Complete Interpretable Lesion Detection Framework for Diabetic Retinopathy in the Era of the Internet of Things (IoT)

by

Farzan Shenavarmasouleh

(Under the Direction of Hamid R. Arabnia)

Abstract

The continuous advancements of Machine Learning and Deep Learning have given birth to new but extremely powerful fields such as the Internet of Things (IoT) and Embodied AI. They can benefit from the powerful new Computer Vision, Natural Language Processing, and Reinforcement Learning algorithms to extract complex features from the underlying data. These features then will either get processed locally or will be sent securely to more powerful centralized processing departments to take advantage of their computing power and take proper actions. This foundation can be utilized in almost all fields of study and industries and one of the important ones is the healthcare industry. Despite the never-ending benefits of AI, it comes with some challenges as well, especially in healthcare. The collection of data and labeling them is often very expensive and time-consuming, different clinics use different devices with varying configurations, and also the decisions about the malignancy of the disease are often very subjective and depend on the doctor. Hence, different approaches such as Transfer Learning and Data Augmentation need to be employed to deal with the insufficient amount of data in healthcare. Also, most of the Machine Learning and Deep Learning models are often referred to as black boxes since tracking every decision of them and understanding the reasoning behind that decision is extremely difficult, and all in all these issues can lead to biases and misleading results that could be ultimately dangerous in the healthcare. In this dissertation, we first address each of the aforementioned issues and propose appropriate solutions for them. Then we use all these lessons to design, implement, and evaluate a multi-purpose

interpretable framework for preprocessing, analyzing, masking, and diagnosing Diabetic Retinopathy in diabetic patients that can be easily expanded with new features and has the ability to be incorporated into medical facilities as a real-time or post-processing assistant.

INDEX WORDS: Diabetic Retinopathy, Internet of Things, Embodied AI, Deep Learning, Machine Learning, Fundus, Lesion, Exudate, Microaneurysm, Retina, Mask R-CNN, LSTM

Complete Interpretable Lesion Detection Framework for Diabetic Retinopathy in the Era of the Internet of Things (IoT)

by

Farzan Shenavarmasouleh

B.S., Iran University of Science and Technology (IUST), Iran, 2018

A Dissertation Submitted to the Graduate Faculty of the
University of Georgia in Partial Fulfillment of the Requirements for the Degree

Doctor of Philosophy

Athens, Georgia

2022

Complete Interpretable Lesion Detection Framework for Diabetic Retinopathy in the Era of the Internet of Things (IoT)

by

Farzan Shenavarmasouleh

Major Professor: Hamid R. Arabnia

Committee: Thiab R. Taha

Khaled Rasheed

Electronic Version Approved:

Ron Walcott

Vice Provost for Graduate Education and Dean of the Graduate School

The University of Georgia

August 2022

Dedication

To the universe, and its creator(s)/programmer(s).

To all the events, their odds, and the apparent randomness.

To the most complicated neural network currently discovered, the human brain.

To my dearest professors, family members, and friends.

To me.

Acknowledgments

First and foremost, I would like to express my sincere gratitude to my advisor, Professor Hamid R. Arabnia, for his invaluable guidance, relentless support, and encouragement throughout my PhD journey without whom I would have never been able to navigate the complexities of the research process and produce a quality finished product. I would also like to thank my supervisory committee members, Professor Thiab Taha and Professor Khaled Rasheed, for their insightful comments and suggestions. I feel extremely fortunate for having such an outstanding and supportive committee. I am also grateful to my friends and colleagues at UGA, in particular, Farid Gharehmohammadi, for being an awesome friend, colleague, and lab mate and also providing professional help with the manuscript. Last but not least, I would like to thank my parents for their unwavering support and encouragement throughout my life.

Contents

Acknowledgments **v**

List of Figures **viii**

List of Tables **x**

1 Introduction **1**

1.1 Application of Deep Learning 2

1.2 Challenges 4

1.3 Contributions 5

1.4 Dissertation Outline 6

2 Causes of Misleading Statistics and Research Results Irreproducibility **8**

2.1 Resources and Tools 10

2.2 Data Collection 12

2.3 Data Analysis and Statistical Methods 13

2.4 Fallacious Deductions 16

2.5 Indecent Reporting 18

2.6 Open Science as a possible solution 18

2.7 Conclusion and Final Remarks 19

3 Embodied AI-Driven Operation of Smart Cities **20**

3.1 Rise of the Embodied AI 23

3.2 Breakdown of Embodied AI 25

3.3 Simulators . . . 28
3.4 Future of Embodied AI . . . 31
3.5 Conclusion and Final Remarks . . . 32

4 Machine Learning and Internet of Things (IOT) in Healthcare 33
4.1 Introduction . . . 34
4.2 IoT Pipeline . . . 37
4.3 Some of the Applications of IoT in Healthcare . . . 41
4.4 Machine learning Challenges in IoT Healthcare . . . 43
4.5 Alternative Promises in IoT Healthcare . . . 47
4.6 Conclusion and Final Remarks . . . 55

5 Applications of Deep Learning in Health Informatics 56
5.1 Classification and Segmentation Models . . . 58
5.2 More Applications . . . 64
5.3 Challenges . . . 66
5.4 Conclusion and Final Remarks . . . 68

6 Case Studies 70
6.1 Case Study I . . . 71
6.2 Case Study II . . . 80
6.3 Final Remarks . . . 96

7 DRDr: Automatic Masking of Exudates and Microaneurysms Caused By Diabetic Retinopathy Using Mask R-CNN and Transfer Learning 98
7.1 Related Work . . . 101
7.2 Methodology . . . 104
7.3 Experiments and Results . . . 108
7.4 Conclusion and Final Remarks . . . 109

8 DRDr II: Detecting the Severity Level of Diabetic Retinopathy Using Mask RCNN and Transfer Learning **111**

8.1 Related Work . . . 113

8.2 Methodology . . . 115

8.3 Experiments and Results . . . 118

8.4 Conclusion and Final Remarks . . . 119

9 DRDrV3: Complete Lesion Detection in Fundus Images Using Mask R-CNN, Transfer Learning, and LSTM **120**

9.1 Related Works . . . 123

9.2 Proposed Method and Problem Description . . . 124

9.3 Evaluation and Results . . . 128

9.4 Conclusion and Final Remarks . . . 131

10 Conclusion **133**

Bibliography **135**

List of Figures

3.1 Embodied AI in Smart Cities 22

4.1 Overview of Machine Learning Techniques 37
4.2 Overview of IoT pipeline 38
4.3 A general schema of IoT and Machine Learning applications in Personal Healthcare (PH) 43
4.4 A general ML challenges in IoT healthcare and associated solutions 44
4.5 A general schema of Online Machine Learning 49
4.6 A general schema of Federated Learning 52

6.1 Proposed framework for big data analysis based on SPARK 72
6.2 COVID-19 Spread plot for Georgia,Alabama and Florida 76
6.3 COVID-19 Spread plot for Illinois, Michigan and New York 77
6.4 COVID-19 Spread Demographic plot for California, Oregon and Washington 78
6.5 COVID-19 Spread Demographic chart for 9 states 78
6.6 COVID-19 Death Demographic chart for 9 states 78
6.7 DeepMSRF architecture. Step 1: A unimodal VGGNET takes the speaker's image as an input and detects the speaker's gender. Step 2: Based on the gender, the image and voice of the speaker is passed to the corresponding parallel multimodal VGGNETs to extract each modality's dense features. Step 3: Feature selection on each modality will be applied first; then, the resultant feature vectors are concatenated, and feature selection is performed again after concatenation. Eventually, a classifier is trained to recognize the speaker's identity. 82
6.8 Training time Comparison (Male Vs. Female) for DeepMSRF with SVM classifier . . . 95

7.1 (a) an image from e-ophtha EX containing exudates. (b) binary mask showing the position of exudates in image (a). (c) an image from e-ophtha MA containing microaneurysms. (d) binary mask showing the location of microaneurysms in picture (c). . . . 105
7.2 (a) an example of the original image from e-ophtha MA and its mask. (b) resulting image and its mask after the preprocessing phase. 106
7.3 (a) original image (b) predicted masks, bounding boxes, their score and IoU (c) Types of the lesions detected and their scores. (d) sample activations of a few layers of the model . 110

8.1 (a) original images in the dataset. (b) perform all the Preprocessing I steps explained in section 3.2. (c) preprocessed images. (d,e) pump all the preprocessed images into the DRDr model and use the output generated by it for all images to create a data frame for features. The steps are explained in section 3.3. (f) apply the Preprocessing II steps explained in section 3.4. (g,h) feed the data frame to an arbitrary classifier and get the severity levels as explained in section 4. 113

9.1 The lesions caused by diabetic retinopathy, namely exudates (red rectangles) and microaneurysms (black rectangles) in the eye 122
9.2 Proposed method in two phases 123
9.3 The architecture of phase one 126
9.4 The architecture of phase two 127
9.5 The input layers of LSTM 129

List of Tables

6.1 Comparing correlation results for Georgia, Albany and Florida State during (Feb. 1st-Aug. 31) . . . 74

6.2 Comparing correlation results for New York, Michigan and Illinois during (Feb. 1st-Aug. 31) . . . 74

6.3 Comparing correlation results for California, Washington and Oregon during (Feb. 1st-Aug. 31) . . . 75

6.4 The accuracy for single/multi modality with/without feature selection . . . 93

6.5 The accuracy of the speakers' face images for four different classifiers associated with different feature extractors with/without Feature Selection (FS) . . . 94

6.6 The accuracy of the whole dataset for SVM classifier associated with Spectrogram feature extractor combined with feature selection . . . 94

6.7 The Accuracy for single/multi modality with/out feature selection . . . 94

6.8 The accuracy of the whole dataset for SVM classifier associated with Spectrogram feature extractor combined with feature selection . . . 96

7.1 mAP for train, validation, and test sets created from e-ophtha EX and e-ophtha MA datasets . . . 109

8.1 Hyper-parameters for Mask RCNN used for DRDr . . . 116

8.2 DRDr II results for different classifiers . . . 119

9.1 result of phase one with three different thresholds. The numbers (35,50 and 75) are the threshold of Intersection over Union (IOU) . . . 130

9.2 result of phase two accuracy and the corresponding confusion matrices 130
9.3 Accuracy of DRDrV3 after incorporating new masking layer versus benchmarks 131

Chapter 1

Introduction

Machine learning is a subfield of artificial intelligence concerned with the design and development of algorithms that can learn from and predict data. It is a strong tool that may be used to enhance systems automatically.

Deep learning models, often known as deep neural learning or deep neural networks (DNNs), are multilayer neural networks that can automatically identify patterns and insights from massive datasets, learning from data in ways that most people cannot. Deep learning is a useful technique for predictive modeling since it allows you to recognize complicated patterns and make predictions about what will happen in the future.

Deep learning models are more accurate than typical machine learning models, and they're easier to build, deploy, and maintain. It's vital to keep in mind, though, that machine learning and deep learning models are only as good as the data they're given. The outputs of the algorithms will be of poor quality if the data is of poor quality.

Image processing is a branch of computer science that deals with digital image manipulation. Digital pictures are ubiquitous, from photography and artwork to medical imaging and satellite data, therefore it's an essential topic of study. There are several strategies for processing an image, and the techniques employed vary based on the type of image and the desired outcome. Image processing can be used to improve image quality, edit or eliminate undesired parts, or add unique effects to an image. It may also be used to automate repetitive activities like eliminating red-eye from images or fixing an image's color balance.

Computer vision is the technique of interpreting and comprehending digital images using computers and image processing algorithms. It is a branch of artificial intelligence concerned with the development of algorithms capable of autonomously processing and comprehending digital pictures.

Computer vision is a fast expanding science with several applications in areas such as car safety, medical image analysis, biometrics, and robotics. It's also been employed in military applications including target detection and navigation. The most often used technique in Computer Vision is the top-down approach which begins with an image's general structure and then breaks it down into smaller bits. Many current computer vision systems operate in this manner, as it is frequently more efficient to begin with the bigger structures. Image segmentation and pattern recognition are two of the most essential computer vision algorithms.

The technique of segmenting a picture into smaller parts that can be analyzed individually is known as image segmentation. This is frequently done in order to limit the quantity of data that must be processed or to isolate certain objects or characteristics in an image. Pattern recognition is the technique of recognizing certain patterns using a set of characteristics. This may be used for classification jobs like recognizing faces in photographs.

1.1 Application of Deep Learning

Machine learning and deep learning have unlimited applications. Retailers may employ machine learning algorithms to forecast what products a customer will buy and when they will buy them. This data may be utilized to develop customized marketing efforts and adjust supply levels in order to fulfill client demand. Machine learning may be used by banks and other financial organizations to detect fraudulent conduct, such as strange patterns of behavior or shifts in spending habits. Deep learning can be utilized to create systems that automatically transcribe speech, which has a wide range of applications ranging from customer service to medical records transcription.

Artificial intelligence that is embodied in a robotic body is referred to as embodied AI. Instead of being bound to a computer screen, the AI may interact with its environment in a more natural way. Embodied AI has shown to be useful in a variety of industries, including healthcare, and education. It can be utilized in healthcare to help doctors conduct surgery or offer physical therapy to patients, it can

be used in education to provide students with interactive learning experiences. And it may be employed in manufacturing to automate operations and increase productivity.

Robots can increasingly be employed in healthcare in the future. They'll be utilized to help doctors and nurses provide care to patients and complete activities that are difficult or risky for humans. In addition, robots can be utilized to assist in the care of the aged and crippled. In the healthcare industry, the employment of robots will result in improved patient care, fewer mistakes, and reduced costs. Nurses and physicians will be able to spend more time with their patients and focus on more complicated situations as a result of this.

Embodied AI is still in its early phases of development, and there are several obstacles to overcome. For instance, how will AI manage difficult jobs that involve human intuition and judgment? Also, how will we assure that AI interactions with humans are safe and do not cause harm? Despite the obstacles, embodied AI has a lot of potentials and might change a lot of sectors.

The Internet of Things (IoT) is a network of physical devices, automobiles, household appliances, and other items that are integrated with electronics, software, sensors, and connections, allowing them to communicate and interact. The link between IoT, deep learning, and healthcare has become more prominent in the past decade. The purpose of these initiatives is to improve the efficiency and patient-centeredness of the healthcare system.

IoT can assist in the collection of data from various devices and sensors, which can subsequently be used by machine learning algorithms to enhance healthcare quality. IoT, for example, can assist detect early indicators of sickness or deterioration by monitoring patients' vital signs and activity levels, which can subsequently be handled by the healthcare system. These technologies have enormous potential to change healthcare. More tailored and preventative care, as well as more efficient and effective disease treatment, can be expected in the future.

As the world becomes increasingly digitized, so too does the healthcare industry. As the healthcare industry becomes more digitized, artificial intelligence (AI) will play an increasingly essential role. AI may be applied in healthcare in a variety of ways, from automating administrative work to performing difficult surgery. And as AI technology advances, so too do the possibilities for how it can be used to improve patient care. The field of diagnostics is one area where AI is already having a significant influence. Human

doctors can't examine large volumes of data as rapidly or precisely as AI-powered diagnostic systems can. As a result, more people will be identified earlier and receive the care they require sooner.

Deep learning is also being employed in the development of novel and more effective illness treatments. By analyzing large data sets, it can identify patterns that may be missed by human doctors. This knowledge can subsequently be applied to the development of new treatments or the improvement of current ones. The application of artificial intelligence in healthcare is still in its infancy. But as the technology continues to develop, it is likely to have an increasingly transformative impact on the way healthcare is delivered.

1.2 Challenges

As mentioned, Deep Learning and Machine Learning models can get only as good as the data they're given. However, this is a big challenge in the healthcare industry since the collection of data and labeling them is often very expensive and time-consuming and as a result, not many big and properly labeled datasets exist for Deep Learning researchers. Also, different clinics use different devices with varying configurations and if the data is collected from varying sources, then adequate preprocessing techniques need to be applied to make the data standard and usable. Additionally, the decisions about the malignancy of the disease are often very subjective and depend on the doctors, hence even the hardly collected labels need to be further analyzed for possible outliers.

Transfer Learning along with Data Augmentation, the process of artificially creating new data points from existing data, are the current best solutions for tackling the aforementioned issues, as they can help produce high accuracy results even with small datasets, thanks to transfer learning ability to benefit from the knowledge it has previously collected from a similar task and on a different but huge dataset.

Furthermore, Machine Learning and more specifically Deep Learning models are extremely difficult to trace and due to their complexity, understanding the reasoning behind their decisions is challenging. That's why they are often referred to as black boxes. This lack of transparency can make it difficult to trust the model, improve the model, or use the model for critical applications. This might be alright in many fields, but in healthcare, there is no room for errors, and understanding the reasoning behind choices or predictions, especially if they are in conflict with the medical team is a must, and failing to do so can lead to biases and incorrect outcomes, which can be disastrous in healthcare.

1.3 Contributions

Diabetes is a long-term illness that impairs your body's sugar metabolism. When you have diabetes, your body produces insufficient insulin or is unable to adequately use the insulin it does produce. This causes sugar to build up in your blood, which can lead to a host of serious health problems including heart disease, stroke, kidney disease, blindness, and amputation.

Diabetic Retinopathy is a condition that can lead to blindness and is caused by damage to the blood vessels in the retina. It is the most common form of diabetic eye disease. It has several stages of malignancy. If diagnosed early, it can be treated with laser surgery, which can stop the progression of the disease and prevent blindness.

Diabetic retinopathy causes different types of deficiencies in the patients' eyes, such as cotton wool spots, exudates, microaneurysms, hemorrhages, and abnormal growth of blood vessels. Due to the lack of datasets, our main focus in this dissertation would be on exudates and microaneurysms, but all the other deficiencies can easily be incorporated into our dynamic architecture should new datasets become available in the future.

These deficiencies, especially the ones which corresponded to microaneurysms are extremely small and only consisted of a handful of pixels (between 1 to 5) as opposed to the total image dimension of 1024x1024. This would be a challenging task for doctors to spot on an everyday basis and it is also a tremendously hard task for deep learning models to tackle. Hence, our main contribution to this dissertation would be to design, implement, and test a complete framework that is able to detect everything about lesions via a single model including their bounding boxes and exact locations, lesions masks and approximate shapes, their type, a confidence score for each of them, along with the overall severity and malignancy of the instances per image taken for the patient. However, this complete framework cannot be implemented in one go. Hence, we employ the incremental approach of software development and create the framework in three milestones.

First, we propose DRDr, a deep learning model derived from a complex Convolutional Neural Network, namely Mask RCNN, that could take a fundus image as the input, and output all the instances of microaneurysms and exudates present in it, their position in addition to their exact shapes as separate black and white binary masks, and a confidence score for each of them, all in near real-time. The datasets

available for this phase are extremely small, hence we utilize transfer learning and data augmentation in our research.

Next, we propose DRDr II, a hybrid of machine learning and deep learning approaches that uses DRDr as a feature extractor in the core of its pipeline, and with that, it becomes able to tackle the overall malignancy problem and classify patients into three severity groups.

Finally, in DRDr V3, we combine the previous separate networks and use LSTM to engineer a new unified model that can generate the entire output using just one fundus image as the input and tackle more tasks simultaneously. One of the most important characteristics of our framework is that it provides different insights as to why a decision is made, such as an attention mask for the image that points to the most important locations in that picture, a list of all the lesions, their type, location, and the model's confidence score and much more, altogether making our framework interpretable and far from a black box. Also, the model is fast enough to be used either as a real-time plugin in medical devices or be used as a post-processing software when the picture is taken.

1.4 Dissertation Outline

Following our discussion about the biases, in **chapter 2**, we elaborate on the common practices that cause them, and how they can lead to misleading statistics and research results irreproducibility. It sets a valuable baseline and familiarize us with a group of bad practices to avoid during our entire research study in order to not include any unwanted biases in our models and results.

