments. In the second phase of the experiment, number of tasks was increased to 20 by keeping the number of instances fixed to 10. Furthermore, ablation study was done to observe the impact of good initialization of the meta-learner on the performance of the model.

5.2 Results

Observing table 5.1, the best result was given by the meta-learning algorithm with mean dice score of 67.12% by using $k - shot = 10$. This was due to the increase in number of instances to 10 shot from each class which contributed in decreasing the generalization error of the model. The hyperparameter setting of the best result were recorded and similar experiments were carried out by changing the target set in each run to observe the generalization capacity of the meta-learner. The total number of tasks was increased to 20. The number of instances in a task during the meta testing phase was kept equal to the number of instances in a task during the meta training phase.

Algorithm	K-shots	No.of task	Target-set	mDice	mIOU
Baseline	1000	-	CVC-Colon DB	62.56	45.51
Meta-learned	5	10	CVC-Colon DB	65.46	48.65
Meta-learned	10	10	CVC-Colon DB	67.12	50.51

TABLE 5.1: Mean Dice and IOU results obtained by testing on CVC-Colon DB dataset. The training tasks were created from the other three datasets as mentioned in section 3.3. Baseline model mentioned in section 4.7 was also tested on $CVC - colonDB$ dataset. Both the meta-learning setup gave a better performance than the baseline model which was trained by the naive merging of three out of four datasets.

All three variants of the set-up produced better results than its corresponding baseline result for which the mean dice score stood at around 75% approximately. This indicates the efficacy of the meta-learning process in learning with few-shots from a diverse set of tasks over learning from scratch with the naive merging of different datasets. The baseline model was trained for 100 *epochs* with the learning rate of 0.0001.

The results from the meta-learning approach also surpassed *Att + UNet* (Khadka, 2020) which was trained over 800 training examples from $Kvasir - seg$ dataset and tested on completely unseen test set provided by the organizer of *Medico 2020*. However, the results from the meta-learning process could not match the state of art result by *pranet* (Deng-Ping et al., 2020) which was was trained over a single dataset source under a standard machine learning process. In comparing our result with another model named as ResUNet+ (Jha et al., 2019) which was also trained on CVC-612 dataset from scratch under standard supervised setting, it provided a ballpark idea

Algorithm	K-shots	Tr. tasks	Target-set	mDice	mIOU
Pranet	800	-	Kvasir-seg	89.8	84.0
ResUNet++	612	-	CVC-612	79.55	79.62
Att.+ UNet	800	-	Kvasir-seg	72.86	64.37
Baseline	1000	-	Kvasir-seg	64.66	47.77
Baseline	1000	-	ETIS-LaribPolypDB	64.53	47.63
Baseline	1000	-	CVC-612	63.76	47.98
Meta-learned	10	20	Kvasir-seg	74.63	59.52
Meta-learned	10	20	ETIS-LaribPolypDB	74.25	59.04
Meta-learned	10	20	CVC-612	75.54	60.06

TABLE 5.2: Testing for generalization capacity of the meta-learner by varying the target set for each run. In each case, the meta-learning approach outperformed the baseline model which was trained by the naive merging of the data set.

about margin of performance between our model and other successful studies in the field of polyp segmentation. The margin of performance score stood just under 4%.

5.3 Hyper-parameters

The hyper-parameters of the model architecture has been described in section 4.4. The hyper-parameters of the meta-learning process that yielded the best performance as observed in table 5.2 has been recorded below in table 5.3.

Target Set	Inner LR	Outer LR	Epochs	Inner Loop	CG	λ	Batch Size	Inner Optimizer	Outer Optimizer
CVC-612	$1e^{-5}$	$1e^{-5}$	50	100	2	100	2	SGD	ADAM

TABLE 5.3: The set of hyper-parameters that resulted in the best meta-learning performance while taking the test tasks from $CVC-612$ dataset.

The number of unseen test tasks was set to two for the above experiments to get the average test performance over two different tasks. The batch size corresponds to the number of tasks per each iteration during training and testing. The batch size of the task was kept 2 to memory constraint of the GPU. $'\lambda'$ acts as the regularization strength and is a scalar hyper-parameter. A weight decay of 0.0005 was also applied in the optimization step.

It was observed that by increasing the number of inner-loop during meta-training, the meta-learner learns a more refined weight. Similarly, for our implementation, the learning rate of inner and outer loop in the range of $1e^{-4}$ *to* $1e^{-5}$ helped producing good results. It was also noticed that the model's performance converged in between 30 to 50 epochs. The number of conjugate gradient steps was kept at 2 as it was a trade-off between having further accurate gradients and speed.

5.4 Ablation Analysis

In this ablation study, the importance of having good initialization of weights belonging to meta-learner was studied. At first, the meta-learner was trained with random initialization and secondly, the meta-learner was initialized with pre-trained weights obtained from training on brain MRI scans. The hyperparameters for the study were taken from the best run during the experiments as tabulated in 5.3. The results obtained have been tabulated below.