In **chapter 3**, we start to get technical and start to explore embodied AI. Here, we talk about the different types of smart agents, and what they are capable of such as embodied and interactive question answering, and multi-agent systems. We also touch base with different modern simulators that are currently being used for training and testing purposes and discuss the future of the field.

We will then talk about the Internet of Things in **chapter 4** and start to focus more on the advantages that it can bring to our healthcare system as a whole. We will discuss wearables and implants, and propose a pipeline with 5 stages that starts with raw data as the input, preprocess and store the data locally, securely transmit the data to powerful centralized analyzing centers, and properly standardize and store them

for future usability after acting on the model output. We will also provide some examples and provide potential solutions for the current challenges.

In **chapter 5**, we will see that the data collected from the healthcare industry is considered as big data. Hence, we will explore different machine learning and deep learning models that help researchers successfully extract useful information and patterns in an accurate and timely manner. We will provide examples for each of these models and also go one step further and touch base with the advanced applications of these models in the healthcare industry such as Computer Aided Diagnosis (CAD). Current challenges and their proposed solutions will be explained as well.

Next, we employ all the lessons learned until now and in **chapter 6** , conduct a big data analysis on the impact of weather conditions on the COVID-19 pandemic in the United State. Also, we engineer a unique multi-modal deep learning model for speaker recognition. It will later, serve as a building block for our main contribution as it will also make use of several parallel branches in its model.

Then, we design, implement, and test our main contribution, the interpretable, complete lesion detection framework for diabetic retinopathy, the so-called Diabetic Retinopathy Doctor (DRDr) in three milestones in **chapters 7, 8, and 9**.

Finally, we summarize this dissertation in **chapter 10** and discuss its potential for future work.

Chapter 2

Causes of Misleading Statistics and Research Results Irreproducibility [1]

[1] F Shenavarmasouleh and HR Arabnia. 2019. International Conference on Computational Science and Computational Intelligence (CSCI). 465-470.

Reprinted here with permission of the publisher.

For the past two centuries, the annual number of articles and journals have both kept growing at a steady rate of about 3% and 3.5% respectively. But, unsurprisingly, the increase in the number of researchers has accelerated this growth over the last few years. About 2.5 million articles get published each year worldwide [260] and yet, only a tiny proportion of their results are reproducible. In fact, more than 70% of researchers claimed that they failed at reproducing another person's work and even more frighteningly, over 50% could not reproduce their own work again, altogether leading to a reproducibility crisis [18].

One may assume that unproducible papers could only be found in poor-quality journals, but unfortunately, that is not the case. Many of these papers are written by very successful researchers and are published in top-gear journals of their fields. Given the fact that no good researcher and journal intend to present false or erroneous information to its audience, the question then arises as to why this incident happens. Knowing that the data do not speak for themselves and they have to be interpreted, it can be safely concluded that the main cause for this is statistics and it can affect the researches' outcomes at multiple levels.

Statistics is an essential part of every research, however, not much effort is being put into its proper education. Very few university majors require their students to take statistics courses and therefore many end up learning it via some available resources themselves. This could lead to a tremendous amount of problems, such as not using the correct statistical methods or graphs for a certain problem or make use of the ones which are not robust to the noises and outliers existing in the dataset.

Sadly, there exists a bigger issue. Even if the paper itself is a hundred percent accurate, the words which get used to describe it can be fallacious. This could simply be unintentional and as before, purely due to bad education, but the numerical data can also be intentionally misused. Truth is not everyone's first priority. Thousands of news get published each day, many reporting a concern seeking to get resolved by politicians. Hence, they must compete in order to get noticed by the public and catch the attention of policymakers and if one succeeds, it consequently gets the biggest share of their time and budget while the others are left disregarded. As alarming news can bring more audiences to the corresponding media, and it means benefits, the people in charge do not hesitate to make use of such a thing. Even if it means that they have to falsely create them and statistics is the key component in this process. But, every move has consequences and propagandas like fake news can alter or even derail government programs and policies, leading to unforeseen circumstances.

This chapter aims to address and classify the causes which make research papers unreproducible and explain how paper results could get reported misleadingly. We intend to look at these challenges as an opportunity to inform our audience. With this goal in mind, we hope that this manuscript would offer the readers, an insight and a better understanding of the causes of the problems; thus helping them not to become too reliant on papers that suffer from fallacies. We conclude the paper by proposing Open Science as an important addition to every research.

2.1 Resources and Tools

Since statistics is not a mandated part of many majors in universities, researchers in those fields mostly choose to learn it by themselves using some books and after learning the fundamentals, they move on to make use of certain software tools to do the complex and time-consuming calculations for them or generate eye-catching visuals to aid them in presenting the results of their work.

2.1.1 Textbooks

Bland et al. [29] explain that most introductory statistic books that get published are written by authors who are experts in their fields but are rarely qualified to write about anything else. Those books tend to be more attractive to a large proportion of researchers of that certain field than a book authored by an experienced statistician, solely because the authors of those books share a common educational background with them and hence their words can be more comprehensible.

From time to time, such books originate incorrect ideas. Bland argues that there were some cases in which the authors had made an error and then used a false argument to justify it. The horrifying part of this is that the people who read these books are unlikely to have enough knowledge to detect these flaws and consequently these errors keep propagating.

2.1.2 Software Tools

With the extensive use of computers, numerous software packages became accessible for research. Although they are massively useful and can save much time, using them without full understanding can be

dangerous. Many statistics' operations have multiple versions, probably for different applications. For example, standard deviations could be calculated with n instead of n-1 in its denominator or it is incorrect to use unpooled variance instead of pooled variance in t-tests. These little differences could lead to getting significant results while there exists none [29].

2.1.3 Graphs

Graphs are considered great tools for visualizing massive complex data and their simplicity helps understand the contents better, but when it comes to statistics being misleading, graphs are the root of all evil. There are numerous types of graphs to choose from and a poor choice can result in a misleading visual. Much worse, they could easily be manipulated in a way to give the impression of a much better result.

Every misleading graph violates at least one of the followings:

- A graph should always be equally spaced on all of its axes.
- If an axis has to show a quantity, it needs to start from zero. Otherwise, it fails at picturing the correct relative increase or decrease in the value.
- If using a pictograph, all symbols should be equally sized and the value which is being represented per symbol should be explicitly written.

2.1.4 Maps

Maps are used to illustrate spatial distributions of data such as cancer rates by county. However, if the sample size differs from area to area, the result can be highly deceptive especially in poorly-sampled areas. In addition, any adjustment such as using posterior means can cause further problems and therefore some spurious patterns could be found while in reality, they do not exist. Gelman et al. [75] state that although employing multiple imputed maps could help, they are still not appropriate for presenting the result as they may confuse the audience even more and hence they could not be generally used.

2.2 Data Collection

2.2.1 Broadened Problem Definition

The definition of the problem should always be specific and on point and this inevitably shrinks the size of the domain and decreases the resultant incident rates. It's common sense that bigger numbers are better and since these two are in conflict of interest with each other, it is not that rare to see definitions intentionally broadened to catch a bigger share of attention. For instance, in conducting a questionnaire for stranger abduction research, two different strategies could be used. One questionnaire could include short-term missings even if only for a few hours, attempted offenses which may not have even ended in actual abduction, to name a few, while the other only includes the children who went missing and found dead later on. The former results in around 15000 annual cases in the United States while the latter shows about 550 [24].

Back in the 90s, an exaggerated image of Australia's tourism was made showing it as a million-job industry, while in reality, the number of real jobs was around 200,000. In consequence, investors, business managers, politicians, and government were misled and by making wrong decisions, national efforts and resources were wasted and the job market was over-supplied. Leiper [142] claims that this was partially because of the broad definition of the word tourist. Its definition from WTTC, WTO and many other institutions include all kinds of visitors who intend to stay somewhere for one or more nights with varying purposes, such as enjoying a holiday, pilgrim, visiting family and friends, doing business, going to school, staying at hospitals, etc. However, if the definition was constrained to a more fair one which only factored in the visitors who were there for leisure, then the numbers would have been closer to reality.

2.2.2 Over-Discretized Sampling

Frequent sampling is a must when it comes to collecting helpful data. Nonetheless, when the time frames are longer than a standard threshold, the built dataset will fail to reveal valuable insights. As an example, hospital surge capacity, which is often measured by the number of empty beds that can get immediately ready to use in the case of an emergency, is usually reported annually and it fails to take the daily changes

in the number of patients and the within-year changes in bed supply into account. When measured daily, the result shows way less availability [50].

2.2.3 Biased Sampling

When you want to use a sample, you have to ensure that it can represent the larger population pretty well. The job of sample designs is to guarantee such behavior. If a design tends to favor a specific outcome, then it's considered a biased bad design. Biased samples can be created when individuals themselves choose to be involved in the research. The reason for this would be because the opinion of people who were not interested or simply didn't care enough has not been collected. Also, interviewing people only in specific places, such as streets, bars, libraries, and gyms or even same places in different cities can produce completely different results. To avoid biases, some good sampling methods have been developed, including but not limited to Simple Random Sampling (SRS), Stratified Random Sampling, Cluster Sampling, Systematic Sampling with Random Start, and Multistage Sampling [155].

2.3 Data Analysis and Statistical Methods

Gross domestic product (GDP) is one of the most popular indicators used to measure a country's economic health [89]. It depends on factors such as consumer spending, government spending, businesses' capital spending, and nation's total net exports while each of these, rely on many other elements related to the goods and services which the nation provides. Because of this innate complexity, calculation of GDP is an extremely hard and time-consuming process. In fact, too complex which even though nations' expert statisticians have the job of doing the calculations, from time to time, the results get revised multiple times afterward [87]. As an example, the U.S. Bureau of Economic Analysis released three different GDP rates for the second quarter of 2015 [88].

The good news is not all statistics problems have that much complexity in terms of the numbers of factors present in them and can be calculated and analyzed quite simply. But, using the right methods is a must. Otherwise, the output can lead to misleading deductions.

2.3.1 Common Scales

It's necessary to have a common scale in order to report and compare new results with baselines, and since there are usually multiple scales and methods to choose from, we should first fully understand the advantages and disadvantages of every single one of them and choose the best one for our work.

Results of clinical trials can be compared using several approaches. The most popular ones are Relative Risk (RR), Absolute Risk Reduction (ARR), Odds Ratio (OR), Log OR and Number Needed to Treat (NNT). It is common to use NNT as the default method because it's easier to comprehend but [109] refutes that this measurement is biased. NNT lacks precision as it rounds the result to its nearest integer which consequently can blur the differences among trials, it loses the time dimension and it has some fundamental issues such as the absence of a value that corresponds to no difference [110]. Hutton [109] further explains that ARR is a lot more reliable and should be used instead.

2.3.2 P-Value and Power

Technically speaking, P-value is the probability of getting a result, assuming the null hypothesis is true [164]. Put in simple words, the null hypothesis is the theory that we want to reject, that is our experiment has no effect, and the alternative hypothesis is the opposite. After calculating the P, it is compared with a threshold, usually 0.05, and if it's above the cut line, the null hypothesis remains true. But, if P-value falls below the cut line, it means that either the null hypothesis is false, or although true, a very rare event has happened, both indicating that our experiment result is significant. There are some well-known downsides to using P-value. First, even though we know the smaller the P-value, the more significant the result is, it cannot be used to show the scale of the difference between null and alternative hypothesis. Second, if we fail to reject the null hypothesis, it does not indicate that our experiment has no effect whatsoever. It just tells us that there's not enough evidence that there is one. Thus, we can never be entirely sure. Third, the probability of the null hypothesis being true remains unclear as we have already assumed that it's true.

Besides these innate disadvantages, there exists a bigger problem. P-Hacking [99] is the action of intentionally manipulating the P-value or the experiment to reach a significant result. The simplest form of it is to choose a bigger threshold than the resultant P and somehow justify it. Next is to divide the experiment into several smaller ones (n) and check all of them individually with the hope of finding a

desirable P-value in at least one of them. This causes family-wise error and the problem is as the P threshold (α) remains untouched and the same as the original experiment in all of them, the probability of getting at least one significant result just by chance would be $1 - (1 - \alpha)^n$. Then usually, only the significant experiment gets published without providing any context of the bigger picture which is clearly misleading and there's a huge chance that the significance is only the result of a Type I error and hence, most probably it cannot be reproduced. Bonferroni correction has to be used in order to avoid family-wise error and it suggests to use $\frac{P}{n}$ as the new threshold for each of the smaller experiments [261].

In this context, type II error means failing to reject the null hypothesis when it's false. While α corresponds to getting a false positive or being a victim of type I error, β is the probability of Type II error or getting a false negative. Power is the probability of detecting an effect when there exists one and its value is $1 - \beta$. In a certain problem $\alpha + \beta$ equals to a fixed number. So, it can be concluded that there's a trade-off between them and depending on the problem, the researcher have to decide which error should be valued more. Increasing sample size can do good to both of them as it makes the distributions more accurate and thinner and as a result reduces the possibility of both errors, by making that fixed number smaller and resulting in a bigger power and a more precise P-value.

2.3.3 Robust Statistics

Classical statistics, also called parametric statistics, require data to be normally distributed. Even though there is no guarantee that the data in smalls samples are normally distributed, more than often it is assumed that they are and the calculations are done based on that. One alternative is to use non-parametric statistics that do not have this requirement, but they have their own set of rules as well. Using parametric statistics on not normally distributed data and using non-parametric statistics on normally distributed data both result in less power [136]. As aforementioned, one way to increase the power, which happens to be the hardest way as well, is to increase the sample size. One alternative is to value Type II error more and by raising α make the β smaller and as a result get a higher power. Second alternative is to use a more powerful statistics, namely robust statistics [263]. It proposes principled ways to overcome parametric and non-parametric issues. For instance, it suggests to use effect size and confidence intervals in addition to P-value since unlike the latter, the result of those stay the same if α changes [164].

Outliers can almost be found in every population. If ignored, they can violate the assumption of normal distribution and result in less power, and if removed manually they can cause non-Independence in remaining data, and in some cases, they could be hard to be found in the first place. One solution that robust statistics proposed is to trim both ends and readjust the normal equation to compromise the effect of non-independence. It also recommends ways to do t-tests, correlations, 1-way ANOVA, etc. in a robust manner [136]. It's worth mentioning that robust statistics might result in a different outcome in terms of significance and non-significance of the research from the conventional statistics but it's most certainly more accurate.

2.4 Fallacious Deductions

Statistics is a subset of mathematics and it should not come as a surprise that understanding, mapping and modeling the concepts may not be a straightforward process. Even experts can find contradictory results with each other and at times such as in the case of Sally Clark [181], the deductions that have made based on those results could be a matter of life or death.

Misinterpretations can affect future research, resources, and fundings as well. For instance, while analyzing the bullets spread on returned American planes at World War II, it has been noticed that most bullets were around the fuselage and least around the engines. The initial reasoning was that in order to increase the survival rate, more armor should be placed on the fuselage area. Wald [254] refuted that conclusion and explained that the reason most planes had more bullet hits on their fuselage was that the ones which got hit on their engines could not eventually make it home and as a result the distribution was uneven. Thus, to increase the chance of survival, more armor is needed on the engine surfaces.

2.4.1 Confusing Correlation With causality

Scatter plots and covariance matrices are used to find relationships between pairs of features in the data. If their relationship is linear, Pearson's correlation ($-1 \leq r \leq 1$) [218] can be useful and just by looking at its sign and value, we can find out that whether the correlation is positive or inverse and how strong it is. If a strong correlation is found, it could mean two things. First, X causes Y or Y causes X. Second, either there is another reason Z that causes both X and Y or they are related by complete coincidence. The latter

is called spurious correlation [226] and means even though these two features seem to correlate with each other, there's no way one can cause another and this confusion between correlation and causality can lead to erroneous conclusions. Besides, not all correlations are linear and therefore scatter plots should always be analyzed.

2.4.2 Ecological Fallacy

Ecological fallacy happens if a deduction about an individuals' characteristic is made based on the results found for a group in which that person belongs to [69]. Data are collected at different levels such as a continent, a country, or a state. As previously discussed, the features can be analyzed in terms of correlation and then lead to discoveries. However, these findings are only valid in the level in which they are being analyzed and if any deduction is made based on them for lower-level groups or individuals, it cannot be trusted.

2.4.3 Will Rogers Phenomenon

Assume that there are two groups and all the people in group A have lower IQ than all the people in group B. If the dumbest person in group B moves to group A, the average IQ for both groups rises since B will be got rid of its least intelligent person who was dragging the average down and A will profit from gaining its smartest member. This effect is called the Will Rogers phenomenon and has caused some erroneous conclusions especially in the area of cancer screening [82]. Feinstein [64] elaborates the very first instance of this phenomenon by showing that improvements reported in the rates of survival of lung cancer patients are flawed. He explains that patients were originally divided into two groups. The ones who their cancer was localized and the ones with higher stages of cancer and metastasis. With the advancements in the technology, micro-metastases could be found in people from the localized group moving those patients to the other group. This simply improved the survival rates in both groups since with this migration, the first group has lost its patients with lower life expectancy and the second group earned a few patients with better overall health than its usual patients.

2.5 Indecent Reporting

When reporting the result, it's conventional to use percentages, but they can be highly misleading as they do not give any information about the initial and end value or at least the difference that has been made. A bigger picture must always be clearly drawn to better understand how significant the new result is. Also, the results should never and under no circumstances, get rounded to their nearest bigger number just to attract more attention. Another bad practice is hiding or not providing enough context to the audience, in order to mislead or fool them into believing that the results were significant. An example of this as aforementioned, could be not using Bonferroni correction and not talking about the original experiment at the same time. Sowell [235] also argues about a few cases in household reports. Such as not mentioning the change that has occurred in the number of people in families while comparing it to old statistics. Assume that in the past each household had 6 people in it and at a later year this number is decreased to 4. Even if the income per person is increased by 25 percent, the total sum shows an economic decline since it equals the amount that 5 people could make in past.

2.6 Open Science as a possible solution

Open Science is a group of ideas and principles promoting openness and transparency of the research in every stage of its life cycle while keeping the integrity of the researcher's work. It starts with sharing the main idea with the intention of inducing collaboration with other researchers, funders, publishers, industries and institutions and then quickly advancing the research to its final stage. Every note, data, code, software and the end result including the paper should be publicly available to everyone and ready to be reused. The long term goal of open science is to encourage good and high-quality research regardless of its outcome by valuing the research process more than its result and judge researchers based on their work and contribution to the community instead of completely relying on formal publications. Open science allows and even inspires researchers to replicate other researches and also publish the negative results which it believes can be very informative. This acts against the public and perish culture [22] which motivates researchers to use biases, P-hacking and all the other bad practices mentioned in this chapter to advance their career and instead enable researchers to do researches based on their joy and curiosity [182].

2.7 Conclusion and Final Remarks

A tremendous amount of data resides in every research. It's almost impossible to analyze and find something meaningful from that data without using statistics. However, if used incorrectly, either intentionally or unintentionally, it can result in erroneous and misleading conclusions and thus make the research unreproducible. It is important to know the common causes of this incident to avoid it. In this review, we attempted to summarize the most common flaws that can happen while using statistics. This sets a valuable baseline and familiarize us with a group of bad practices to avoid during our entire research study in order to not include any unwanted biases in our models and results that will be discussed in next chapters.

Chapter 3

Embodied AI-Driven Operation of Smart Cities [1]

[1] F Shenavarmasouleh, FG Mohammadi, MH Amini, and HR Arabnia. 2022. Cyberphysical Smart Cities Infrastructures: Optimal Operation and Intelligent Decision Making. 29-45.

Reprinted here with permission of the publisher.

A smart city is an urban area that employs Information and Communication Technologies (ICT) [189] , an intelligent network of connected devices and sensors that can work interdependently [8], [172] and a distributive manner to continuously monitor the environment, collect data, and share them among the other assets in the ecosystem. A smart city uses all the available data to make real-time decisions about the many individual components of the city to ease up the livelihood of its citizens, and make the whole system more efficient, more environmentally friendly, and more sustainable [7]. This serves as a catalyst for creating a city with faster transportation, fewer accidents, enhanced manufacturing, more reliable medical services and utilities, and much more. The good news is any city, even with traditional infrastructures, can be transformed into a Smart City by integrating IoT technologies.

An undeniable part of a smart city is its use of smart agents. These agents can vary a lot in sizes, shapes, and functionalities. They can simply be light sensors that along with their controller act as the energy-saving agents or could be more advanced machines, with complicated controllers and interconnected components that are capable of tackling more advanced problems. The latter agents usually come with an embodiment with numerous sensors and controllers built in them that enable them to perform high-level and human-level tasks such as talking, walking, seeing, and complex reasoning along with the ability to interact with the environment. Embodied Artificial Intelligence is the field of study that takes a deeper look into these agents and explores how they can fit into the real-world and how they can eventually act as our future community workers, personal assistants, robocops, and much more.

Imagine arriving home after a long working day and seeing your home robot waiting for you at the entrance door. Although it is not the most romantic thing ever, you then walk up to it and ask it to make a cup of coffee for you and also add two teaspoons of sugar if there is any in the cabinet. For this to become reality, the robot has to have a vast range of skills. It should be able to understand your language and be able to translate questions and instructions to the action. It should be able to see its surroundings and have the ability to recognize objects and scenes. Last but not the least, it must know how to navigate in a big dynamic environment, interact with the objects within it, and be capable of doing long-term planning and reasoning.

In the past few years, there has been significant progress in the fields of computer vision, natural language processing, and reinforcement learning thanks to the advancements in deep learning models. Many things are now possible because of these that seemed impossible a few years ago. However, most of

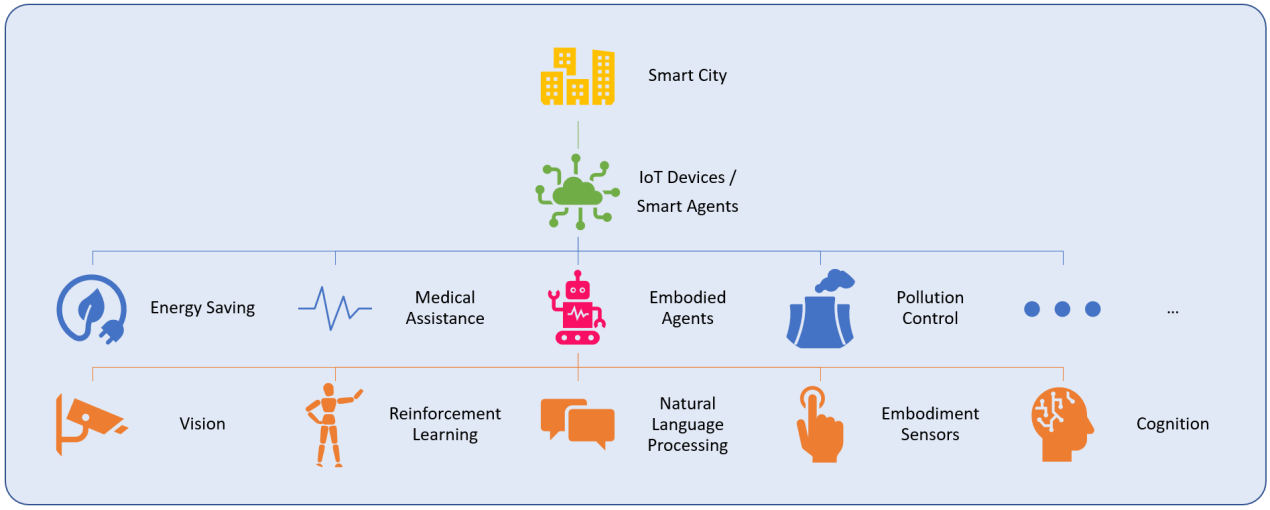


Figure 3.1: Embodied AI in Smart Cities

the work has been done in isolation from other lines of work. Meaning that the trained model can only take one type of data (eg. image, text, video) as the input and perform a single task that it is asked for. Consequently, such a model act as a single-sensory machine as opposed to a multi-sensory one. Also, for the most part, they all belong to Internet AI rather than Embodied AI. The goal of Internet AI is learning patterns in text, images, and videos from the datasets collected from the internet.

If we zoom out and look at the way models in Internet AI are being trained, we realize that generally supervised classification is the way to go. For instance, we provide a certain number of dog and cat photos along with the corresponding labels to a perception model and if the number is large enough, the model then can successfully learn the differences that exist between these two animals and discriminate between them. Learning via flashcards falls under the same umbrella for humans.

Extensive amount of time has been devoted in the past years to gather and build huge datasets for the imaging and language communities. A few considerable markers of this can be IMAGENET [51], MS COCO [148], created for vision tasks; Glue [255], Swag [281] built for language objectives; and also Visual Genome [132] and VQA [12] datasets created for joint purposes to name a few.

Apart from playing a pivotal role in the recent advances of the main fields, these datasets also proved to be useful when used with transfer learning methods to help underlying disciplines such as biomedical imaging [222], [223], [225]. However, the aforementioned datasets are prune to restrictions. Firstly, at

times it can get extremely costly, both in terms of time and money, to gather all the required data for the collection and label them. Secondly, the collection has to be monitored constantly to assure that they follow certain rules to avoid creating biases that could lead to erroneous results in future works [221] and also make sure that the collected data are all normal and uniform in terms of attributes such as background, size, position of the objects, lighting conditions, etc. while in contrast, we know that in real-world scenarios this cannot be the case and robots have to deal with a mixture of unnormalized noisy irrelevant data plus the relevant well-curated ones. Additionally, the agent would be able to interact with the objects in the wild (e.g. picking it up and looking at the object from another angle) and also use its other senses such as smell and hearing to collect information.

Us humans, we do learn from interactions and it's a must for true intelligence in the real world. In fact, it's not only humans and all the other animals do the same. In, kitten carousel experiment [101], Held and Hein exhibited this beautifully. They studied the visual development of two kittens in a carousel over time. One of which had the ability to touch the ground and control its motions within the restrictions of the device while the other was just a passive observer. At the end of the experiment, they found out that the visual development of the former kitten was normal whereas for the latter one it was not, even though they both saw the same thing. This proves that being able to physically experience the world and interact with it is a key element for learning.

The goal of Embodied AI is to bring the ability to interact and being able to use multi senses simultaneously into play to enable the robot to continuously learn in a lightly supervised or even unsupervised way in a rich dynamic environment.

3.1 Rise of the Embodied AI

In the mid-1980s a major paradigm shift took place towards embodiment and computer science started to become more practical than theoretical algorithms and approaches. Embedded systems started to appear in all kinds of forms to aid humans in everyday life. Controllers for trains, airplanes, elevators, air conditioners, and Softwares for translation and audio manipulation are some of the most important ones to name a few [106].

Embodied Artificial Intelligence is a broad term, and those successes were for sure great ones to start with; yet, it could clearly be seen that it was a huge room for improvement. Theoretically, the ultimate goal of AI is not only to master any given algorithm or task that is given to, but also gain the ability to multitask and get to human-level intelligence, and that as mentioned requires meaningful interaction with the real world. There are many specialized robots for a vast set of tasks out there, especially in large industries, which can do the assigned task to them to perfection, let it be cutting different metals, painting, soldering circuits, and much more, but until one single machine emerges to have the ability to do different tasks or at least a small subset of them by itself and not just by following orders, it cannot be called intelligence.

Humanoids are the main thing that comes to mind when we talk about robots with intelligence. Although it is the ultimate goal, it is not the only form of intelligence on the earth. Other animals, such as insects have their own kind of intelligence and due to being relatively simpler in comparison to humans, they are a very good place, to begin with.

Rodney Brooks has a famous argument that says it took the evolution much longer to create insects from scratch than getting to human-level intelligence from there. Consequently, he suggested that these simpler biorobotics should be first dealt with in the road to make much more complex ones. Genghis, a six-legged walking robot [32] is one of his contributions to this field.

This line of thought was a fundamental change and led researchers to have a change of direction in their work and with that came attention to new domains and topics such as robotics, locomotion, artificial life, bio-inspired systems, and much more. The classical approach did not care about tasks related to interaction with the real world and consequently, locomotion and grasping were the ones to start the journey with.

Since not much computational power was available at the time of this shift, a big challenge for the researchers was the trade-off between simplicity and the potential to operate in complex environments. An extensive amount of work has been done in this area to explore or invent ways to exploit natural body dynamics, materials used in the modules, and their morphologies to make the robots move and become able to grasp and manipulate items without sophisticated processing units [111]. It goes without saying that the ones who could use the physical properties of themselves and the environment to function were more energy-efficient, but they had their own limitations. Not being able to generalize well to complex environments was a major drawback. But, they were fast as the machines with huge processing units

needed a reasonable amount of time to think and plan their next action and often move their rigid and non-smooth actuators.

Nowadays, a big part of these issues are solved and we can see extremely fast and smooth natural moving robots capable of doing different types of maneuvers [30], but yet it is foreseen that with the advances of artificial muscles, joints, and tendons this progress can be further improved.

3.2 Breakdown of Embodied AI

In this section, we try to categorize a broad range of research that has been done under the field of Embodied AI. Due to the huge diversity, each section will necessarily be abstract, selective, and reflect the authors' personal opinion.

3.2.1 Language Grounding

Machine and human communication has always been a topic of interest. As time goes on, more and more aspects of our lives are controlled by AIs, and hence it is crucial to have ways to talk with them. This is a must for giving new instructions to them or receiving an answer from them, and since we are talking about general day to day machines, we desire this interface to be higher level than programming languages and closer to spoken language. To achieve this, machines must be capable of relating language to actions and the world. Language grounding is the field that tries to tackle this and map natural language instructions to robot behavior.

Hermann et al.'s work show that this can be achieved by rewarding an agent upon successful execution of written instructions in a 3D environment with a combination of unsupervised learning and reinforcement learning [102]. They also argue that their agent can generalize well after training and can interpret new unseen instructions and operate in unfamiliar situations.

3.2.2 Language plus Vision

Now that we know that machines can understand languages and there exist sophisticated models just for this purpose out there [243], it is time to bring another sense into play. One of the most popular ways to show the potential of joint training of vision and language is the image and video captioning.

More recently, a new line of work has been introduced to take advantage of this connection. Visual Question Answering (VQA) [12] is the task of receiving an image along with a natural language question about that image as an input and attempting to find the accurate natural language answer for it as the output. The beauty of this task is that both the questions and the answers can be open-ended and also the questions can target different aspects of the image such as the objects that are present in them, their relationship or relative positions, colors, and background.

Following this research, Singh et al. [228] cleverly added an OCR module to the VQA model to enable the agent to read the texts available in the image as well and answer questions asked from them or use the additional context indirectly to answer the question better.

One may ask where does the new task stands relative to the previous one. Do agents who can answer questions more intelligent than the ones who deal with captions or not? The answer is yes. In [12] the authors show that VQA agents need a deeper and more detailed understanding of the image and reasoning than models for captioning.

3.2.3 Embodied Visual Recognition

Passive or fixed agents may fail to recognize objects in scenes if they are partially or heavily occluded. Embodiment comes to the rescue here and gifts the possibility of moving in the environment to actively control the viewing position and angle to remove any ambiguity in object shapes and semantics.

Jayaraman et al. [116] started to learn representations that will exploit the link between how the agent moves and how it will affect its visual surrounding. To do this they used raw unlabeled videos along with an external GPS sensor that provided the agent's coordinates and trained their model to learn a representation linking these two. So, after this, the agent would have the ability to predict the outcome of its future actions and guess how the scene would look like after moving forward or turning to a side.

This was powerful and in a sense, the agent developed imagination. But, there was an issue here. If we pay attention we realize that the agent is still being fed pre-recorded video as the input and is learning similar to the observer kitten in the kitten carousel experiment explained above. So, following this, the authors went after this problem and proposed to train an agent that takes any given object from an arbitrary angle and then predict or better to say imagine the other views by finding the representation in a self-supervised manner [115].

Up until this point, the agent does not use the sound of its surroundings while humans are all about experiencing the world in a multi-sensory manner. We can see, hear, smell, touch all at the same time, and extract and use the relevant information that could be beneficial to our task at hand. All that said, understanding and learning the sound of objects present in a scene is not easy since all the sounds are overlapped and are being received via a single channel sensor. This is often dealt with as an audio source separation problem and lots of work has been done on it in the literature [70], [188].

Now it was the reinforcement learning turn to make a difference. Policies have to be learned to aid agents move around a scene and this is the task of active recognition [19]). The policy will be learned at the same time it is learning other tasks and representation and it will tell the agent where and how to strategically move to recognize things faster [247].

Results show that policies indeed help the agent to achieve better visual recognition performance and the agents can strategize their future moves and path for better results that are mostly different from shortest paths [275].

3.2.4 Embodied Question Answering

Embodied Question Answering brings QA into the embodied world. The task starts by an agent being spawned at a random location in a 3D environment and asked a question which its answer can be found somewhere in the environment. In order for the agent to answer it, it must first strategically navigate to explore the environment, gathers necessary data via its vision, and then answer the question when the agent finds it [46].

Following this, Das et al. [47] also presented a modular approach to further enhance this process by teaching the agent to break the master policy into sub-goals that are also interpretable by humans and execute them to answer the question. This proved to increase the success rate.

3.2.5 Interactive Question Answering

Interactive Question Answering (IQA) is closely related to the Embodied version of it. The only main issue is that question is designed in a way that the agent must interact with the environment to find the

answer. For example, it has to open the refrigerator, or pick up something from the cabinet and then and plan for a series of actions conditioned on the question [85].

3.2.6 Multi-Agent Systems

Multi-Agent Systems (MAS) is another interesting line of development. The default standpoint of AI has a strong focus on individual agents. MAS research which has its origins in the field of biology tries to change this and studies the emergence of behaviors in groups of agents or swarms instead [258].

Every agent has a set of abilities and is good in them to an extent. The point of interest in MAS is how a sophisticated global behavior can emerge from a population of agents working together. A real-life example of such behavior can be found in insects like ants and bees [33]. One of the interesting goals of this research is to ultimately make agents that could self-repair [179].

The emerging behavior of MAS can be tailored by researchers to let the group of agents tackle various tasks such as rescue missions, traffic control, fun sports events, surveillance, and much more. Additionally, when fused with other fields unexpected outcomes can occur. Take for instance "Talking Heads" experiment by Luc Steels [239] that showed a common vocabulary emerges through the interaction of agents with each other and their environment via a language game.

3.3 Simulators

Now that we know about the fields and tasks that Embodied AI can shine in, the question is how our agents should be trained. One may say it's good to directly train in the physical world and expose them to its richness. Although a valid solution, this choice comes with a few drawbacks. First, The training process in the real-world is slow, and the process cannot be sped up or parallelized. Second, it is very hard to control the environment and create custom scenarios. Third, it's expensive, both in terms of power and time. Fourth, it's not safe, and improperly trained or not fully trained robots can hurt themselves, humans, animals, and other assets. Fifth, in order for the agent to generalize the training, has to be done in plenty of different environments that is not feasible in this case.

Our next choice is simulators, which can successfully deal with all the aforementioned problems pretty well. In the shift from Internet AI to Embodied AI, simulators take the role that was previously

played by traditional datasets. Additionally, one more advantage of using simulators is that the physics in the environment can be tweaked as well. For instance, some traditional approaches in this field [56] are sensitive to noise and for the remedy, the noise in the sensors can be turned off for the purpose of this task.

As a result, agents nowadays are often developed and benchmarked in simulators [92] and once a promising model has been trained and tested, it can then be transferred to the physical world [207].

House3D [266], Gibson [267], MINOS [215] and Habitat [216] are some of the popular simulators for the Embodied AI studies. These platforms vary with respect to the 3D environments they use, the tasks they can handle, and the evaluation protocols they provide. These simulators support different sensors such as vision, depth, touch, and semantic segmentation.

In this chapter we mainly focus on MINOS and Habitat since they provide more customization abilities (number of sensors, their positions, and their parameters) and are implemented in a loosely coupled manner to generalize well to new multi-sensory tasks and environments. As their API can be used to define any high-level task and the material, object clutter variation, and much more can be programmatically configured for the environment. They both support navigation with both continuous and discrete state spaces. Also, for the purpose of their benchmarks, all the actuators are noiseless, but they both have the ability to enable noises if desired [48].

In the last section, we saw numerous task definitions and how they each can be tackled by the agents. So, before jumping into MINOS and Habitat simulators and reviewing them, let's first get more familiarized with the three main goal-directed navigation tasks, namely, PointGoal Navigation, ObjectGoal Navigation, and RoomGoal Navigation.

In PointGoal Navigation, an agent is appeared at a random starting position and orientation in a 3D environment and is asked to navigate to target coordinates which are given relative to the agent's position. The agent can access its position via an indoor GPS. There exists no ground-truth map and the agent must only use its sensors to do the task. The scenarios start the same for ObjectGoal Navigation, and RoomGoal Navigation as well, however, instead of coordinates, the agent is asked to find an object or go to a specific room.

3.3.1 MINOS

Minos simulator provides access to 45,000 three-dimensional models of furnished houses with more than 750K rooms of different types available in the SUNCG [232] dataset and 90 multi-floor residences with approximately 2,000 annotated room regions that are in the Matterport3D [36] dataset by default. Environments in Matterport3D are more realistic looking than the ones in SUNCG. MINOS simulator can approximately reach hundreds of frames per second on a normal workstation.

In order to benchmark the system, the authors studied four navigation algorithms; three of which were based on asynchronous advantage actor-critic (A3C) approach [113] and the remaining one was Direct Future Prediction (DFP) [54].

The most basic one among the algorithms was Feedforward A3C. In this algorithm, a feedforward CNN model is employed as the function approximator to learn the policy along with the total value function that is the expected sum of rewards from the current timestamp until the end of the episode. The second one was LSTM A3C that used an LSTM model with the Feedforward A3C act as a simple memory. Next was UNREAL, an LSTM A3C model boosted with auxiliary tasks such as value function replay and reward prediction. Last but not the least, the DFP algorithm was employed that can be considered as Monte Carlo RL [230] with a decomposed reward.

The authors benchmarked these algorithms on PointGoal and RoomGoal tasks and found out that firstly, the naive feedforward algorithm fails to learn any useful representation; secondly, in small environments, DFP performs better while in big and more complex environments UNREAL beat the others.

3.3.2 Habitat

Habitat was designed and built in a way to provide the maximum customizability in terms of the datasets that can be used and how the agents and the environment can be configured. That being said, Habitat works with all the major 3D environment datasets without a problem. Moreover, it's extremely fast in comparison to other simulators. AI2-THOR and CHALET can get to an fps of roughly ten, MINOS and Gibson can get to around a hundred, and House3D yields 300 fps in the best case, while Habitat is capable of getting up to 10,000 frames per second. Habitat also provides a more realistic collision model in which if a collision happens, the agent can be moved partially or not at all in the intended direction.

To benchmark Habitat, the owners employed a few naive algorithm baselines, Proximal Policy Optimization (PPO) [217] as the representer of learning algorithms versus ORB-SLAM2 [167] as the chosen candidate for non-learning agents and tested them on the PointGoal Navigation task on Gibson and Matterport3D. They used Success weighted by Path Length (SPL) [10] as the metric for their performance. The PPO agent was tested with different levels of sensors (e.g. No visual sensor, only depth, only RGB, and RGBD) to perform an ablation study and find the proportion in which each sensor helps the progress. SLAM agents were given RGBD sensors in all the episodes.