Algorithm	K-shots	Tr. tasks	Target-set	mDice	mIOU
UNet	10	10	Kvasir-seg	45.81	29.7
UNet+Pretrained weights	10	10	Kvasir-seg	68.34	51.90

TABLE 5.4: The results show improvement in performance of the meta-learner when its parameters were initialized by pre-trained weights.

During meta-test time, the prior weights of the meta-learner impacts the weight of the task specific parameter. So, imposing a good prior will help the meta-learner to generalize well on unseen meta-test tasks. The ablation study suggested that initializing the meta-learner with a good relevant pre-trained weight will help to generalize well over unseen tasks.

5.5 Discussion

Empirically, it has been demonstrated that the learning procedure acquired by implementing iMAML can generalize well over unseen segmentation task even under a few-shot setting. In comparing the meta-learner's performance with models trained with standard approaches for polyp segmentation as *pranet* and $ResUNet++$, it was found that the performance score of our model was within 14% and 4% margin respectively. It indicates that the meta-learning approach adopted for polyp segmentation task can be a plausible method. The meta-learner's performance was improved further with the tuning of some hyperparameters and increasing the number of training tasks to 20. This suggests that increasing the number of tasks would improve the performance of the model. The performance also improved when

more variation into tasks was induced by applying some image augmentation techniques. This indicates that with the increase in contrasting classes or datasets in the task, the performance of the meta-learning algorithm on polyp segmentation can be further improved. It was also found that iMAML was highly resistive to over-fitting. It was also demonstrated that a decent dice score could be obtained through meta-learning under a few-shot setting with room for improvement by tuning hyperparameters and with good initialization of meta-learner's weights such as using weights from pretrained network.

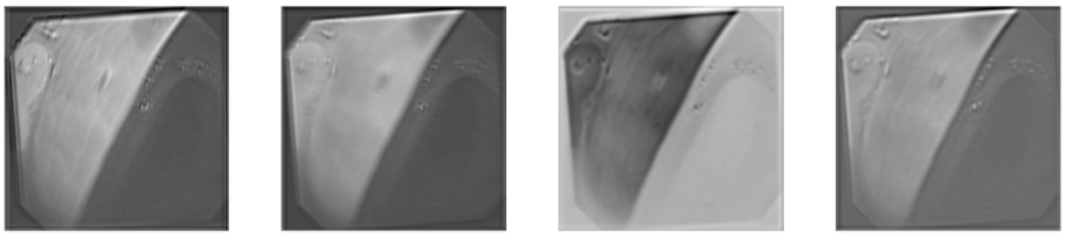

FIGURE 5.2: Activation map from the 17th CONV layer of the baseline model.

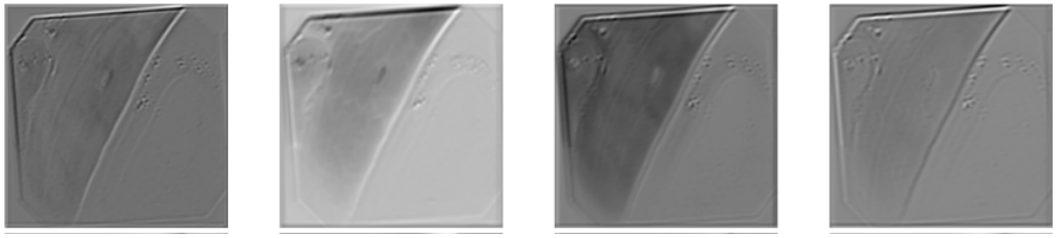

FIGURE 5.3: Activation map from the 17th CONV layer of the meta-learned model

The activation of the network for both the baseline model and meta-learned model were compared. The activations were taken from the 17th convolutional layer (CONV layer). It can be observed that the activation map from the baseline model is more active and non-sparse in comparison to the activation map from the meta-learned model. It suggests that the meta-learning process intelligently learns to activate neurons which contributes to the generalization capacity of the model.

The loss plot in figure 5.4 corresponds to the result in table 5.1. The performance is further improved when K, i.e., the number of instances per task was increased from 5 to 10. There is a clear trend that the performance can be further improved by including more instances in the task.

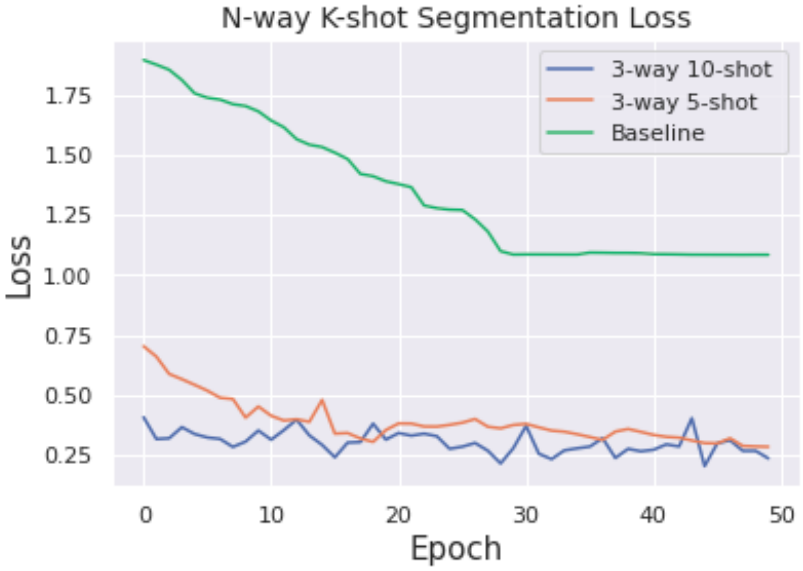

FIGURE 5.4: Comparison between the baseline model trained in a standard way and meta-learning with algorithm trained under different task setup. handled tasks more successfully when it was given unseen tasks from a different source. Loss values were taken from meta-testing phase.

5.6 Summary

The findings from the empirical study of meta-learning polyp segmentation tasks demonstrate that using optimization base meta-learning algorithm like , a difficult task like polyp segmentation can be learned successfully under a few-shot supervised setting. It also showed that the generalization capacity of the meta-learner is better than the results obtained from training models on merged datasets directly. The results also indicated that the performance of the meta-learner for polyp segmentation can be improved by doing some hyperparameter search, increasing the number of tasks and the number of k-shot in each task. The application results point out to future work that involves learning flexibly from various diverse tasks to perform well on some unseen task .

Chapter 6

Conclusion

6.1 Summary

In this thesis, we took learning to learn approach known as meta-learning to solve *polyp segmentation* problem using tasks containing few shot examples from different polyp datasets. This thesis investigated how effectively a difficult problem like segmentation in the medical domain can be tackled using a meta-learning approach. The various approaches of meta-learning were reviewed and the optimization-based meta-learning algorithm namely iMAML was chosen. During the literature review, it was found that the optimization-based-meta learning in comparison to the black-box and non-parametric meta-learning approach, was more stable, consistent, and expressive. This family of meta-learning algorithms are model agnostic, have a greater positive inductive bias, and can handle varying numbers of shots (K) in the tasks. However, the MAML algorithm is compute and memory heavy as it back propagates through the inner loop optimizer steps. A better version of MAML is iMAML in which the gradient depends upon the solution of the inner loop and not the steps taken. This makes iMAML faster and more efficient which is the reason for adopting the algorithm in this thesis.

The UNet architecture with attention mechanism was chosen as the model for the segmentation task. It was trained under different hyperparameter settings across various experimental setups. The tasks for the meta-training process were generated based on three out of four datasets and the held out dataset was used to generate the meta-testing set.

6.2 Contributions and Remarks

During the implementation of the optimization based meta-learning, data pipeline for task generation from multi-data sources was customized. Then the iMAML algorithm was adopted to meta-learn under *N way K shot* setting for polyp segmentation. The key conclusion of this thesis is that the meta-learning process can be a plausible approach to solve segmentation tasks in a medical setting even with few training examples. The findings support the outlined hypothesis in section 1.2. The subset of this work on the concept of knowledge transfer applied in conjunction with attention mechanism was submitted to *MediaEval20* and the model adopted in the paper was taken as the baseline model. The best dice score stood at 0.7554 with task setting of

3-way 10-shot which outperformed the baseline model that was trained with 1000 data points. This suggests that the meta-learning approach can efficiently handle segmentation tasks and generalize well to unseen tasks with careful tuning of hyperparameters. However, tasks with a 3-way 5-shot gave a dice score of value 0.6546, but improved significantly as the number of shots per class and the number of training tasks were increased. The results indicates that the meta-learning process can be advantageous in a medical setting where data is scarce and biased with data shifts during deployment. The experimental analysis also suggested that the inclusion of diverse classes in the task improved the generalization capacity of the model. This opens up the possibility of creating a robust model by training it with different classes in a medical setting such as CT scans, polyp images, kidney scans, and others.

The research also posed some questions. One of the questions is related to *data generation for building a task*. Meta-learning tasks are generated by sampling data points from a pool of different data sources or classes. The question is whether synthetic images can be used with real images during the meta-learning process. The synthetic images could increase diversity within tasks which can help increase the generalization capacity of the model. The second question is can meta-learning algorithm handle a large number of classes or sources in a task. The increment in the number of classes within a task could lead to *catastrophic forgetting*. The third question came as we were not able to cover in the experiment the effect of a large number of instances in a task. By increasing, the number of instances in a task, will the meta-learning process *surpass* the result of a model that is trained on a large number of data points from a single source? The increase in number of instances could help the meta-learner to generalize well on unseen task in compare to the model trained under standard supervised setting .