The authors found out that first, PPO agents with only RGB perform as bad as agents with no visual sensors. Second, all agents perform better and generalize more on Gibson rather than Matterport3D since the size of environments in the latter is bigger. Third, agents with only depth sensors generalize across datasets the best and can achieve the highest SPL. But most importantly, they realized that unlike what has been mentioned in the previous work, if the PPO agent learns long enough, it will eventually outperform the traditional SLAM pipeline. This finding was only possible because the Habitat simulator was fast enough to train PPO agents for 75million time steps as opposed to only 5million time steps in the previous investigations.

3.4 Future of Embodied AI

3.4.1 Higher Intelligence

Consciousness has always been considered as the ultimate characteristic for true intelligence. Qualia [248] is the philosophical view of consciousness and it is related to the subjective sensory qualities like "the redness of red" that humans have in their mind. If at some point machines can understand this concept and objectively measure such things, then the ultimate goal can be marked as accomplished.

Robots still struggle at performing a wide spectrum of tasks effortlessly and smoothly, and this mainly due to actuator technology as currently mostly electrical motors are used. Advances in artificial muscles and skin sensors that could cover the entire embodiment of the agent would be essential to fully mitigate the human experience in the real world and eventually unlock the desired cognition [107].

3.4.2 Evolution

One more key component for cognition is the ability to grow and evolve over time [67]. It's easy to evolve the agent's controller via an evolutionary algorithm but it's not enough. If we aim to have completely different agents, we might as well give them the ability to evolve in terms of embodiment and the sensors as well. This again requires the above mentioned artificial cell organism to encode different physical attributes in them and flip them slightly over time. Of course, we are far from this to become reality, but it is always good to know the furthermost step that has to be done one day.

3.5 Conclusion and Final Remarks

Embodied AI is the field of study that takes us one step closer to the true intelligence. It is a shift from Internet AI towards embodiment intelligence that tries to exploit the multi-sensory abilities of agents such as vision, hearing, touch, and together with language understanding and reinforcement learning attempts to interact with real-world in a more sensible way. In this chapter, we tried to do a review of this field, and its current advancements, subfields, and tools hoping that this would help and accelerate future researches in this area. In the next two chapters, we will explore the Internet of Things and eHealth and we will disclose the relationship among these three.

Chapter 4

Machine Learning and Internet of Things (IOT) in Healthcare [1]

[1] F Shenavarmasouleh*, FG Mohammadi*, and HR Arabnia. 2021. International Conference on Computational Science and Computational Intelligence (CSCI). 1516-1522.

Reprinted here with permission of the publisher.

* The authors contributed equally to this work.

4.1 Introduction

4.1.1 Internet Of Things

Imagine being outside in a cafe and suddenly remembering that you forgot to turn off the stove before leaving the house. Traditionally, the only way to deal with this was to halt everything that you were doing at the moment and go back home to prevent hazardous scenarios. But, wouldn't it be easier if you could address the problem remotely? Well, you are in luck! Because in today's world Internet Of Things (IoT) has become a thing. IoT refers to a smart system of inter-related devices (computers, sensors, actuators, etc.) connected to the internet each with a unique identifier that are able to continuously communicate with each other with a common language over the network and collectively make intelligent decisions by analyzing the gathered raw data [15].

Nowadays, IoT is being used systematically to make human lives easier. This includes but is not limited to smart cities [224], smart farming [194], smart homes and grids [119], and smart healthcare [146]. Altogether, the automatic and ubiquitous nature of IoT increases the Quality of Service (QoS) and makes the entire system more efficient, more environmentally friendly, more sustainable.

4.1.2 IoT in Healthcare

Traditional monitoring in healthcare uses the resources and time insufficiently. A certified clinician needs to constantly check on the patient in-person and test results sometimes need days to get ready. Also, after hospital discharge, recovering patients may need to make a couple of appointments for the following check ups in order to make sure everything is on the right track with their health.

With the advent of IoT, these issues are being addressed in a systematic way. Wearables and implantable devices are being utilized to continuously monitor the health of the patients independent from the time of the day and their physical locations. In addition to this, these devices can make use of local and/or more powerful centralized artificial intelligence (AI) models to not only detect but also predict diseases and hazardous scenarios. They can then automatically alert the patient and the corresponding doctor. We talk about the system pipeline in more detail in section 4.2. Ubiquitous health (uHealth), electronic health (eHealth), and mobile health (mHealth) are all different names that cover the researches in this area and

their ultimate goal is to reduce the cost of healthcare, increase patients satisfaction, decrease the load of the hospitals especially in the event of a crisis, and provide easy to understand yet accurate and powerful AI models to aid doctors in detecting and preventing diseases and providing personalized treatments [222], [223], [225].

4.1.3 Big Data in IoT Health

Nowadays, millions if not billions of sensors are connected to patients that monitor and collect environmental, physical, physiological. and behavioral parameters non-stop. Recent trends also show the emergence of medical super sensors with more memory and processing power that can utilize Improved Particle Swarm Optimization algorithm to help accurate drug delivery to different organs of the human body to detect whether the drug has reached a particular position or not and much more [208]. As expected these sensors generate a massive amount of data each second. This huge amount of heterogeneous data that contains highly redundant and correlated info is called Big Data [209]. In the most naive way, all this data needs to be sent to a centralized server for feature extraction and analysis and this induces challenges such as network bottlenecks for transmitting the data and insufficient processing power and resources for real-time analysis of such data. Several solutions have been provided to solve the former problem such as removing redundant data and outliers in the local machine, aggregating the data before transmitting it, and trying to do a very basic analysis using light mobile AI models and only transmit data when the results hint to a problem [249]. Machine Learning and Deep Learning techniques are the most dominant method for processing, understanding, and extracting knowledge from the collected data and improving the decision-making process, since after their training phase is done, they do not require any further supervision and can automatically perform their task.

4.1.4 Machine Learning for Big Data in IoT Health

As mentioned above in section 4.1.1, machine learning and deep learning techniques have been applied and are being used in all sorts of smart systems. These smart systems are comprised of many smaller components, each providing a very different service. Hence, if we want to understand smart systems we need to break them down into their building blocks. Computer Networks, Computer Vision, Natural Language

Processing (NLP), Reinforcement Learning, and General Reasoning are among the main components. Each of these has its own separate world and an immense amount of ongoing research has been done over the past few decades. Hence, we would not go deep into them and try to touch on the most important architectures and algorithms that have been common in recent case studies.

In one broad view categorization, we can divide the machine learning algorithms into supervised and unsupervised techniques. In unsupervised techniques, the model receives unlabeled data, and hence its goal is to self-discover any sort of meaning and hidden patterns within the data to perform the data grouping. In more technical terms, this process is called clustering, and K-means, Fuzzy C-Means, Expectation-Maximization Algorithm, and Hidden Markov Model are some of the most prominent techniques. With supervised models, we can go one level deeper. They can further be split into classification and regression. In contrast to clustering, in classification, the model is given the labels in the training phase and it has to learn to categorize the input data in a number of pre-known classes with the least amount of error. K-nearest neighbor (KNN), Random Forest, C4.5 model, Naive Bayes, Support Vector Machine (SVM), Neural Networks, and Deep Belief Networks are a few of the most commonly used classifiers. When it comes to the images and videos, these classifiers cannot extract the best features all on their own and will be in need of help. Convolutional Neural Networks (CNN) address this issue using their pooling and convolutional layers and have been being used for any task performed on media since AlexNet [133] won the ImageNet image classification competition in 2012. They have been a tremendous amount of research on CNNs and the models that are being used nowadays are several folds stronger than the initial model. This is all thanks to researchers who thoroughly studied and analyzed CNNS which their efforts ultimately led to expanding and improving these networks. VGG [227], Inception [240], and ResNet [98] are some of the best CNNs that are being used recently to handle all kinds of vision problems, namely, Image classification, Object Localization, Object Detection, Semantic Segmentation, and Instance Segmentation. Also, if temporal features are of importance, then Recurrent Neural Networks (RNN) such as Long short-term memory (LSTM) [105] or Gated recurrent unit (GRU) [41] can be employed on top of the CNNs to learn the time-related patterns. CNN architectures mostly include a simple classifier such as KNN or a feed-forward neural network as their last few layers to perform the classification after successfully extracting features using their convolutional layers, but after the training, the default classifier

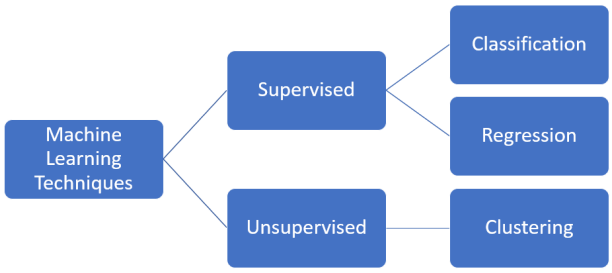


Figure 4.1: Overview of Machine Learning Techniques

can be detached and the convolutional layers can be solely used as feature extractors and they can be fused with other features and be fed to other classifiers or ensembles of them as needed [14].

While classification tries to assign a label to a given input, regression aims to predict a quantity in the continuous space. Hence, these models are useful for predicting numerical values. Linear regression, Support vector regression (SVR), and a properly structured neural network are among the to-go algorithms for this task.

4.2 IoT Pipeline

4.2.1 Sensing and Data Gathering

IoT aims to automate tasks that otherwise need to be tackled manually and by doing so improves the quality of lives of humans, and in the context of healthcare, this would mean ubiquitous monitoring and treatments of patients with improved responsiveness and accuracy. Advancements of portable microcontrollers and microprocessors such as Arduino and Raspberry pi, and the emergence of efficient, low-cost, yet accurate sensors and communication modules such as Zigbee that could easily be integrated with these devices led to the rise of wearables and implants. Wearables, as their name suggests, are products that could be worn as accessories. They come in various types, sizes, and shapes, but smartwatches, fitness trackers, bracelets, and smart rings are considered the most prominent ones. They can track and monitor heart rate, blood pressure, workout time, calories burned, sleep time, and much more. Implants, on the other hand,

Figure 4.2: Overview of IoT pipeline

are devices that have to be inserted into the body or placed under the skin of people. Implants are used for applications such as heart pacemakers [53], glucose monitoring [44], and spiral neurons [137]. Aside from monitoring vitals, implants usually have a more dedicated goal and aim to help a particular organ in the body to restore its biological system. For instance, in the case of glucose monitoring, the sensor measures the amount of glucose in the blood regularly and also has the ability to regularize it by injecting insulin from its embedded insulin tank as needed.

4.2.2 Data Storage and Preprocessing

Real-time transmission of gathered data is often costly and hence usually data collected by wearables and implants will be stored on a local memory on the device for initial analysis and preprocessing. The aforementioned models in 4.1.4 often require a lot of processing power and working memory and are not able to perform well on resources embedded in wearables and implants. So, lighter neural network models are employed to address this. They often sacrifice a bit of accuracy for faster and less resource-hungry computations and are getting better day by day. Trained versions of pruned or reduced neural networks [28] such as MobileNet [214], and EfficientNet [242] can be employed in such devices to perform general analysis on data streams and even video streams collected from small cameras on some implants.

4.2.3 Secured Transmission

With the presence of preprocessing models in devices, the transmission of data to centralized servers can be done in an asynchronous and passive way. These models analyze and keep track of patterns in streamed

data and if no anomaly is detected, then the data will be saved and transferred at a later time or on a fixed schedule when the network is less busy. But, if the models detect, suspect, or predict a problem with the underlying data, they can proceed and immediately transfer the required information for further analysis.

Even though preprocessing models help a lot, but still they have limited computational power and cannot do heavy analysis using the embedded battery and processor and without access to the data from other patients. Hence, the entire data needs to be transferred to the server at some point and that proposes two main problems. First, different sensors generate a lot of heavily correlated data, and altogether the size of generated data is big. So, it would be a good idea to reduce the size of data as much as possible before actually transferring them. Here, again different light local models can be incorporated to detect and remove outliers and noise and remove redundant data. The remaining data can be then get aggregated and compressed and broken down into chunks to be sent to the central server when the load is low [108]. There is also a lot that can be learned from swarm intelligence and how the data transmission is being addressed in that field [258].

Second, the transmitted data includes a lot of information about the patient, and secured transfer must be taken into account to prevent data leaks, eavesdroppers, DoS attacks, and much more caused by misconfigured network and communication devices. It goes without saying, that the data itself should be encrypted to maintain the confidentiality of the patients even in the event of network breaches. Hence, security standards and techniques are constantly being enforced to avoid these scenarios at all cost [114].

4.2.4 More in-depth Analysis

The main strength of central computing servers is their access to a relatively large amount of resources and their ability to make use of the computing power to deeply analyze and find patterns in the data. Here, the machine learning algorithms access to data goes above and beyond just one patient and they could leverage all the information from different patients at the same time by fusing them and trying to find patterns and relations on a much higher semantic level. Given their more horsepower, central server analytical frameworks can also merge any possible historical information that the patients may have from their clinical visits and/or by using different services and sensors from other providers, their demographic information, and other more complementary data before performing the analytics to get the complete picture more easily.

Deep learning and machine learning techniques are perfect for analyzing the data at hand, but sometimes we could only perceive them as black boxes, and even though this might not be an issue in many fields but it is not acceptable in healthcare. The results and the corresponding decisions that would be made using those models must be interpretable by doctors and clinicians since computers are here to aid the healthcare community and for making it possible to work side-by-side with them, they have to find a way to elaborate their complex decisions especially if they happen to be the incorrect ones.

Machine learning algorithms have an easier time achieving this since most of the time their logic is not inherently very complicated and is easier to follow. Decision trees for instance can easily illustrate their "decision tree" and show all the underlying conditions and attributes that lead them into making a particular decision. Even if the number of features is hundreds of dimensions, they are still ways to reduce the number of dimensions and show the results of methods such as KNN in relatively accurate 2-3D plots [177].

However, given the complex architecture of deep learning models, this would not be a trivial task and the decisions cannot be easily traceable. To tackle this issue, there is ongoing research on how to generate rules from neural networks and trying to find the most dominant paths in them to make some sense out of their decision [259]. Also, when it comes to processing images and videos, attention models can be used to visualize the areas that the model mostly focuses on while deciding on a particular problem [257].

4.2.5 Proper storage and Interoperability

The newly received data should be properly stored for future use cases; however, it is not as easy as dumping the entire data into a database. As mentioned, central frameworks can lend or borrow information from other central frameworks and services. Yet, there is no guarantee that the other servers use the same standards for data formats and some norm such as interfaces or database level rules and schemas needs to be incorporated. Hence, semantic interoperability is a must while dealing with multiple computer systems exchanging and using each other's information which if we look at it again is the core concept that the entire IoT is built on. Every data must have meta-information about different entities within it to give context to the corresponding values and bring in the possibility to connect and link these little pieces of data and do some automatic reasoning and inferring to ultimately transform data into knowledge. Resource Description Framework (RDF) [165] and Web Ontology Language (OWL) [163] are two very

different but complementary ways to perform such tasks and query languages such as SPARQL [192] can be used to query such data from different sources all around the internet. Semantic interoperability is a very big field with numerous standards and techniques that are beyond the scope of this chapter so we would spare the details for the sake of this review.

4.3 Some of the Applications of IoT in Healthcare

Machine learning and deep learning is being used in many different parts of health care such as the discovery of new drugs, manufacturing them, aiding doctors in surgeries, performing radiation treatments, and much more. But, when it comes to IoT, the primary use cases of them would be in developing remote monitoring, prediction and recommendation systems, and living assistants. The borders between these systems are not as clear-cut as you might imagine, since almost all the papers that will be discussed in this section can fall under two or more of the categories. Hence, to avoid redundancy, we abstain from categorizing them.

First and the foremost is ubiquitous monitoring using sensors embedded in wearables and implants because without them there would be no data to analyze. [126], [253] all propose different telemonitoring systems with varying configurations but with the same purpose. Their systems aim to monitor and investigate the health of patients whether ordinary, critically ill, elderly, or recently discharged from hospital by tracking parameters such as heart rate, blood pressure, body temperature, ECG, EEG, EKG, and other physiological attributes and sending them to central servers using various wireless technologies like wifi, Zigbee-based WBAN, WSN, and BSN, altogether enabling the doctors to monitor the health of the patients from the convenience of their rooms.

In a more unique use case, authors in [83] propose a system to track the location and monitor the health and condition of war soldiers on the battlefield that in return makes the search and rescue operation faster and more accurate in the event that an individual gets injured. Aside from the aforementioned physiological parameters, they also use a humidity sensor, vibration sensor, bomb detector and then use a hybrid of ZigBee and LoRaWAN network infrastructures to transfer the data to the control room either after some fixed interval or only when there is a significant change in conditions of a soldier. In [196] authors also present ways to track the health of soldiers as well as the ammunition on them.

All the collected data would be then get merged with possible demographic information and historic sensory data and get thoroughly analyzed using different AI techniques to predict if everything is on the right track and the patient is healthy or whether something is wrong and require immediate attention. For instance in the case of soldier data, in [83], the authors apply K-Means classification to the data and classify individuals into healthy, ill, abnormal, and dead. [277] uses ECG and other sensory data for earlier prediction of heart-related diseases. In [274] authors utilize different fuzzy rules and techniques such as Probabilistic Fuzzy Random Forest (FRF) to classify and efficiently predict various diseases. The latter also shows that FRF models perform better than Linear Regression and Q- Learning algorithms in their study. Hybrid and more general frameworks have been also proposed in [252] to detect and monitor several diseases such as diabetes and cardiovascular-related ones and forecast their severity at the same time. In [183] authors proposed to use IoT frameworks to identify transmissible diseases at their early stages and prevent them from turning into a crisis.

Living Assistants are without a doubt one of the main use cases of IoT and are assisting millions of patients, especially the elderly population on a daily basis by providing personalized analysis for disease prevention and collaborative care, all thanks to the advancements of wearables, implants, AI models, and communication networks. We have covered most of the services and functionalities that wearables and implants can provide in earlier sections, but aside from them, authors in [127], [143], propose frameworks and tools such as home-based wireless medical boxes to track and monitor medicine consumption of patients, help them in categorizing of drugs, keep logs of intake history, and warn them if the pill is not taken on time. Of course, this can also be synced with vital sensors of the patient wearables and if a significant change occurs due to drug consumption, the corresponding doctor can be notified immediately. It goes without saying that this would be a huge help for our elderly citizens, especially the analphabets.

Also, most medications must be maintained at a constant condition and temperature. [20] suggests that millions of dollars get wasted annually due to random and unexpected refrigerator failures. So, the authors in [20] propose IoT techniques to constantly monitor the condition of drugs and prevent costly errors.

4.4 Machine learning Challenges in IoT Healthcare

In this section, we mainly list and discuss the major limitations and challenges of machine learning in IoT healthcare. Figure 4.3 presents a tight relationship among IoT, machine learning (ML) and personal healthcare(PH). There are many studies that state the applications of machine learning in IoT and PH [3]. The figure shows that IoT generates data that feeds ML algorithms and then the outputs provide solutions for PH such as disease diagnosis, patient behaviors analysis, and assistive care advice.

ML and IoT-based assistive PH services have already had and will have so many impacts on peoples' lives due to technological advancement. However, assistive PH will require to face challenging issues such as usability and affordability [220]. Additionally, privacy and authentication problems in IoT devices can attract hackers' attention and cause issues as they will be hacked if not correctly secured [220].