6.3 Future Research

The result of the 5-shot task was poor and suggest that the model could not handle the domain shift given our dataset pool. A task with data from diverse domains could increase the model's ability to detect the contrast in domain shift as the medical images of polyps do not have enough contrasting features though they were gathered from different data sources. Thus, further research with data sourced from diverse domains is required.

Meta-learning opens up the possibility of integrating the process with federated learning which is the idea of training models across multiple edge devices . This would be the natural setting for meta-learner algorithms to learn from different hospital settings around the world. So, there lies the potential of improving the generalization capacity of meta-learning models by integrating with the federated learning process. Moreover, meta-learning can be married with federated learning to improve privacy in the field of healthcare. [section 2.1.6]

Having a system that can build the learning based on prior experiences, it can be envisioned to build a life long learner without catastrophic forgetting.

This setting would allow the model to take on a new task with ease. However, the question remains how much data or diversity in data is required to have a previous experience that can handle new unseen tasks smoothly and learn continuously.

6.4 Opinion

It is scary and at the same time exciting to imagine our world with the artificial learner that can transfer prior experience, learn continuously, and take on a completely unseen task. This leads us to the direction of having AI that can mimic human intelligence closely.

Appendix A

Appendix

A.1 Proof for implicit gradient

Lemma 1, restated. Consider $Alg_i(\theta)$ as defined in Eqn. 2.13 for task T_i. Let $\phi = Alg_i^*(\theta)$ be the result of $Alg_i^*(\theta)$. If $(I + \frac{1}{\lambda}\nabla_\phi^2 \mathcal{L}_i(\phi_i))$ is invertible, then the derivative Jacobian is (Aravind Rajeswaran, 2019)

$$\frac{d\mathcal{A}lg_i^\star(\boldsymbol{\theta})}{d\boldsymbol{\theta}} = \left(\boldsymbol{I} + \frac{1}{\lambda}\nabla_\phi^2 \hat{\mathcal{L}}_i(\phi_i)\right)^{-1}.$$

Proof. We drop the task i subscripts in the proof for convenience. Since $\phi = \mathcal{A}lg^\star(\boldsymbol{\theta})$ is the minimizer of $G(\phi', \boldsymbol{\theta})$ in Eq. 4, the stationary point conditions imply that

$$\nabla_{\phi'} G(\phi', \boldsymbol{\theta})\,|_{\phi'=\phi} = 0 \implies \nabla\hat{\mathcal{L}}(\phi) + \lambda(\phi - \boldsymbol{\theta}) = 0 \implies \phi = \boldsymbol{\theta} - \frac{1}{\lambda}\nabla\hat{\mathcal{L}}(\phi),$$

which is an implicit equation that often arises in proximal point methods. When the derivative exists, we can differentiate the above equation to obtain:

$$\frac{d\phi}{d\boldsymbol{\theta}} = \boldsymbol{I} - \frac{1}{\lambda}\nabla^2\hat{\mathcal{L}}(\phi)\frac{d\phi}{d\boldsymbol{\theta}} \implies \left(\boldsymbol{I} + \frac{1}{\lambda}\nabla^2\hat{\mathcal{L}}(\phi)\right)\frac{d\phi}{d\boldsymbol{\theta}} = \boldsymbol{I}.$$

which completes the proof. □

Recall that:

$$G_i(\phi', \boldsymbol{\theta}) := \hat{\mathcal{L}}_i(\phi') + \frac{\lambda}{2}\,||\phi' - \boldsymbol{\theta}||^2.$$

A.2 Compute and Memory Efficiency of iMAML

The memory used in iMAML is independent of the number of gradient descent steps in the inner loop. The memory consumed is also independent of the number of CG iterations. We can observed below in figure A.1, memory consumption of MAML grows linearly while for iMAML stays constant no matter the number of conjugate gradient (CG) steps. Compute time for iMAML increases with CG steps but it gives accurate gradient in compare to FOMAML.

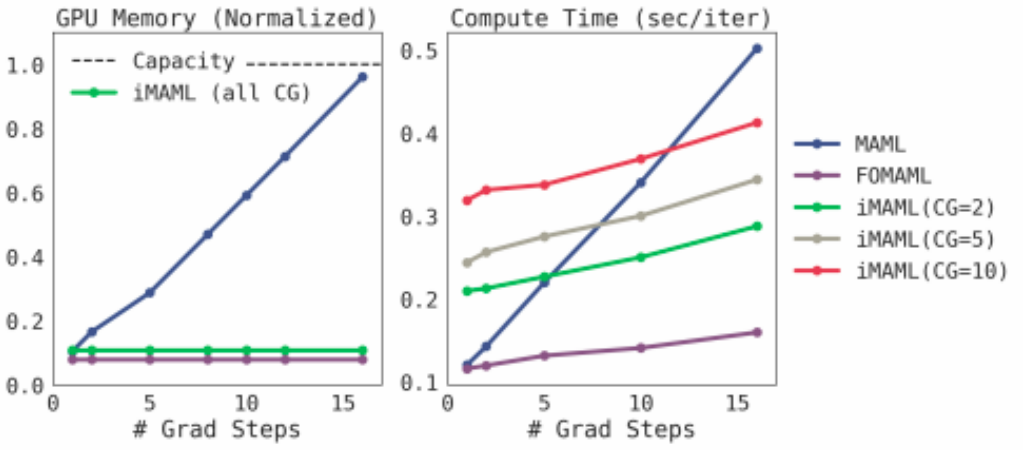

FIGURE A.1: Memory and Computation tradeoffs of iMAML, MAML and FOMAML. (Nikhil Mishra, 2018)

A.3 Package for Data Augmentation

Albumentations is a python library for image augmentation (*Albumentations : Data augmentation package*). This package was particularly chosen for its range of augmentation options. The augmentation used were rotation, flips, affine transformation, solar flare, color shift and normalization. Some of the effects of using augmentation on the images can be seen below:

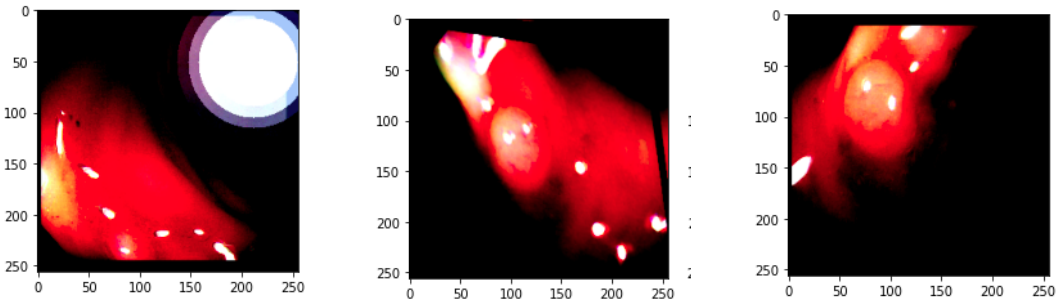

FIGURE A.2: Augmentation using Albumentation

A.4 Prediction

Samples of predicted outline by the meta-learning algorithm trained on N-way k-shot tasks where $N = 3$ and $K = 10$ can be viewed below in figure A.3

A.5 Source Code

The code for this thesis work has been uploaded to github at : https://github.com/IamRabin/Meta-Learning-Seg. All the packages' names with their version are included in *requirement.txt*.

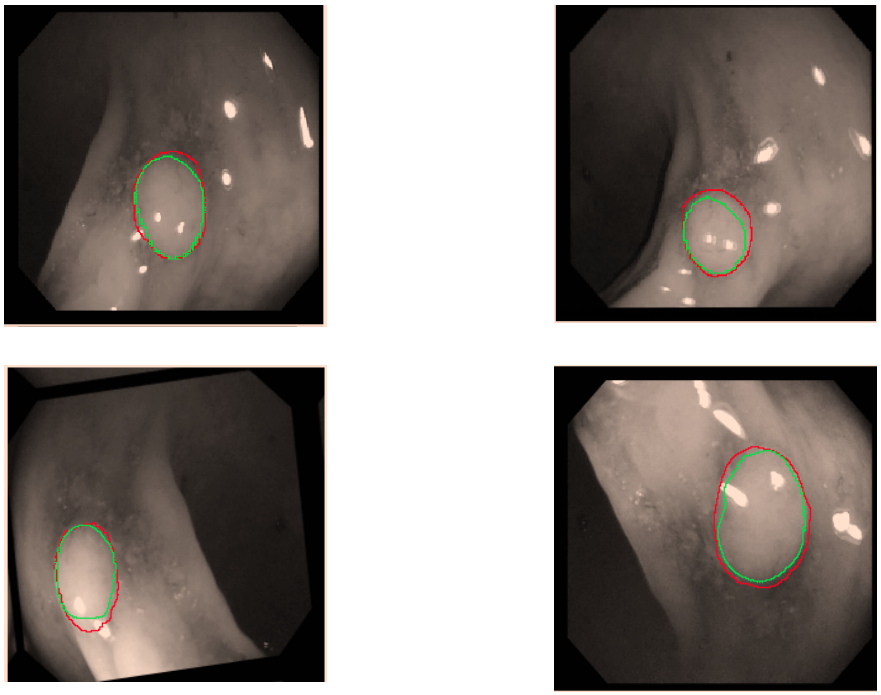

FIGURE A.3: Predicted outline is in red and groundtruth is in green.

Bibliography

Albumentations : Data augmentation package. URL: https://github.com/albumentations-team/albumentations.