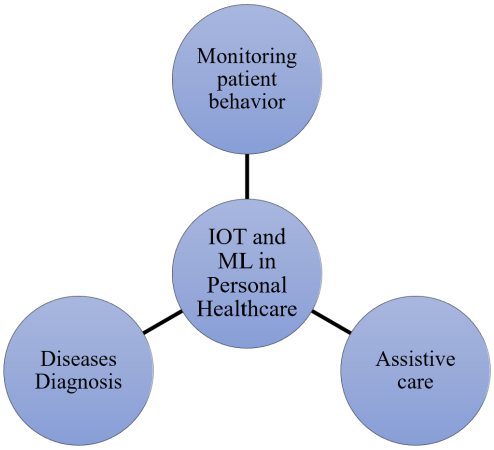


Figure 4.3: A general schema of IoT and Machine Learning applications in Personal Healthcare (PH)

Further discovery shows that using ML-based PH service enables us to use a predictive analysis approach that assists released patients from a hospital who may need to readmit to the hospital. The goal of predictive analysis is to create a risk classification model in which particular patients with higher risk are controlled with additional effort and assistive care such as providing additional monitoring IoT devices/ sensors and constant (real-time) follow-up and analysis. These models are highly generated based on past historical experience and data. The dynamic PH system which would assist in re-admission avoidance

initiatives must also leverage dynamic data from the patient and take it into consideration to predict future possibilities and initiate an action plan to mitigate probable complications [3].

We present an abstract view of challenges and solutions in figure 4.4 which shows two challenges: 1) outdated dataset which leads us to incorrect decisions, and 2) Data security and data privacy which lower the reliability of IoT devices. On the other hand, the figure provides us two associated solutions: 1) Online learning to keep learning for the new entry data and 2) Federated learning for learning from distributed data among end-users. In the current section, we explain the first challenge and the second one by elaborating on them when they may occur. In the next section, we elaborate on the corresponding solutions for giving two solutions. However, the solutions also have their own pros and cons.

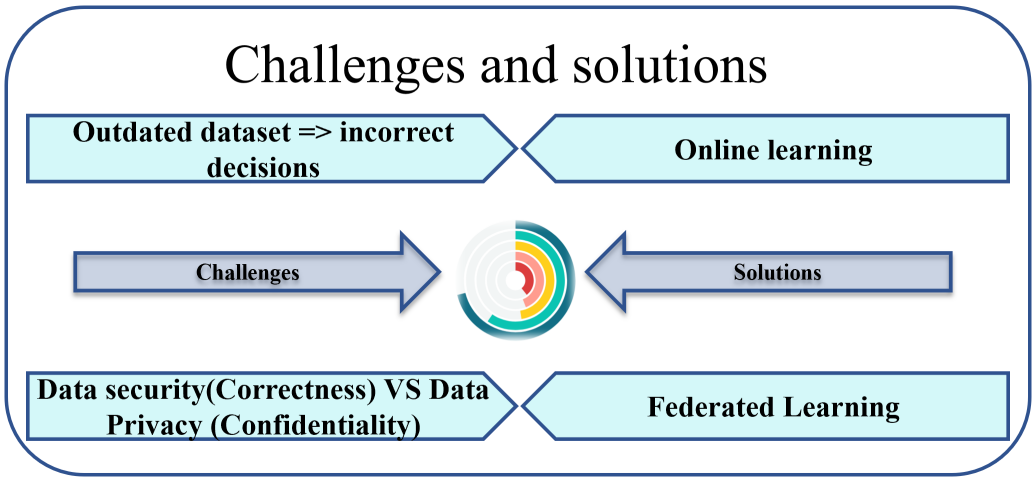


Figure 4.4: A general ML challenges in IoT healthcare and associated solutions

4.4.1 Challenge #1 : Outdated dataset

There have been many research studies working on applications of machine learning (ML) in IoT healthcare. The ML algorithms develop analytic models integrated into diverse healthcare service applications and clinical smart systems [219]. These models mainly get evaluated on the collected data from IoT devices to recognize behavioral patterns and different clinical conditions of the patients such as identifying the patient's improvements, habits and their anomaly actions in daily routine activities, different behavior of sleeping, digestive, drinking, and eating pattern. Having those patterns available, the smart decision-making systems recommend particular lifestyle advice, care plans, and special treatment for the patients. Additionally, the doctors can further be engaged in the care plan process to evaluate and validate the

lifestyle advice and care plans. Medical data including clinical, lifestyle, behavior are so sensitive and it is highly likely that there may be different types of biased engaged in the process of data collection and may not be diverse enough to govern all scenarios. Furthermore, the noisy, incomplete data could lead to a lower probability rate to detect and predict a health-related diagnosis and advisory notice.

The training dataset and generated model could have an outdated version of the dataset and even if we had a rich model, it will not be effective anymore. Having old and outdated models and datasets, lead us to an incorrect decision derived from the smart systems [3].

ML challenges in Assistive Care

We believe that ML algorithms are highly related to statistical analogy and deduction in which the ML algorithms make decisions and predict using existing and previous experience (training dataset). In the case of monitoring a patient, the ML-based method will monitor and analyze the situation according to the training dataset. Thus, the training dataset plays an imperative role in seeking the current pattern and predicting the future trend of a given new problem during a test phase. This dataset sometimes is biased and would not be as diverse as possible to cover many scenarios. For example, let's consider a sleep monitoring case, sleep patterns can vary from person to person, from kid to elderly and health status. Therefore, a complete dataset of all case studies are not existing during training time to keep a track of sleep patterns, and this may lead us to an incorrect estimation in PH [219].

Furthermore, using IoT and ML enable PH to make a decision for diagnosis, prediction. There are some examples, in which ML-based decisions could not be correct, and it is not possible to state why a certain decision was made. For instance, in the case of autonomous cars, a few accidents occurred due to wrong decisions made by the cars [219].

The main point here is how to assess the decision taken by an AI machine when unsupervised machine learning algorithms were used. This may lead us to an ethical question of who would be responsible in case of a false statement and how to recognize or rectify that incompleteness in the process of decision-making and get to know how would unsupervised machine learning algorithms work. These challenges would limit the usage of ML algorithms in using PH for a sensitive application, particularly personalized medicine care [219].

4.4.2 Challenge #2: Confidentiality & Correctness

The invention of medical Internet of Things (IoT) devices and the high popularity of ubiquitous wearable devices enables us to have non-stop personal healthcare monitoring everywhere such as home, work, and hospital environments for different applications like childcare and assisted living. These devices record real-time electronic health measurements from a variety of objects and sensors and transfer these patient data to an application server to be pre-analyzed and pre-processed to restore on a data server [121].

These processing and analysis use machine learning algorithms to provide different services such as motion tracking, presenting the number of steps, burned calories, sleep monitoring, traveled distance, and essential signs measurements like heart rate, electrocardiogram (ECG), skin temperature, and electroencephalogram (EEG) [94]. So, analyzing the data generated by IoT devices and sharing them through a connected network to a server raise uncertainty, trust, and confidentiality issues. The data stored on a server and the communication networks are vulnerable to breach. So, data security and privacy are important challenges to take into consideration. There are basic solutions provided to increase data security by applying encryption algorithms and keeping the data safe however, if a hacker gets to know the key to the decryption algorithm and uncovers the message, then the confidential information will be everywhere. Furthermore, while encryption, there is a possibility of losing some information and if the decryption algorithms fail to retrieve all original data, the process of encryption and decryption is not useful anymore.

ML Challenges in Monitoring Patient Activities

In a certain case of monitoring patient activities, several research studies take advantage of data generated from motion sensors to define physical activity in real-world settings for a variety of case studies. Research studies have proved that motion sensors type is one of the reliable tools to assess long-term human physical activity for particular cancer patients during the time they are in their therapy sessions [91].

However, due to IoT devices features, gathering data from medical IoT sensors/devices for healthcare application are very sensitive. Recent advances in web technologies and wireless communication/transformation enhance the gathering data process and the remote real-time monitoring [121]. But, the complicated workflow of gathering medical data increases the security issues and privacy risks during the life-cycle of the data gathering.

In the activity recognition process using mobile devices rather than wearable devices, the challenging issue relies on determining the data that can preserve the privacy of users while it is relevant and important for machine learning tasks [121]. To address this challenging issue, we may face and answer two questions: first, does the gathered data has high protection layers or encryption protocol so that no one is granted access to it? Second, how to assess if the protected data is maintained as accurately as captured? Obtaining a trade-off between data computation and data privacy is an essential objective to have a secure transfer of data and protect data using mobile devices and enhance the end-users trust.

One of the main challenges of using IoT for healthcare monitoring is the security, particularly privacy of patient information in the machine learning analysis process. An efficient user authentication framework ensures that only legitimate account users have access to data and services. So, the problem is the data accessibility for users that is vulnerable for hackers to access sensitive. Sharing information from IoT devices as it is will be problematic [121].

4.5 Alternative Promises in IoT Healthcare

The IoT usage statistics have shown the potential application to many medical and healthcare domains with access to large volumes of corresponding data gathered by IoT sensors/devices. However, the increasing need for high healthcare data security and data privacy forces IoT devices to be known as a disconnected island of data [280]. Furthermore, due to lack of updated data model issues, data privacy, and security which are the main challenges using traditional machine learning algorithms mentioned in section 4.4.1 and 4.4.2, we investigate advanced machine learning promises with IoT-healthcare applications in this section. We elaborate on the promises in the following sections. First, we discuss the online learning algorithm which addresses the lack of updated data and provides a solution in which the classifier can learn from upcoming new data and update the model in real-time on each learning iteration. Second, we present a federated learning technique that enables us to learn from distributed data collected from end-users without transferring the collected data to an application server.

4.5.1 Online Machine Learning or Adaptive Learning

A basic procedure of a machine learning algorithm is already discussed and shown in section 4.2. Additionally, we discuss in section 4.4.1 what are the challenges of traditional machine learning algorithms. Thus, in this section, we aim to propose a solution that enables scientists to address the challenge properly by using an online learning algorithm.

In machine learning, there is a fundamental algorithm called online learning or adaptive learning in which we feed data in a sequential order which is highly recommended and used to update the best classifier for future data even if the training process is done. In comparison to batch learning algorithms which generate the best classifier by learning only from the training dataset and never get updated afterward. In other words, online or adaptive learning consider tasks (during the training process and after that) arriving in a stream rather than an offline finite dataset. But, the tasks are associated with the ability to efficiently adapt to the current task in the stream with respect to the corresponding learning rate, more than remembering the old tasks [95].

Furthermore, online learning is a new technique in machine learning where it is physically and logically infeasible to train over the training data such as patient data over therapy sessions, requiring the need for data will be generated in the future. It is also used in conditions when it is essential for the algorithm to train whole data due to lack of memory [68].

General Process of Online Machine Learning

In this section, we aim to provide a very brief overview of ab online machine learning algorithm. Figure 4.5 presents a general schema of the online machine learning algorithm. In general, the online machine learning schema works with three main packages: 1) Input package including wearable or healthcare IoT devices, 2) Online machine learning package involves of 5 steps, 3) The results and predictions projected on AI devices. The main online machine learning steps are the same traditional machine learning process, except one more step which is re-training the learning model with a learning rate that denotes the importance of new input data. The larger the learning rate, the more important input data is considered. However, if a larger learning rate is chosen, this model will be highly likely exposed to outliers and may end up stuck in local minimum or maximum. On the other hand, the smaller the learning rate, the less

exposed to outliers. However, it takes a long time to get the model updated based on new entry data due to the smaller learning rate.

The five steps in the online machine learning package presented in figure 4.5 play an important role in the process of applying online learning in IoT healthcare. Within these five steps, steps 1 to 3 are the typical learning process in traditional machine learning including reading input data (training data, training a model, and applying the model). Online learning algorithm starts with step 4 where the algorithm accepts a new sequence of data and in step 5 the algorithm learns from the newly added data and updates the model with a learning rate customized value.

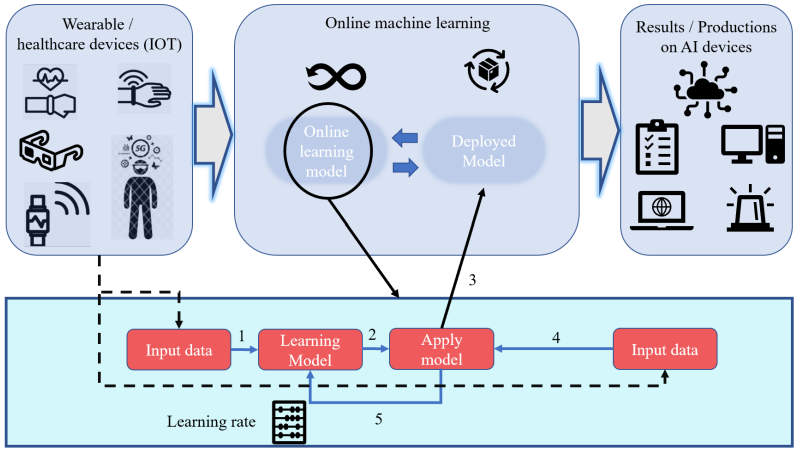


Figure 4.5: A general schema of Online Machine Learning

Online Learning Applications in IoT Healthcare

With advances in the world of technologies such as artificial intelligence, particularly in machine learning and IoT devices, the AI applications and IoT objects usage have proliferated in many domains, more specifically and demanding in medicine and healthcare. Adaptive learning, also known as never-ending learning [168] or online machine learning [68], is a new basic idea in machine learning in which models learn from data continuously and evolve based on the new input of data with a certain learning rate while maintaining previously learned knowledge. This non-static process of supervised machine learning algorithms allows the model to iteratively learn from data and automatically update its behavior with the learning rate. The smaller the learning rate, the smaller range of behavioral we will have. In other words, we do not add more weight to the upcoming input data. However, we aim to have the model updated

sooner based on the new data, we may need to initialize a pretty large value for the learning rate which will not get affected by outliers. One of the examples of this online learning are recommender systems utilized by companies such as Netflix and Amazon, however, such systems have their own challenging issues like fairness in the machine learning process [4].

Big Data streaming:

A big data streaming computation has been highly applied as a fundamental role in real-time healthcare analytics. There are applications of online machine learning such as real-time monitoring and tracking systems which play an essential role in the healthcare and medical domain. Mobile applications and sensors and wearable medical devices are the examples of popular rich sources that have been generating constantly a high volume of data namely, streaming data [95]. Apparently and logically, using traditional machine learning algorithms to apply on the streaming data to make real-time actions in case of emergencies looks a difficult task. Therefore, a solution is required for real-time big data processing to make sure the results are effective and explainable. For example, a real-time solution for monitoring flu and cancer patients is proposed by applying Twitter mining tool [139]. Another big data model for real-time medical analysis is investigated in [5]. The model used Spark streaming and Apache Kafka to get evaluated on a stream of healthcare data. In [45] a real-time health status prediction solution was proposed by applying Apache Spark, which is a powerful big data analysis tool. Spark was used successfully for streaming data due to traditional machine learning limitations and handling distributed computations.

The second important extension of online machine learning is online meta-learning. Finn *et. al.* [66] introduced an online meta-learning approach based on the regret-based meta-learner. The approach performs MAML-style, which is a model agnostic meta-learning for adaptive deep learning, meta-training online during a task sequence. In this approach, there are meta-training and meta-testing phases in which we learn from input data in both phases training and testing phases. Even while testing, the meta-testing phase takes advantage of a training phase too. This ability make MAML looks powerful enough to yield a better result than other deep learning methods. Furthermore, MAML is one of the approaches that work best for few-shot learning (FSL) and one-shot learning (OSL) that enables classifiers to learn from a few samples of each category to predict unseen class labels with high accuracy [173].

Further Challenges of Online Learning Although online machine learning looks ideal for IoT healthcare applications, the challenging issue left in applying them with high accuracy [173]. One of the main challenging issues is catastrophic forgetting, where the new information prevents from learning what the model has already trained [138]. This obstacle leads us to a significant failure in the classifier's performance while the new input data is being integrated or, regenerates a new model rather than keeping previous knowledge [162]. Most of the applications for online learning in non-medical domains are less influenced by this limitation [195]. Online learning models in health care address a number of different challenges where it is needed multiple complex tasks.

A simple solution to address the catastrophic interference problem is to redo the training phase completely to regenerate the model every time new data are available, however, this process is computationally expensive and prevents from having real-time inferences [138].

4.5.2 Federated learning

Having heterogeneous IoT devices and associated different end-users ' (patients) information on them make it highly likely vulnerable for hacking while transferring information to an application server or when the data is restored on the server which raises data privacy and security issues. In this section, we aim to address these issues and investigate a solution that helps us solve them at an acceptable level. This solution must be able to train from isolated devices and integrate a model in a way that preserves users' information.

In a traditional machine learning-based approach, data gathered by IoT devices are uploaded to an application/data server and then trained models leveraged by machine learning algorithms. However, data owners (devices) have proliferated and data privacy gets important [147], particularly in the medicine and healthcare field. In order to address the privacy requirement regarding individually identifiable, here, we propose to use federated learning, which was firstly proposed by Google [131], which is a new approach to address this data dilemma, which is denoted as a challenging issue of training a high-quality shared global model with a centralized server with decentralized data distributed among a large number of devices or end-users [269].

We propose to take advantage of federated learning (FL) which is a machine learning algorithm where the goal is to train a centralized rich model while training data remains de-centralized over a large number

of users and devices where the network connections are unreliable and non-fast [147]. We use the federated learning algorithm for this situation where IoT devices independently perform an update to an input model (received from a central server) based on their local gathered data on demand and transfer this updated model to the central server, where the users-side updates are aggregated to generate/update a new global or generalized model. The common devices in this setting are mobile phones, in which data transferring and local sharing efficiency play an essential role in this setting.

General Process of Federated Learning

Federated learning is an approach to generate models, unlike other machine learning algorithms that need to have all data available as a central training dataset. In federated learning, a model is trained iteratively by aggregating collective models gathered from among multiple sources (devices: users information extracted from their cell phone). The users keep preserving their privacy and their local (training) dataset to themselves, but still may be able to participate in a shared federated learning process. Then, an aggregator would broadcast again the aggregated model to all the users and after that integrates again the updated model learning from the local training data in the devices [147].

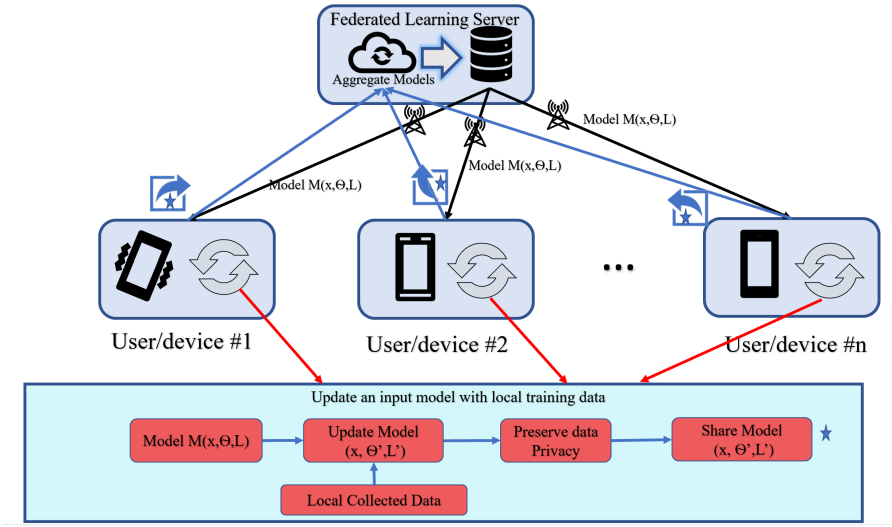


Figure 4.6: A general schema of Federated Learning

We present a very abstract overview of the federated learning process in figure 4.6. This figure shows that a federated learning server communicates with all end-users / devices and gets them to check in the

server and then start broadcasting the current model. The end-users / devices receive the model M with input data (x), parameters θ and loss function . Next, the device starts to update the model by learning from the local training dataset which is collected. The updated model M with updated values of parameters θ' and loss function ', per each device will send back the updated model. Finally, the server aggregates the received models from devices and then stores them in the server and keeps repeating this whole process. Having this process, enable specialists to monitor their patients accurately with updated information almost real-time. Furthermore, this FL prevents from sharing sensitive information. However, federated learning uses only cellphones as smart devices which is currently is one of the logistic limitations.