Altman, ·N.S. (1992). "An Introduction to Kernel and Nearest Neighbor Nonparametric Regression". In: *The American Statistician* 46, pp. 175–185. URL: https://ecommons.cornell.edu/bitstream/handle/1813/31637/BU-1065-MA.pdf;jsessionid=FE47A7B5673FEB676971726684EB9AAB?sequence=1.

Amato F López A, Peña-Méndez EM-Vaňhara P Hampl A Havel J. (2013). "Artificial neural networks in medical diagnosis". In: *Appl Biomed 11(2)*, pp. 47–58. URL: http://jab.zsf.jcu.cz/artkey/jab-201302-0001_artificial-neural-networks-in-medical-diagnosis.php.

Ann Pietrangelo, Graham Rogers (2018). "What Are the Symptoms, Types, and Treatments for Polyps?" In: *Healthline*. URL: https://www.healthline.com/health/polyps.

Antreas Antoniou Harrison Edwards, Amos Storkey (2019). "How to train your MAML". In: *ICLR*. URL: https://arxiv.org/abs/1810.09502.

Aravind Rajeswaran Chelsea Finn, Sham Kakade1 Sergey Levine (2019). "Meta-Learning with Implicit Gradients". In: *NeurIPS*. URL: https://arxiv.org/abs/1909.04630.

B. McMahan E. Moore, D. Ramage S. Hampson and B. A. y Arcas (2017). "Communication-efficient learning of deep networks from decentralized data". In: *Artificial Intelligence and Statistics* 10,no.1, 1273–1282.

Bahdanau, D. et al. (2014). "Neural machine translation by jointly learning to align and translate." In: *arXiv preprint arXiv:1409.0473*.

Bird, J. J. et al. (2020). "Cross-Domain MLP and CNN Transfer Learning for Biological Signal Processing: EEG and EMG". In: *IEEE Access* 8, pp. 54789–54801. URL: https://ieeexplore.ieee.org/document/9027853/.

Chaitanya, K. et al (2019). "Semi-supervised and task-driven data augmentation". In: *Information Processing in Medical Imaging (IPMI 2019), vol. 11492*.

Chapelle O. Scholkopf, B. Zien A. (eds.) (2009). "Semi-Supervised Learning". In: *MIT Press, Cambridge, MA*.

Chelsea Finn, Pieter Abbeel and Sergey Levine (2017). "Model-agnostic meta-learning for fast adaptation of deep networks". In: *ICML*. URL: https://openreview.net/pdf?id=rJY0-Kcll.

Cireşan, Dan C. et al. (2013). "Mitosis Detection in Breast Cancer Histology Images with Deep Neural Networks". In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2013*. Ed. by Kensaku Mori et al. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 411–418. DOI: 10.1007/978-3-642-40763-5_51. URL: https://doi.org/10.1007\%2F978-3-642-40763-5_51.

D. Jha M. A. Riegler, D. Johansen P. Halvorsen and H. D. Johansen (2020). "Doubleu-net: A deep convolutional neural network for medical image segmentation". In: *Proceedings of the IEEE conference Computer Based Medical Systems (CBMS)*.

Daniel C. Castro Ian Walker, Ben Glocker (2019). "Causality matters in medical imaging". In: *Nature Communications 11 (2020) 3673*. URL: https://arxiv.org/pdf/1912.08142.pdf.

Deng-Ping, Fan et al. (2020). "PraNet: Parallel Reverse Attention Network for Polyp Segmentation". In: *IMICCAI*.

Dilsizian S.E., Siegel E.L. (2014). "Artificial Intelligence in Medicine and Cardiac Imaging: Harnessing Big Data and Advanced Computing to Provide Personalized Medical Diagnosis and Treatment". In: *Curr Cardiol,Rep 16, 441*, p. 441. URL: https://doi.org/10.1007/s11886-013-0441-8.

Donahue, Jeff et al. (2014). "DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition". In: *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*. ICML'14. Beijing, China: JMLR.org, I–647–I–655. DOI: 10.5555/3044805.3044879. URL: https://dl.acm.org/doi/10.5555/3044805.3044879.

DY. Shin H. A. Qadir, L. Aabakken J. Bergsland and I. Balasingham (2018). "Automatic colon polyp detection using region based deep cnn and post learning approaches". In: *IEEE* 6, pp. 40,950–40962.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville (2016). *Deep Learning*. http://www.deeplearningbook.org. MIT Press.

Higher. URL: https://github.com/facebookresearch/higher.

I. Goodfellow J. Pouget-Abadie, M. Mirza B. Xu D. Warde-Farley S. Ozair A. Courville and Y. Bengio (2014). "Generative adversarial nets". In: *in Advances in neural information processing systems*, pp. 2672–2680.

J. Bernal F. J. Sánchez, G. Fernández-Esparrach D. Gil C. Rodríguez and F. Vilariño (2015). "Wm-dova maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians". In: *Computerized Medical Imaging and Graphics* 43, 99–111.

J. Redmon S. Divvala, R. Girshick and A. Farhadi (2016). "You only look once: Unified, real-time object detection". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788.

J. Silva A. Histace, O. Romain X. Dray and B. Granado (2014). "Toward embedded detection of polyps in wce images for early diagnosis of colorectal cancer". In: *International Journal of Computer Assisted Radiology and Surgery* 9, 283–293.

Jadon, Shruti (2020). "A survey of loss functions for semantic segmentation". In: URL: https://arxiv.org/pdf/2006.14822.pdf.

Jaehong Kim Sangyeul Lee, Sungwan Kim Moonsu Cha Jung Kwon Lee Youngduck Choi Yongseok Choi Dong-Yeon Cho Jiwon Kim (2018). "Auto-Meta: Automated Gradient Based Meta Learner Search". In: *NIPS*. URL: https://arxiv.org/abs/1806.06927.

Jha, Debesh et al. (2019). "Resunet++: An advanced architecture for medical image segmentation". In: *Proc of. IEEE International Symposium on Multimedia (ISM)*. IEEE, pp. 225–2255.

Jha, Debesh et al. (2020a). "Kvasir-seg: A segmented polyp dataset". In: *Proc. of International Conference on Multimedia Modeling (MMM)*, pp. 451–462.

Jha, Debesh et al. (2020b). "Medico Multimedia Task at MediaEval 2020:Automatic Polyp Segmentation". In: *Proc. of MediaEval 2020 CEUR Workshop*.

Judea Pearl, Elias Bareinboim (2014). "External Validity: From Do-Calculus to Transportability Across Populations". In: *Statistical Science*. URL: https://arxiv.org/pdf/1503.01603.pdf.

Kaixin Wang Jun Hao Liew, Yingtian Zou Daquan Zhou Jiashi Feng (2019). "PANet: Few-Shot Image Semantic Segmentation with Prototype Alignment". In: *ICCV*. URL: https://arxiv.org/abs/1908.06391.

Kaminski (2010). "Quality indicators for colonoscopy and the risk of interval cancer". In: *New England Journal of Medicine* 362, pp. 1795–1803.

Khadka, Rabindra (2020). "Transfer of Knowledge: Fine-tuning for Polyp Segmentation with Attention". In: *Medico 2020*. URL: https://iamrabin.github.io/project_work/MediaEval_2020_paper_Rabindra_Khadka_UNITRK.pdf.

Koch, Gregory, Richard Zemel, and Ruslan Salakhutdinov (2015). "Siamese Neural Networks for One-shot Image Recognition". In: *ICML Deep Learning Workshop*.

L. Rocher J. M. Hendrickx, Y.-A. De Montjoye (2019). "Estimating the success of re-identifications in incomplete datasets using generative models". In: *Nature communications* 10,no.1, pp. 1–9.

LeCun, Yann; Léon Bottou; Yoshua Bengio; Patrick Haffner (1998). "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE. 86 (11)*.

Lee, J. (2016). "Resection of diminutive and small colorectal polyps: what is the optimal technique?" In: *Clinical endoscopy* 49, p. 355.

Lee, J. Y. (2020). "Real-time detection of colon polyps during colonoscopy using deep learning: systematic validation with four independent datasets". In: *Scientific Reports* 10,no.1, 1–9.

Lienhard Dina A., David H. Hubel and Torsten N. Wiesel (1964). "Research on Optical Development in Kittens". In: *Embryo Project Encyclopedia (2017-10-11),ISSN: 1940-5030*. URL: http://embryo.asu.edu/handle/10776/12995..

Mackworth, Poole (2017). *Artificial Intelligence: Foundations of Computational Agents*. https://artint.info/2e/html/ArtInt2e.Ch1.html. Cambridge University.

Mateuszbuda (2015). "U-NET FOR BRAIN MRI". In: URL: https://pytorch.org/hub/mateuszbuda_brain-segmentation-pytorch_unet/.

Maxim Berman, Amal Rannen Triki and Matthew B. Blaschko. (2017). "The lovsz-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks". In:

Mediaeval (2020). URL: https://multimediaeval.github.io/editions/2020/.

Minh-Thang, Luong, Pham Hieu, and Manning Christopher D. (2015). "Effective Approaches to Attention-based Neural Machine Translation". In: *arXiv:1508.04025*. URL: https://arxiv.org/abs/1508.04025.

Mnih V., Kavukcuoglu K. Silver D. et al. (2015). "Human-level control through deep reinforcement learning". In: *Nature 518*, 529–533. URL: https://doi.org/10.1038/nature14236.

Nichol, Alex, Joshua Achiam, and John Schulman (2018). "On First-Order Meta-Learning Algorithms". In: URL: https://arxiv.org/pdf/1803.02999.pdf.