Federated Learning Applications in IoT-healthcare

Federated learning enhances the collaborative training of models such that the sharing raw data is nonesense. We need a federated learning system capable of preventing inference over both the messages exchanged during training and the final trained model while ensuring the resulting model also has acceptable predictive accuracy [246]. In [265] researchers proposed an FL-based solution for learning a machine learning model namely, PerFit. PerFit works with a cloud-based architecture that provides computing power for IoT devices. In [112] the importance of the computation power is discussed in detail. Having the architecture well-structured provides a situation for IoT devices to unload their computing tasks due to efficiency and low latency requirements. As FL is communicating through different devices, servers, and the cloud, so models can be shared locally by preventing from compromising sensitive data. The PerFit framework's learning process works mainly with three steps: 1) unloading tasks phase, 2) The learning phase, and 3) Personalizing phase. Researchers in [265] evaluate PerFit's efficiency on a data-set called Mobile-Act, which centers on human activity recognition, which has ten different activities such as walking, jumping, and jogging.

In [39], a new FL-based framework, FedHealth, using transfer learning was proposed for IoT healthcare applications. The transfer learning technique has been used in FedHealth to decrease the distribution divergence among a wide variety of fields. Furthermore, FedHealth uses a certain encryption algorithm to enhance the security of communicating current model updates between the aggregation server and end-users, and also vice versa. FedAvg has also been utilized as a federated optimization algorithm in the proposed FedHealth. There is a certain chance that the global model updated in FL centralized server

is available to all end-users, but the personalized model in each user does not guarantee that performs well locally [123]. In order to handle this, to enhance the performance of the federated learning model for end-users, the researchers in [39] leveraged transfer learning to train a personalized model for each user.

In [58], researchers proposed a deep-learning based FL framework for decentralized healthcare applications that preserves data privacy and enhances data security in a decentralized architecture. They also proposed a scheme in which an automated training data acquiring process is applied. Additionally, they applied and evaluated the algorithm on skin diseases and leveraged transfer learning techniques to address the problem of lack of healthcare data existence in generating deep learning models.

Further Challenges of Federated Learning Although FL provides high data security levels and preserves data privacy, some posted attacks determined that simply sharing only local data during the training process, updated the model M does not guarantee sufficient data privacy [246]. Researchers in [246] proposed a hybrid approach namely, Privacy-Preserving Federated Learning, to yield a model with acceptable predictive accuracy and also is capable of preventing inference over both the messages communicating during the training process locally and the final trained model.

Furthermore, FL on IoT devices provides the following challenges:

Heterogeneity of devices: IoT devices used for medical and healthcare issues and the ones used for general purposes are all different in terms of technologies and hardware such as the version of CPU, memory storage, and network connections bandwidth, storage capacity, and last but not least power. This could add more cost using FL, as numerous factors are required to get configured as fault tolerance. Furthermore, some devices may drop out of different learning processes due to various reasons like bad network connectivity and energy constraints.

Statistical Heterogeneity and Heterogeneity of models : In all machine learning, particularly FL, from collecting data point of view users' distributions of activity and settings play a fundamental role. Thus, different scenarios and settings due to very different physical features and behavior lead us to diverse data samples among devices in the FL that may cause an obstacle for making a rich model. Speaking of this, we end up having different diverse models in the server application to get them aggregate before broadcasting again.

4.6 Conclusion and Final Remarks

The Internet of Things (IoT) is getting stronger and more powerful with the implementation of machine learning algorithms for network monitoring and user activity management. However, traditional machine learning algorithms may fail while applying decentralized data collected from IoT devices, since the nature of these algorithms is to get entire training data at once and generate a rich model which is capable of predicting unseen class labels. In this chapter, we discussed IoT, its pipeline, and its application in healthcare, and address the challenging issues ML algorithms may face. Finally, we investigated the process of data collection and its issues, the impact of Big Data in IoT health, general IoT challenges, particularly the challenges of machine learning in IoT healthcare, and associated novel solutions to address them. All in all, we can see that there exists a strong relationship between IoT, Embodied AI, and eHealth. Understanding different aspects of each of the core fields will enable us to employ this knowledge and develop a hybrid model that can aid doctors and patients in the healthcare industry. In the next chapter we also explore different machine learning and deep learning algorithms that can be employed to analyze health data.

Chapter 5

Applications of Deep Learning in Health Informatics[1]

[1]F Shenavarmasouleh, FG Mohammadi, K Rasheed, and HR Arabnia. Accepted by 2022 International Conference on Health Informatics and Medical Systems (HIMS).

Reprinted here with permission of the publisher.

Healthcare industry generates almost a third of all the data on the planet annually [206]. Every second, a massive amount of data is generated from all types of devices, imaging tools, patient management portals located within the hospitals and clinics, or from all the emerging Internet of Things (IoT) sensors and implants. This huge load of heterogeneous data that contains highly redundant and correlated info is called Big Data [209]. Extracting valuable knowledge from big data is inherently a difficult task. But, analyzing healthcare data enables us to take modern diagnosis and treatment to an entirely new level. It can aid doctors and clinicians in analyzing radiology images with more accuracy and direct their attention to the most important locations of the picture and thus help them find the nodules, masses, and other deficiencies that otherwise would have been missed. Aside from this, they can work as stand-alone classifiers and/or clustering tools to diagnose various diseases and even predict the malignity of them. Additionally, they can be utilized in creating novel personalized medicines and help researchers in genomics.

Over the past few decades, Machine Learning proved itself as a valuable analysis technique and researchers have benefited from all the different powerful methods that come with it, such as support vector machine (SVM), tensor decomposition, random forests, Bayesian networks, and much more, to extract useful features and discover hidden patterns within the data. However, for the most part, using these methods required engineering custom features by experts with extensive knowledge about the domain of the task at hand and this led to the emergence of a new branch of machine learning, so-called deep learning that occupied researchers with much more sophisticated tools and methods that overcame the previous shortcomings. A Deep Learning architecture can be viewed as an Artificial Neural Network (ANN) with two or more hidden layers and it is capable of extracting high-level features from data automatically and using that to perform the task, thus removing the need for expensive and time-consuming feature engineering phase while yielding a better accuracy.

With the advancement of deep learning, it is getting employed in more and more fields by researchers and it is now a must-have in most of the interdisciplinary studies. Healthcare and bioinformatics are not exceptions. A wide range of deep learning architectures has been used to analyze the massive data in healthcare throughout the years. In this article, we aim to provide a review of deep learning methods, their diverse applications for bioinformatics and healthcare research categorized by their most prominent architecture, along with the challenges that come with them. We believe that this article can serve as a starting

point for researchers and provide valuable insight for future studies in deep learning in bioinformatics research.

5.1 Classification and Segmentation Models

5.1.1 Convolutional Neural Networks

The ultimate objective of computer vision is to mimic the behavior and operation of human eyes. Convolutional Neural Network (CNN) is a machine learning algorithm developed based on biological visual cortex processing and it aids computers in analyzing pictures and movies and, as a result, understanding the objects contained in them. CNNs have become the go-to technique for any work that requires dealing with media after AlexNet [133] won the ImageNet image classification competition in 2012.

CNN is a supervised deep learning architecture that employs three types of layers namely, convolutional layers, pooling layers, and fully connected layers to solve all sorts of different classes of tasks. The top five categories are Image Classification, Object Localization, Object Detection, Semantic Segmentation, and Instance Segmentation, listed in ascending order of complexity.

In an Image Classification task, usually, a single core object is present in the image, and the aim is to determine which category that image belongs to. Object Localization is a little more complex. The model's aim in object localization is to output the position of those items as bounding boxes, a rectangular box surrounding the object, in addition to predicting the category to which they all belong to.

Object detection comes next. CNNs are capable of detecting edges and, as a result, determining object boundaries. As a consequence of this characteristic, they can be used to detect a variety of objects in a given image. However, doing so necessitates applying them to a large bunch of areas with varying scales on each image, which takes a long time. As a result, a significant amount of study has been conducted to address this problem.

It all started with R-CNN (Region-based CNN) in [81] where the authors used a region proposal module to tackle the aforementioned problem. Following this publication, the same authors went deeper and improved the model by addressing the model's overlapping nature in previous work, eliminating unnecessary calculations, and allowing the framework to train all three concurrent models at the same

time. This model was given the name Fast R-CNN [80]. Finally, Faster R-CNN [200] was proposed to fix the bottleneck that existed within the Fast R-CNN by employing Region Proposal Network (RPN), altogether making Faster R-CNN the go-to model for most of the use cases regarding object detection.

Object detection, as great as it is, is still unable to comprehend and supply us with the actual shape of the objects, and instead just provides the bounding boxes. This is where Image Segmentation shines since it solves the problem by producing a pixel-by-pixel mask for each object. The task of image segmentation can be classified into two broad categories. Every pixel in the picture must be assigned to a predetermined class in Semantic Segmentation. Furthermore, all pixels relating to a class are treated the same and are given the same color, thus differences between various object instances belonging to the same class are ignored. Instance Segmentation, on the other hand, treats each instance of a certain class as a separate entity with its own color and label.

In the case of Semantic Segmentation, deep learning solutions started with [152] where the authors presented a fully convolutional end-to-end trainable network. The big picture for architecture was for the model to have a well-known classification model such as AlexNet as encoder and then use transpose convolutional layers with pixel-wise cross-entropy loss as the decoder to upsample the result into the same size as the original image. However, since the encoder reduces the resolution of the image, the decoder failed to produce accurate masks. Thus, to counter this, the authors decided to add skip connections from earlier layers adding their values to later layers to offer required information for the decoder to properly create the masks more accurately. This approach has shown to be highly effective. Following the success of this paper, the authors of [203] presented the U-Net architecture which consists of a contracting path to capture context and a symmetric expanding path for accurate localization. U-Net gained a lot of popularity, particularly in the medical field in which we will talk more about in detail in later sections. The idea of skip connections attracted a lot of attention and caused researchers to further study and analyze them that led to the creation of architectures such as DenseNet [117], SegNet [17], and ResNet [98].

The Mask R-CNN model [97] was proposed to help image instance segmentation. It has built on Faster R-CNN by specializing in producing pixel-by-pixel masks for each object in addition to identifying the bounding boxes and class labels. It ingeniously adds another Fully Convolutional Network (FCN) on top of RPN, technically adding a new parallel branch to the Faster R-CNN model architecture that generates a binary mask for the object discovered in a given region. It's also worth mentioning that the

authors had to make minor changes to the model to address a problem with location misalignment caused by its quantization behavior. Authors in [222], [223], [225] utilize Mask R-CNN to localize, mask, and detect the type of lesions that appear in the eyes of diabetic patients.

CNNs are primarily designed for fixed-size 2D images; however, most of the tasks in medical profession use MRIs and CTs which are inherently 3D or 4D with varying sizes, and with objects being relatively very small and being positioned in arbitrary locations. The naive solution is to use 3D images themselves as they are. Convolutional layers have been reconfigured and extended to 3D kernels to create 3D convolutional networks and they proved to surpass other approaches [161]. However, processing 3D images using 3D convolutional networks is computationally intensive.

There has been an extensive amount of research to enhance this architecture. In [43], authors expanded U-Net from 2D to 3D architecture. In another similar work [166], V-Net, a 3D version of U-Net architecture was presented. And authors in [284] used 3D Faster R-CNN to segment nodules. The other two commonly used techniques were multi-stream learning and 2.5D models.

Multi-stream aims to look at the data with varying perspectives, angles, directions, scales, resolutions, and even a combination of different types of data over the same period of time. Most of the time, the proposed architecture train one channel for each of the variations of the data and then concatenate them all into one main channel before doing the image analysis task such as classification or segmentation. Multi-scale analysis, for instance, comes from the intuition that blurring pictures with Gaussian Blur vanishes details that are smaller than a certain resolution. The power of multi-scale image analysis comes from the ability to change the sharpness of the image dynamically and hence look at details with different levels of resolution [93]. For example in [21] authors analyze and segment tumors in brain images using multi-scale approaches. [154] employs multi-scale features to create prognostic classifiers for predicting treatment response and patient outcome. In [211] a multi-scale segmentation model, namely MANA was presented for nuclei detection. Authors in [122] propose a novel multi-resolution CNN and use it to detect skin lesions. And, [279] utilize dilated convolution to perform multi-scale semantic image segmentation. In multi-modality approaches, researchers make use of different types of imaging and screening techniques all at the same time. Doing so provides two main advantages. First, this prevents the model from overfitting, and second, it helps different streams of data collected over the same period of data complement each other and cover the shortcomings of the others in the event of one not being

able to catch a particular kind of detail about the data [14]. Similarly in 2.5D architectures, the researchers try to mainly convert $k \times k \times k$ media into k orthogonal $k \times k$ 2D channels with possible extra ones for additional information. [268] proposes 2.5D architecture for semantic segmentation and [272] use 2.5D models for analyzing brain images and segmenting stroke lesions.

5.1.2 Autoencoders

Autoencoder is a type of unsupervised deep learning model architecture that aims to encode data efficiently and use the encoding to reconstruct the original input with minimum loss. It accomplishes this by mapping input to itself through an interconnected neural network. Feature reduction, extraction of latent feature representations, and inferring missing data are some of the applications that autoencoders are inherently good at given their architecture characteristic.

There exist several variations on the base model each tailored for a specific task. Denoising and stacked denoising autoencoders, as their name suggests, are perfect for removing noise from the input data and can be used as a preprocessing step in all sorts of medical tasks. Sparse autoencoders add a sparsity penalty to the loss function and even though they can have more hidden units than inputs, only a small group of the hidden units are allowed to be active at the same time forcing the model to compress related unique features to each other. Sparsity improves performance on classification tasks. Thus, they have been utilized in various classification tasks such as breast cancer nuclei detection [270], and diagnosis of Alzheimer's [26] and Parkinson's disease [145]. Sparse autoencoders can be stacked over each other and form stacked autoencoders which are more powerful in capturing more complex features [184]. Also, any kind of autoencoder with more than one hidden layer is called a deep autoencoder, again giving the model a better ability to understand inter-related characteristics among the input features. As an example, authors in [158] use deep autoencoders to classify nodules in lung images and researchers in [233] also employ them to detect different types of cells in bone marrow biopsy images.

Unlike traditional autoencoders that map the data to a single value in latent space, variational autoencoders tend to map input features into a probability distribution for each latent attribute. Given this characteristics of them, they can easily act as generative models as well by providing the possibility to interact with latent space probabilities. Generative models such as generative adversarial networks (GAN) [84] and variational autoencoders can be used for image super-resolution where a bigger or more

enhanced image is needed such as in endomicroscopy [199], single molecules images [237], and in cases where low-dose CT (LDCT) scans have also been considered instead of normal-dose CT (NDCT) to reduce potential health risks [153].

5.1.3 Deep Belief Networks

Deep Belief Networks (DBN) are a multi-layer neural network with both directed and undirected connections. Although at the first glance, it appears to have a basic neural network design, the training procedure is vastly different between the two. Deep Belief Networks are a stack of Restricted Boltzman Machines (RBMs) that have been trained greedily layer by layer and fine-tuned using the up-down algorithm to learn high-dimensional manifolds of data as opposed to end-to-end training of neural networks using backpropagation. In neural networks, the layers are built on each other and each layer uses previous embeddings to create a more abstract meaning out of the data and learn higher-level features. In DBNs, however, each layer learns and encodes the entire input. Given the characteristics of DBNs, they can be incorporated in either supervised tasks as stand-alone classifiers or be used in unsupervised manners like for instance the way they are being employed in autoencoders or as generative models. DBN was employed in [193] as a classifier to differentiate between healthy and schizophreniac patients, and it was shown that it outperforms a famous machine learning technique known as support vector machine (SVM). In [125], authors utilize DBN in an unsupervised manner to create a framework for analyzing radiology images. And in [144], authors use unsupervised DBN to extract more useful latent space features for fMRI images.

5.1.4 Recurrent Neural Network

Recurrent Neural Network (RNN) is used for extracting patterns from sequential data and bringing in temporal features to the learning process. They can be utilized in all tasks that can benefit from this characteristic such as text processing and video processing. Since videos can be viewed as a sequence of images, in the context of biomedical imaging these models can be useful in any task that is related to the time such as analyzing the progression of diseases. RNN is the base model and in the past few years two more complex models, namely Long Short Term Memory (LSTM) [105] and Gated Recurrent Units (GRUs) [41], have been emerged to address its shortcomings such as the vanishing gradient problem and

by doing so extended the ability of sequential models and gave them the ability to keep track of older data more successfully. Later, their performance increased even more by incorporating attention techniques into recurrent models [251] as well.

In [159], authors employ LSTM to create a Spatio-temporal deep learning method that uses resting state functional magnetic resonance imaging (rs-fMRI) to diagnose Attention Deficit/Hyperactivity Disorder (ADHD). Authors in [74] utilize same approach on fMRI images for autism disease classification. Spider U-Net [140] also utilizes LSTM to segment blood vessels in 3D using computed tomography and magnetic resonance angiography (MRA) images. In [271], authors use GRU to predict lung cancer Treatment response in patients. [71] also proposed a deep learning method for 4D segmentation of longitudinal MRI while considering the brain maturation in infant brain imaging.

5.1.5 Reinforcement Learning

Reinforcement learning (RL) is a machine learning approach that rewards desirable actions while penalizing undesirable ones. Doing this will force the agent to learn the best behavior, also known as policy, which leads to the most cumulative reward by utilizing Markov decision process (MDP) mathematical formulation. RL agents can constantly communicate with their environment via sensors and possible actuators and decide on their next action based on their current state. After enough trial and error, we end up with agents that know what they want in a given environment and can simply act on it to maximize their gain.

Deep reinforcement learning extends RL and enables it to deal with higher-dimensional problems by bringing deep learning into the equation. This way, data types such as images that have a relatively large number of input size can benefit from be RL algorithms as they can be fed into deep RL models without worrying about having to manually engineer the state space of the problem and avoid the curse of dimensionality [177].

RGB pixels of images cannot be used in RL to make decisions by themselves. Instead, CNNs, mostly 2D and 3D, are utilized as the primary feature extractors in the tasks that require working with media. These features can then be used to aid the model to decide on the optimal next move that enforces achieving the most expected return. In [78] authors propose Marginal Space Deep Learning (MSDL) framework that uses RL to perform tasks of object localization and boundary estimation for arbitrarily shaped land-

marks in medical images. This is particularly useful in finding tumors and cysts with varying sizes and shapes in images taken from the patients and successfully locating and masking them using RL methods. This is comparable to methods mentioned in section 2.1 that perform semantic and instance segmentation. Similarly, [79] uses deep RL to perform 3D-landmark detection in CT scans in real-time. [180] make use of a deep RL model to localize organs in the body by running agents on CT images. And in [151] authors employ deep RL and Q-learning techniques for the task of lung cancer detection.

5.2 More Applications

In previous section we went through the most prominent deep learning models, saw their progression through the years, and learned about a few of their use cases mostly in the medical industry. Aside from these models, deep learning and artificial intelligence, in general, can do much more and be utilized in things other than specialized disease detection and prediction models. Just as an instance in [175] authors try to analyze the spread of diseases such as influenza and Covid-19 in relation to social behaviors and environmental factors. In this section, we review various tools and fields that benefited from advancements in deep learning and in return made the lives of patients and clinicians a lot easier.

5.2.1 Internet of Things

As time goes on deep learning gets more and more intertwined with other fields of study and medicine, military, and internet of things (IoT) are no exceptions. IoT is a system of interconnected devices that altogether aims to increase the quality of life of humans and make their lives easier. The sustainable, autonomic, and ubiquitous characteristic of IoT makes it perfect for all sort of tasks such as smart cities [176], [224], smart homes and grids [130], and smart healthcare [146]. Smart healthcare, also known as Ubiquitous health (uHealth), electronic health (eHealth), and mobile health (mHealth), enables us to not only automate a big part of traditional clinical workflow, but also sufficiently monitor patients health in a continuous and ubiquitous manner even after discharge and remotely while at their homes or workplace using modern wearables and implantable devices. Implants are utilized for tasks such as heart pacemakers [53], and glucose monitoring [44] and are usually inserted into the body by light operations and invasive methods. On the other hand, wearables are worn as accessories. They come in various types such as

smartwatches, bracelets, and smart rings and are able to monitor heart rate, blood pressure, sleep time, body temperature, and much more.

Aside from monitoring features, these devices can also embed a combination of pre-trained complex deep learning models discussed in previous sections in themselves and aid doctors in detecting and predicting various diseases. Also, the collected data would be sent to a central server at fixed intervals or if a significant change is detected via the sensors. Central servers can benefit from larger and more complex data feed and hardware resources to then extract deeper patterns and features and notify the corresponding doctor if needed. Thus, personalized treatments will also be more accessible.

Building on this, various telemonitoring frameworks and systems have been proposed by researchers using different machine learning and deep learning models and diverse transmission network technologies to monitor and solve all sorts of health conditions in varying locations and importances [126]. For instance, [83] uses all these technologies to monitor the health of war soldiers on the battlefield that altogether accelerating the search and rescue operation if an individual is injured. Aside from this, other living assistants such as smart medicine boxes [143] have been proposed to track the intake of different medicines and make sure that the patients consume the correct dosage at the right time, and if any deviation occurs they would notify the patient as well as the doctor to address the issue.

5.2.2 Computer-aided Diagnosis

Many types of imaging and radiology techniques, such as magnetic resonance imaging (MRI), radiography, ultrasound, thermography, and tomography, provide us with detailed and valuable images that are crucial for physicians and researchers and play a pivotal role in healthcare nowadays. Before this, doctors had to do an invasive surgery to perform an autopsy if it was a need to acquire additional info about the patient's condition. But now, imaging techniques offer several ways to prevent unnecessary surgeries and with the help from deep learning, huge multi-modal datasets can be formed using 3D and/or layers of 2D images [65]. Datasets containing these images are inherently high-dimensional and cannot be processed using traditional machine learning algorithms in real-time. Thus, deep learning technologies are vital to enable the analysis and visualization of such data and help physicians see and diagnose illnesses more effectively.

Computer-aided design (CAD) is essential in the development of biomedical systems for a variety of applications. It aids in the detection, diagnosis, prediction, analysis, and categorization of illnesses, as

well as the management and delivery of health care. Thanks to CAD, it is now possible and reasonably simple to use data from medical imaging techniques to construct comprehensive and detailed models of patients, as CAD models are able to capture and represent a patient's unique and complicated organ, bone, and tissue structure. Given these characteristics, aside from the usual use cases of disease diagnosis such as brain tumor [6], breast cancer [212], and lung cancer [244], they are increasingly being utilized in operations, particularly those involving the implantation of medical devices or prosthetics [103], as well as creating interactive tools for training future doctors, and surgeons [262]. Additionally, CAD models are the best approach to track the progression of diseases such as Alzheimer's over time [185].

Advancement of deep learning has made it now a part of most of the interdisciplinary research and genomics is not an exception. Gene clustering [264], gene expression [38], and phenotyping [229] are all different areas that genomics benefited from deep learning techniques in recent years as bigger raw sequences of data can be plugged into DL models and yet more complex patterns and relationships can be found faster and more accurately. A deeper level of integration between CAD and genomics gives birth to the computer-aided drug design discipline in which on a molecule-by-molecule level, sophisticated computer modeling of molecular dynamics is employed to anticipate how medications would interact with the biological architecture of the human body. Computational models are used to simulate and occasionally depict atom-to-atom processes. This leads to creating targeted treatment and drug delivery that can target specific cells at the molecular level and as a result, increase the chance of success while considerably reducing the adverse effects of the medication [149].

5.3 Challenges

5.3.1 Interpretability

Deep learning and machine learning techniques are wonderful for assessing data, but they might look to humans as black boxes, which is undesirable in healthcare. Because computers are here to assist the healthcare community and allow them to operate side by side with them, the models' outputs and corresponding decisions must be interpretable by physicians and clinicians to further examine them if necessary especially if they're wrong.

Machine learning algorithms have an easier time accomplishing this since their reasoning is typically not overly complex and hence easy to comprehend. For example, decision trees can easily illustrate their "decision tree" to demonstrate all of the underlying factors and features that lead to a certain conclusion. Given the intricate architecture of deep learning models, this would not be a simple operation, and the judgments would be difficult to track. To address this problem, researchers are investigating ways to build rules from neural networks and attempting to identify the most prevalent routes in them in order to make sense of their decisions [259]. While it comes to processing photos and videos, attention models have also been used to highlight the areas where the model focuses the most when deciding on a topic [257].

5.3.2 Transfer Learning

In healthcare industry incorporating expert knowledge into data is very expensive and time consuming. When the size of the database at hand is small, deep learning models such as CNNs are unable to extract enough features from the data and end up either not learning anything or getting overfit and not being able to perform well on similar unseen data. Transfer learning is the suggested method for overcoming this problem by transferring knowledge from one model to another. The human brain excels in transferring its knowledge from one activity to the next. We seldom learn a task from the scratch, preferring instead to build on previous experience with a related activity or topic and by doing so, we expedite our new learning process. Likewise, When there isn't a good enough dataset for the destination domain, but one exists for the source domain, transfer learning is employed by reusing parts of the model which have been already trained on a similar task as the basis for the new task at hand. This method has been shown to be highly practical. Furthermore, because the pre-trained weights are used, the new model just needs to train the final few layers, and hence it saves a lot of time and computing resources. The remaining layers will be either frozen and left untouched during the training or can be slightly fine-tuned [278].

5.3.3 Data Quality

Most of the datasets that can be found in the internet do not have exact same number of instances for each of the classes. This can cause a problem in many classification approaches as the model can overfit to the class with the most number of instances and get biased [221] to yield the best accuracy possible while

failing to extract useful features. Also, datasets are often noisy and have missing values in them, hence preprocessing techniques should be employed to deal with these issues and normalize the data before training our deep learning model. Additionally, sometimes we encounter the curse of dimensionality and that basically means that we have too many features and thus there's a need to use feature extraction and reduction approaches to remedy this problem [177].

5.3.4 Interoperability

When it comes to IoT, the entire obtained data should be appropriately saved for future use cases. But, this is not as simple as just putting everything into a database. Central servers can exchange data with other services. However, there is no assurance that the other servers will use the same data format standards, thus some kind of standard, such as interfaces or database level rules and schemas, must be incorporated. As a result, when dealing with numerous computer systems sharing and utilizing each other's information, semantic interoperability plays a pivotal role in the system.

Every piece of data must contain meta-information about various entities in order to provide context for the corresponding values to enable the possibility of connecting and linking these small pieces of data, as well as automatic reasoning and inference to eventually transform data into knowledge. The Resource Description Framework (RDF) [165] and the Web Ontology Language (OWL) [163] are two very distinct but complementary approaches to accomplish these goals, and query languages like SPARQL [192] may be used to query data from many sources all across the internet.

5.4 Conclusion and Final Remarks

Machine learning and deep learning have been a big part of every interdiciplinary research and when we talk about big data, health informatics is the first discipline that comes to mind. Given the amount of different DL architectures and the numerous places that each of them can be employed, in this chapter, we reviewed the most prominent DL models and went through some of their applications. To conclude, we also debated on some of the challenges that show up when we incorporate DL in healthcare. Equipped with the knowledge that we learned in the past three chapters, we can now begin to design, implement, and

evaluate many case studies. In the next chapter, we will explore different ways to create a model capable of analyzing different types of data in healthcare and computer science.

Chapter 6

Case Studies [1] [2]

[1] F Shenavarmasouleh [*], FG Mohammadi[*], MH Amini, and HR Arabnia. 2020. International Conference on Computational Science and Computational Intelligence (CSCI). 418-423.

Reprinted here with permission of the publisher.

[2] F Shenavarmasouleh[*], E. Asali[*], FG Mohammadi, PS Suresh, and HR Arabnia. 2021. Advances in Computer Vision and Computational Biology. 39-56.

Reprinted here with permission of the publisher.

[*] Authors contributed equally to this work.

6.1 Case Study I

Motivation: Recently, one of the biggest challenges of the world is COVID-19 death rates and spread cases. Some scientists have conducted significant research studies in different countries across the globe and their results [202], [245] have stated that a large number of factors may have a positive/negative impact in the COVID-19 death rates and spread cases.

Epstein [59] who was working on the West Nile virus in 2001, discovered that extreme climate changes and conditions may contribute to the spread of the virus in the United States and Europe. Early discovery in 2010 by Casanova *et al* [34] stated that the virus's transmission by droplets is boosted in dry and cold weather situations. Hence, considering the climatic conditions, and the high population distribution with a wide variety of cultures, there is a high potential for the USA to be one of the most affected countries by COVID-19. In this study, we aim to understand the climatic effects on the spread and death of COVID-19 in all the counties of the United States and we hope that our results can assist with the decision-making process of monitoring and handling the pandemic.

COVID-19 pandemic emanated in Wuhan, China in late 2019 and later spread worldwide. Johns Hopkins University Center for Systems Science and Engineering (JHU CSSE), ESRI Living Atlas Team, and the Johns Hopkins University Applied Physics Lab (JHU APL) have recently developed and maintained a repository for the COVID-19 and provided wonderful visual and interactive dashboards for it. Based on their data, Cook, Washington, and King, Illinois were the first two cities to report cases in the USA. The former was detected on Jan. 2 and the latter on Jan. 24.

Contribution: Our main contribution is to run Spatiotemporal analysis to model the effects of climate change factors in COVID-19 death rates and spread cases in the United States. We analyze how meteorological factors like maximum/minimum temperature per day may influence the death rates and spread cases of COVID-19 in all counties within 50 states of the USA, visualizing the locations with weather more susceptible to the disease death rates and spread cases.

Organization: The rest of this chapter is organized as follows. First, we review related works regarding different impacted factor analysis in COVID-19. We then present the methodology and platform that we use for this study.

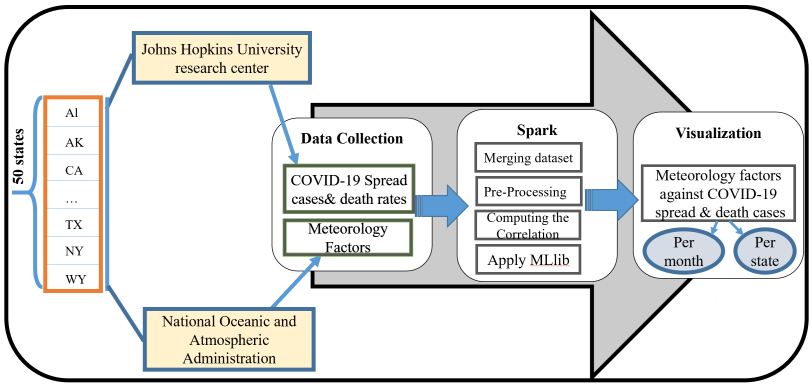


Figure 6.1: Proposed framework for big data analysis based on SPARK

Algorithm 1 The pseudocode of customized Pearson correlation in this study

Require: Input dataset $= \{x_0, x_1, x_2, ..., x_n\}$, Number of colums=n ▷ given input dataset
Ensure: correlation matrix : F_{n*n} ▷ a symmetric generated matrix.
  for i=0 $\cdots$ Number of colums **do**
    for j=0 $\cdots$ Number of colums **do**
      $\rho([col[i], col[j]) = \frac{E[col[i]col[j]]}{\sigma_{col[i]}\sigma_{col[j]}}$ ▷ $-1 < \rho([col[i], col[j]) < 1$
      if $|\rho([col[i], col[j])|$ close to 0 **then** : **return** *No Correlation*
      end if
      if $|\rho([col[i], col[j])|$ close to 0.5 **then** : **return** *Medium Correlation*
      end if
      if $|\rho([col[i], col[j])|$ close to 1 **then** : **return** *High Correlation*
      end if
    end for
  end for

6.1.1 Related Works

Researchers have done significant amount of work [23], [202] to analyze the impact of external triggers such as meteorological factors [16] and air pollution [23] in COVID-19 death rates and spread cases. Researchers in [204] analyzed recovered and death rates in the COVID-19 epidemic in which they studied the effect of factors such as sex, birth year, infection reasons, and region on the reported number of recovered and deceased cases.

Further impacts of the pandemic are to increase as time goes on, and there might be some positive impact on the tourism industry and/or on certain tourist destinations [86]. It is noteworthy that most

of these works are done in different countries across the world and since their result is general. Moreover, there are a few research studies that compute impact of external factors on COVID-19 death rates spread cases for USA. In this chapter, we address this issue and propose a new framework to do data analysis to find the impact of meteorology factors on COVID-19 condition.

6.1.2 Methodology

Taking action in time, and have faith in scientific results are the key to decrease climate changes impact on the COVID-19 death rates and spread cases [157]. In this chapter, we propose a framework for Spark-based big data analysis, an overview of which is presented in figure 6.1. This figure encompasses three main phases: Data Gathering, Spatiotemporal Analysis using Spark, and Visualization, which stands for the important phase indicating the consistent results.

Data Gathering

We collect our initial raw dataset for COVID-19 and meteorology factors from two individual separate sources. They are collected for a period of six months, more specifically from February 1 to August 31. COVID-19 datasets are stored with a variety of data types on sources like Github repositories where we find it easy to download and keep track of the death rates and spread cases. However, gathering the weather information for all 50 states was not a trivial task. We had to manually submit a request to download them individually per state for a limited period of time. We found some states like California and Texas where they have a plethora of stations that prevented us from downloading relevant data for certain periods. Hence, it can be deduced that the task at hand is a time and data-intensive problem.

Challenges

The first problem of this big data analysis is the unmatched data between weather and COVID-19 repository. The weather data has been gathered city by city by their corresponding stations. On the contrary, the data of COVID-19 has been collected county by county. Thus, we need another dataset (bridge dataset) or an algorithm that knows how to match these two datasets and merge them. To that end, we find a dataset that enables us to find a bridge to merge these two datasets.

The second challenge is our sparse datasets for meteorology factors. The majority of these factors per day are not available and we need to either remove them or replace the missing values with a suitable value like mathematical mean or mode. The latter challenge makes the data analysis process not looking realistic. However, we try to find a value that is fair enough without considering any biased information to ensure reproducibility of the results [221].

Table 6.1: Comparing correlation results for Georgia, Albany and Florida State during (Feb. 1st-Aug. 31)

state	Georgia		Alabama		Florida	
COVID-19 / **Meteorology**	Death	Spread	Death	Spread	Death	Spread
Pearson correlation						
TMAX	0.4321	0.7260	0.5092	0.7211	0.6399	0.4265
TMIN	0.4244	0.7033	0.4713	0.7143	0.3391	0.5760
TAVG	0.4285	0.7209	0.4885	0.7213	0.5626	0.2535
Spearman correlation						
TMAX	0.5110	0.8515	0.5838	0.8735	0.7638	0.8343
TMIN	0.4903	0.8005	0.5401	0.8348	0.4837	0.7101
TAVG	0.4996	0.8395	0.5692	0.8699	0.7594	0.8354

Table 6.2: Comparing correlation results for New York, Michigan and Illinois during (Feb. 1st-Aug. 31)

state	New York		Michigan		Illinois	
COVID-19 / **Meteorology**	Death	Spread	Death	Spread	Death	Spread
Pearson correlation						
TMAX	-0.2320	-0.2313	-0.1370	0.2632	0.1913	0.4929
TMIN	-0.2299	-0.2150	-0.1463	0.2774	0.1695	0.4679
TAVG	-0.2418	-0.2374	-0.1429	0.2663	0.1794	0.4849
Spearman correlation						
TMAX	0.0704	0.0690	0.1548	0.3358	0.3323	0.4362
TMIN	0.0526	0.0742	0.1413	0.3791	0.3248	0.4471
TAVG	0.0430	0.0604	0.1500	0.3512	0.3178	0.4458

Spark Platform

In this section, We discuss in detail the steps we need to take to compute spatiotemporal analysis using Spark in order to seek a correlation between meteorology factors and COVID-19 death rates and spread

Table 6.3: Comparing correlation results for California, Washington and Oregon during (Feb. 1st-Aug. 31)

state	California		Washington		Oregon	
COVID-19 / **Meteorology**	Death	Spread	Death	Spread	Death	Spread
Pearson correlation						
TMAX	0.0757	-0.1564	0.1994	0.6581	0.4818	0.7818
TMIN	0.6565	0.7267	0.1603	0.6543	0.4523	0.7698
TAVG	0.6331	0.8306	0.1774	0.6578	0.4743	0.7873
Spearman correlation						
TMAX	0.0606	-0.1438	0.3156	0.7165	0.5191	0.8003
TMIN	0.7001	0.8765	0.2685	0.7132	0.4945	0.8058
TAVG	0.6765	0.8813	0.2886	0.7101	0.5148	0.8064

cases. Our exploratory analysis of the data is carried out for each county within each state with respect to COVID-19 death rates and spread cases using SPARK, which is the computational tools for data analysis in this work.

We aim to use tools and libraries such as SPARK and MLlib for big data analysis. There are some advantages to applying those tools to our datasets. In this study, we use different libraries along with MLlib, which is Apache Spark's scalable machine learning library. Its main goal is to develop a framework to make machine learning scalable and easy. In general, MLlib includes ML Algorithms like classification, clustering, regression, and collaborative filtering. In this work, we leverage MLlib to apply machine learning algorithms to the generated warehouse.

Firstly, we merge these two big datasets where one has metadata based on cities and another one has metadata based on counties using a bridge dataset that enables us to match these two datasets. Spark facilitates this process with its powerful architecture. The reason we want to use this bridge dataset is that this dataset plays like a tool to match cities to their counties. Secondly, we do a pre-processing step by transforming the merged dataset into a standard dataset by replacing missing value, removing irrelevant factors, and generate a warehouse. The meteorology-factor dataset has lack of value in their columns, thus we only choose three important factors(columns), which have values in most days. we replace those missing value by computed average of each associated column. This process all is done by spark functions.

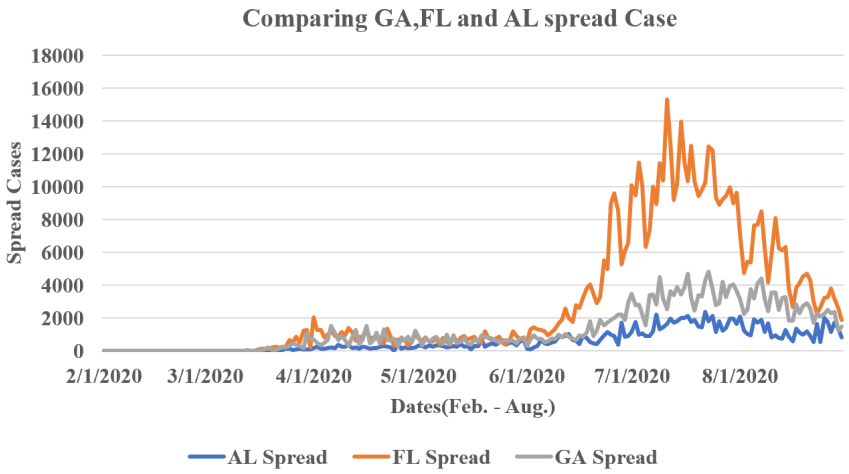


Figure 6.2: COVID-19 Spread plot for Georgia,Alabama and Florida

Thirdly, we compute correlation among meteorology factors and Covid-19 death rate and spread cases and clustering them into relevant categories.Correlation is important factor to determine whether one variable is important with respect to the target value or no, there are some works like [14] leverages correlations as feature selection, looking for relevant features, within the speaker recognition task. Furthermore, we leverage spark based machine learning library called MLlib to evaluate the goodness of each category.