Nikhil Mishra Mostafa Rohaninejad, Xi Chen Pieter Abbeel (2018). "A Simple Neural Attentive Meta-Learner". In: *ICLR*. URL: https://arxiv.org/abs/1810.09502.

Oktay, Ozan et al. (2018). "Attention U-Net: Learning Where to Look for the Pancreas". In: *arXiv:1804.03999*.

Pennachin, ·Cassio and Ben Goertzel (1992). *Contemporary Approaches to Artificial General Intelligence*. Springer, Berlin, Heidelberg. URL: https://doi.org/10.1007/978-3-540-68677-4_1.

Qi Dou Daniel C. Castro, Konstantinos Kamnitsas Ben Glocker (2019). "Domain Generalization via Model-Agnostic Learning of Semantic Features". In: URL: https://arxiv.org/abs/1910.13580.

Qianru Sun Yaoyao Liu, Tat-Seng Chua Bernt Schiele (2019). "Meta-Transfer Learning for Few-Shot Learning". In: *CVPR*. URL: https://openaccess.thecvf.com/content_CVPR_2019/papers/Sun_Meta-Transfer_Learning_for_Few-Shot_Learning_CVPR_2019_paper.pdf.

R. Wang S. Chen, C. Ji J. Fan and Y. Li (2020). "Boundary-aware context neural network for medical image segmentation". In: *arXiv preprint arXiv:2005.00966*.

Raina, Rajat, Andrew Y. Ng, and Daphne Koller (2006). "Constructing Informative Priors Using Transfer Learning". In: *Proceedings of the 23rd International Conference on Machine Learning*. ICML '06. Pittsburgh, Pennsylvania, USA: Association for Computing Machinery, 713–720. DOI: 10.1145/1143844.1143934. URL: https://doi.org/10.1145/1143844.1143934.

Ronneberger O., Fischer P. Brox (2015). "U-net:Convolutional networks for biomedical image segmentation". In: *MICCAI*.

Russell Stuart J.; Norvig, Peter (2003). *Artificial Intelligence: A Modern Approach*. Upper Saddle River, New Jersey: Prentice Hall,ISBN 0-13-790395-2.

Sachin Ravi, Hugo Larochelle (2017). "Optimization as a Model for Few-Shot Learning". In: *ICLR*. URL: https://openreview.net/pdf?id=rJY0-Kcll.

Sean M. Hendryx Andrew B. Leach, Paul D. Hein-Clayton T. Morrison Download PDF (2019). "Meta-Learning Initializations for Image Segmentation". In: URL: https://arxiv.org/pdf/1912.06290.pdf.

Selmer Bringsjord, Naveen Sundar Govindarajulu (2018). "Artificial Intelligence". In: URL: https://plato.stanford.edu/entries/artificial-intelligence/#StroVersWeakAI.

Silver, David et al. (2017). "Mastering the game of Go without human knowledge". In: *Nature* 550.7676, pp. 354–359. DOI: 10.1038/nature24270. URL: https://www.nature.com/articles/nature24270.

Tao Hu Chiliang Zhang, Gang Yu (2019). "Attention-Based Multi-Context Guiding for Few-Shot Semantic Segmentation". In: URL: https://doi.org/10.1609/aaai.v33i01.3301844.

Thrun Sebastian, Pratt Lorien (2012). *Learning to Learn*. Springer Science Business Media, pp. 3–5. ISBN: 9781461555292.

Tianhan Wei Xiang Li, Yau Pun Chen Yu-Wing Tai and Chi-Keung Tang (2018). "Fss-1000: A 1000-class dataset for few-shot segmentation." In: URL: arXivpreprintarXiv:1907.12347,2019.

Vapnik Vladimir N., Cortes Corinna (1995). "Support-vector networks". In: *Machine Learning, 20, 273-297*. URL: http://image.diku.dk/imagecanon/material/cortes_vapnik95.pdf.

Vilalta, Ricardo and Youssef Drissi (2002). "A Perspective View and Survey of Meta-Learning". In: *Artificial Intelligence Review* 18.2, pp. 77–95. DOI: 10.1023/A:1019956318069. URL: https://doi.org/10.1023/A:1019956318069.

Vinyals, Oriol et al. (2016). "Matching Networks for One Shot Learning". In: *Advances in Neural Information Processing Systems 29*. Ed. by D. D. Lee et al. Curran Associates, Inc., pp. 3630–3638. URL: http://papers.nips.cc/paper/6385-matching-networks-for-one-shot-learning.pdf.

Yosinski, Jason et al. (2014). "How Transferable Are Features in Deep Neural Networks?" In: NIPS'14. Montreal, Canada: MIT Press, 3320–3328.

Zhenguo Li Fengwei Zhou, Fei Chen Hang Li (2017). "Meta-SGD: Learning to Learn Quickly for Few-Shot Learning". In: URL: https://arxiv.org/abs/1707.09835.