Finally, in the last section, we take the generated warehouse and visualize data per month and state. This step plays a pivotal role in indicating the impact of meteorology factors on states COVID-19 spread cases and death rate visually.

6.1.3 Evaluation

In this section, we aim to evaluate the generated dataset (data warehouse) of meteorology factors and COVID-19 death rate and spread cases. To that end, we propose to use two important criteria to test the impact of these factors and how they relate to one another. We apply two popular statistic correlations, borrowed from Spark MLlib library. The Pearson correlation coefficient [190] and the Spearman rank correlation coefficient [236] were proposed in the last century or so.

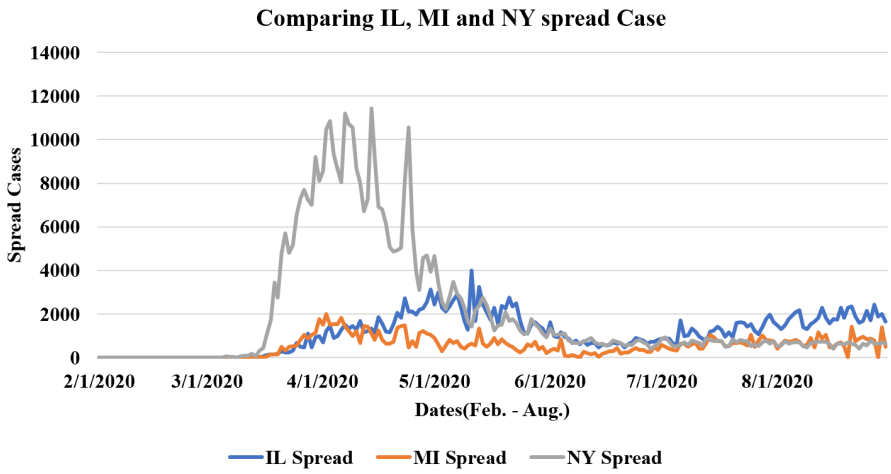


Figure 6.3: COVID-19 Spread plot for Illinois, Michigan and New York

Evaluation criteria

◇ **Pearson Correlation** The idea of correlation had been proposed earlier, but Mr. Pearson in [190] proposed Pearson correlation and later on other researchers discussed in consecutive years [55] in 1987 and [141] in 1988. In statistics, the Pearson correlation coefficient (PCC) is a statistic that evaluate linear correlation between two input variables X and Y, which would be two columns of the given dataset. We provide a pseudocode of pearson correlation calculation [190] for a given dataset in the proposed framework in algorithm 1.

◇**Spearman Correlation** Spearman correlation coefficient is proposed in 1904 [236] and it computes the correlation between two variables the same way Pearson does. However, before computing the Algorithm 1, we need to transform these two variables into new space using ranking system [256]. Then, The algorithm takes these two ranking variables to calculate the correlation between those variables.

Experimental results

◇**State-based Result** We experiment with 9 randomly selected states considering three geography points of view: hot, cold and medium. Hot states: Georgia, Alabama and Florida; cold states: New York, Michigan and Illinois; medium states: California, Oregon and Washington. Tables which appear below present the two computed correlations (Pearson and Spearman) between meteorology factors against CVOID-19

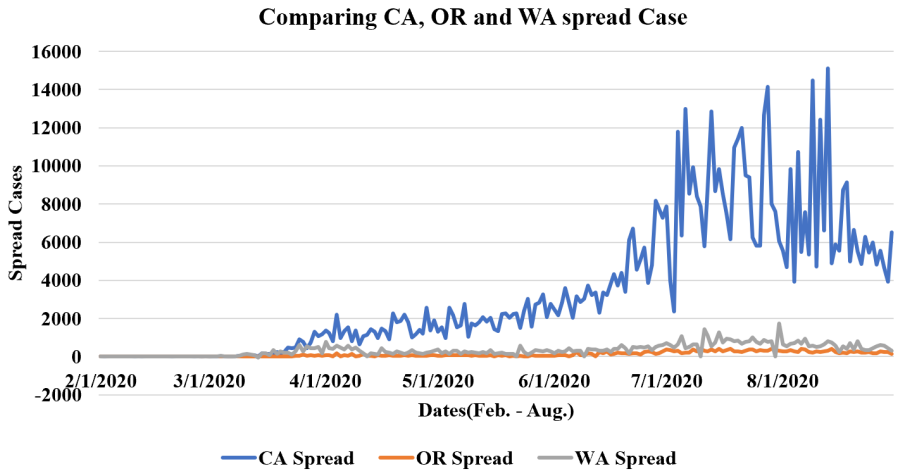


Figure 6.4: COVID-19 Spread Demographic plot for California, Oregon and Washington

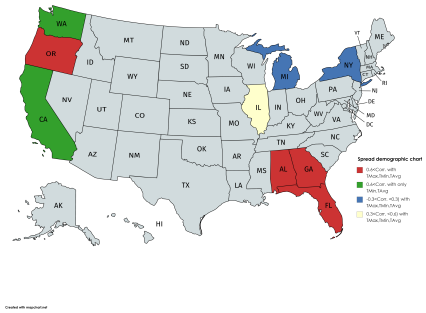


Figure 6.5: COVID-19 Spread Demographic chart for 9 states

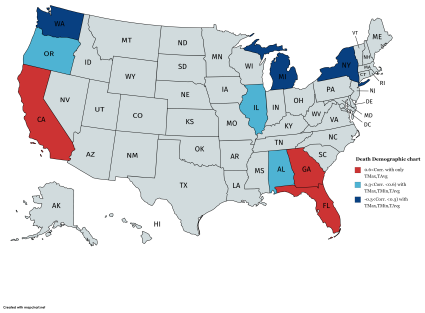


Figure 6.6: COVID-19 Death Demographic chart for 9 states

*Note that we generate these two figures using a website: https://mapchart.net/usa.html, access date Nov. 2020

death rate and spread cases where TMAX stands for maximum temperature, TMIN stands for minimum temperature and TAVG stands for average temperature. Among all these state, we observe that Spearman correlation states provides better result in comparison with Pearson correlation. Table 6.1 provides the correlations for states like Georgia, Alabama and Florida. This table results express that these three states behave almost the same with respect ot the meteorology factors for spread distribution cases. On the contrary, Alabama has opposite behavior on death rate whereas Georgia and Florida still have the same behavior.

Table 6.2 shows that cold states: New York, Michigan and Illinois do not have strong correlation with COVID-19 death rate and spread cases. All these states behave almost independent of COVID-19 factors.

Furthermore, we observe in table 6.3 that medium states like California, Washington and Oregon have high correlation with COVID-19 factors except TMAX for Oregon state.

◇**Month-based Result** In this section, we would like to illustrate COVID-19 spread cases per month for each selected hot, cold and medium state. Figures 6.2, 6.3 and 6.4 illustrate that COVID-19 spread within each category looks consistent among states.

6.1.4 Discussion and future work

We evaluate results of these tables based on the data warehouse generated, and discover some interesting relationships among these 9 selected states with respect to COVID-19 factors. We draw demographic chart of COVID-19 death rate and spread cases on US map. Figure 6.5 presents US map with 4 different colors and classifications. This figure shows that only Illinois is the only state out of 9 selected states where its correlations relies between 0.3 and 0.6. Additionally, figure 6.6 illustrates the behaviour of states with respect to the COVID-19 death rate. It is interesting that Illinois and Alabama and Oregon have the same correlation category with respect to COVID-19 cases.

In future, we aim to compute the correlation for all states and find the same relationships if any exists and apply our proposed method to all states. Then, we plan to present them on the US map to search for states have similar treat with others with respect to COVID-19 factors.

6.1.5 Conclusions

In this chapter, we dealt with a worldwide challenging situation, namely COVID-19. We kept track of COVID-19 death rates and spread cases from the first of February up to the end of August 2020. Furthermore, we analyzed meteorology factors like maximum and minimum temperature per day. In this chapter, we aimed to seek the correlation between meteorology factors and the COVID-19 situation in the United States. Furthermore, we aimed to cluster states according to their behavior toward COVID-19. We took advantage of two datasets for this study. After gathering and engineering data we merged them all to generate the expected data warehouse. Our results showed that the relationship between the weather conditions against the COVID-19 spread cases and the number of death rates in different states of the United States would be different.

6.2 Case Study II

Artificial Intelligence (AI) has impacted almost all research fields in the last decades. There exist countless number of applications of AI algorithms in various areas such as medicine, robotics, and multi-agent systems . Deep learning is, with no doubt, the leading AI methodology that revolutionized almost all computer science sub-categories such as IoT, Computer Vision, Robotics, and Data Science [173]. The field of Computer Vision has been looking to identify human beings, animals and other objects in single photo or video streams for at least two decades. Computer vision provides variety of techniques such as image/video recognition [174], image/video analysis, image/video captioning, expert's state or action recognition [231], and object detection within image/video [222]. Object detection plays a pivotal role to help researchers find the most matching object with respect to the ground truth. The greatest challenge of object recognition task is the effective usage of noisy and imperfect datasets, especially video streams. In this chapter, we aim to address this issue and propose a new framework to detect speakers.

Problem Statement

Copious amount of research has been done to leverage single modality which is either using audio or image frames. However, very little attention has been given to multimodality based frameworks. The main problem is speaker recognition where the number of speakers are around 40. In fact, when the number of classes (speakers) proliferate to a big number and the dimension of extracted features becomes too high, traditional machine learning approaches may not be able to yield high performance due to the problem of curse of dimensionality [77][76]. To explore the possibility of using multimodality, we feed the video streams to the proposed network and extract two modalities including audio and image frames. We aim to use feature selection techniques in two different phases in the proposed method.

Now this approach may prompt some questions: Why do we need multimodality if just single modality would give us a high enough accuracy? Is it always better to add more modalities or would an additional modality actually bring down the performance? If so, by how much? Bolstered by our experimental results, these are some questions we are going to delve into and answer in this chapter. Let's start by looking at the potential impediments we could run into while using a single modality. Let's say, for instance, we just use audio-based recognition systems; in this case, we often face a bottleneck called SNR(Signal-to-

Noise-Ratio) degradation, as mentioned in [40]. In short, when SNR is low in the input dataset, we observe our model efficiency plummets. On the other hand, image-based data is not unfettered by such predicaments as well. Images face problems like pose and illumination variation, occlusion, and poor image quality [178]. Thus, we hypothesize that combining the two modalities and assigning appropriate weights to each of the input streams would bring down the error rate.

Feature Selection

Feature selection is arguably one of the important steps in pre-processing before applying any machine learning algorithms. Feature selection or dimension reduction works based on two categories, including (i) filter-based and (ii) wrapper-based feature selection. Filter-based feature selection algorithms evaluate each feature independent of other features and only relies on the relation of that feature with target value or class label. This type of feature selection is cheap, as it does not apply any machine learning algorithms to examine the features. On the contrary, wrapper-based feature selection algorithms choose subsets of features and evaluate them using machine learning algorithms. That is the main reason why wrapper-based feature selection algorithms are more expensive. Mohammadi and Abadeh [170], [171] applied wrapper-based algorithms for binary feature selections using artificial bee colony. In this study, we apply wrapper based feature selection, as it yields a high performance on supervised datasets.

Contribution

Most of the speaker recognition systems currently deployed are based on modelling a speaker, based on single modality information, i.e., either audio or visual features. The main contributions of this work are as following:

- Integration of audio and image input streams extracted from a video stream, forming a multimodality deep architecture to perform speaker recognition.
- Effectively identifying the key features and the extent of contribution of each input stream.
- Creating a unique architecture that allows segregation and seamless end-to-end processing by overcoming dimensionality bottlenecks.

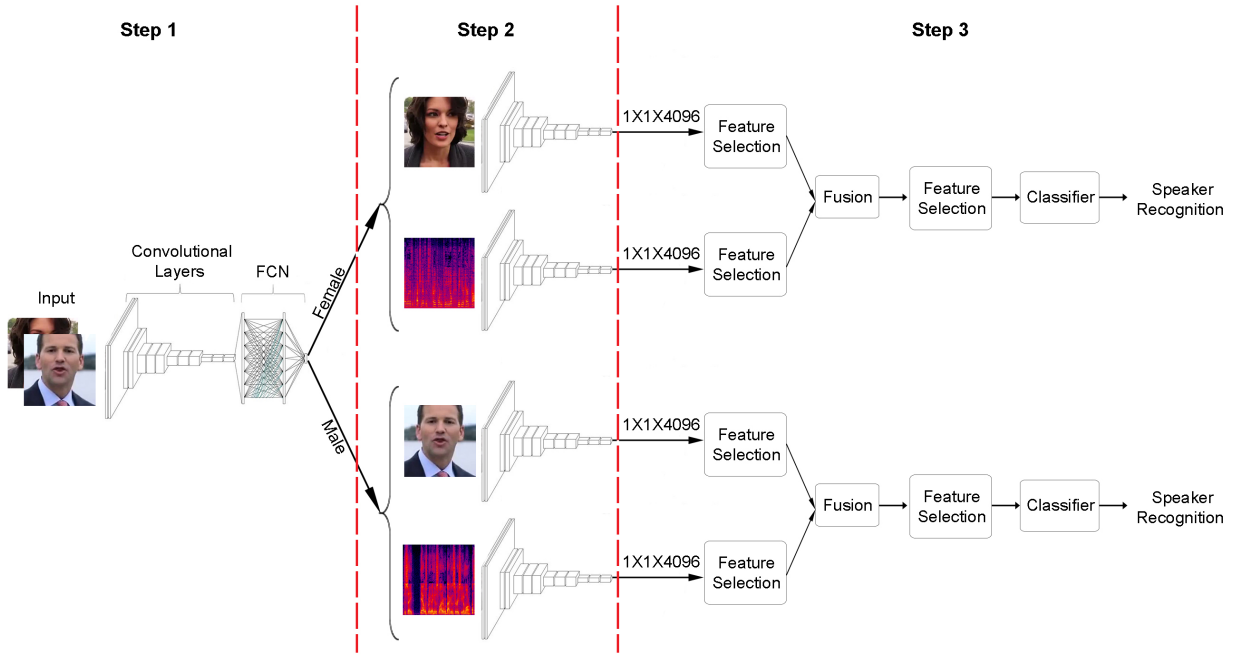


Figure 6.7: DeepMSRF architecture. Step 1: A unimodal VGGNET takes the speaker's image as an input and detects the speaker's gender. Step 2: Based on the gender, the image and voice of the speaker is passed to the corresponding parallel multimodal VGGNETs to extract each modality's dense features. Step 3: Feature selection on each modality will be applied first; then, the resultant feature vectors are concatenated, and feature selection is performed again after concatenation. Eventually, a classifier is trained to recognize the speaker's identity.

The rest of the paper is arranged as follows: First, we touch base with the related work that has been done in this field, then we explain the overview working methodology of CNNs, followed by how we handle the data effectively. We also compare and contrast other classifiers that could be used instead of the built-in neural network of VGGNET. Then, we explain the experiments performed, compare the results with some baseline performance and conclude with discussion, future work and varied applications of the model developed in this work.

6.2.1 Related Work

As explained before, most of the work done so far on speaker recognition is based on unimodal strategies. However, with the advancement of machine learning and deep learning in the past few years, it has been proven that multimodal architectures can easily surpass unimodal designs. [40][129] were some of the very first attempts to tackle the task of person identity verification or speaker recognition while leveraging multiple streams to combine data collected from different sources such as clips recorded via regular or thermal cameras, and the varieties of corresponding features extracted from them via different external speech recognition systems, optical flow modules and much more. After the features were extracted and fused, some basic machine learning models such as Hidden Markov Model (HMM), Latent Dirichlet Allocation (LDA), and Principal Component Analysis (PCA) were trained over them to act as the final classifier.

As impressive as these look, they can never beat the accuracy that one can achieve with deep learning models. Almost all of the architectures that are at the cutting edge in the modern tasks, make use of two or more streams. Video action recognition is one example. [62] employs two parallel streams, one for capturing spatial data and the other for extracting temporal information from videos to classify the actions. Similarly, [191] uses two separate streams for RGB frame and sequence of different flows, whereas [276] brings four modalities into play and makes use of 2D and 3D CNNs and their flows simultaneously. Another excellent work [231] that deals with multimodal inputs, suggests a unique framework for recognizing robots' state-action pairs which uses two streams of RGB and Depth data in parallel. More creatively, [61] utilizes one slow and one fast stream, proving that the former is good to understand spatial semantics, and the latter, which is a lighter pathway, can be beneficial in finding out the temporal motion features.

Tracking objects in videos, finding tampered images, and testing audio-visual correspondence (AVC) are some other tasks that parallel streams have been used for them to achieve the state of the art performance. [63] leverages two streams to jointly learn how to detect and track objects in the videos at the same time. [96] uses two parallel Siamese networks to do real-time object tracking and [283] employs face classification and patch triplet streams to investigate the possible alterations to the face images. Also, [13] uses parallel streams to enable their models to identify whether the input audio corresponds to a given video or not.

It can be perceived that an extensive amount of research has been done in the field of multimodal deep architectures. Nevertheless, to the best of our knowledge, [52] is the most related work that has been done for the task of speaker recognition and it only uses multimodality in the process of feature extraction. Additionally, it only uses audio data and tests on two datasets with 22 and 26 speakers, respectively. On the contrary, in this chapter, we design our architecture to make use of video frames along with the audios in separate streams. On top of that, we create our custom dataset with 40 unique speakers and extend the scale of previous works.

6.2.2 Proposed Method

In this section, we propose a late-fusion framework using a dual-modality speaker recognition architecture using audios and frames extracted from videos. Firstly, we discuss challenges in speaker classification and recognition, then we talk about the bottlenecks of the architecture, and finally we present our model's architectural details.

Challenges

As the number of images and videos proliferate, the process of image/video classification becomes more challenging; so the task of real world computer vision and data analysis becomes crucial when the number of classes exceeds 10. The more the classes, the more time and computational power is required to do the task of classification. To learn a model to classify the speakers, we are required to have a proper dataset and a structured framework to do that. In this work, the greatest challenge was to recognize 40 unique speakers. More so than that, since no standard dataset is available for our hypothesis, we had to create ours by subsampling from a combination of two other datasets. During the last ten years, researchers have proposed different frameworks for deep learning using complex combination of neural networks such as ResNet [98] for image classification. These only focus on singular modality, either image or voice of the video, to do speaker recognition. In this work, we address this problem and propose a new architecture leveraging VGGNET (VGG-16) [282] for speaker recognition task using multimodality to overcome all these limitations. The simple VGGNET, like other frameworks, suffers from having insufficient performance on speaker recognition. We provide three main steps consisting of combining two networks of VGGNET followed by feature selection and performing late-fusion on top of them.

Another common conundrum is on how to interpret the audio signals into a format which is suitable for VGGNET to work with. In general, in order to deal with audio streams, we have three options to choose from. One is to map the input audios to waveform images and feed the resultant diagrams to VGGNET as input. Another choice is to apply feature extraction to obtain a meaningful representation of the audio streams which is now a feature vector. The last but not the least, we can perform one more step on top of feature extraction by visualizing them as a two-dimensional image. Later in this chapter we will see that the third choice has the best performance and is utilized in our final model.

Video Speaker Recognition

We present a base speaker recognition architecture that leverages two VGG16 networks in parallel; One for speakers' images and another for speakers' audio frames. We discuss generating speaker audio frames in the next subsection. Figure 6.7 illustrates the base speaker recognition architecture. VGGNET produces a 1-D vector of 4,096 features for each input frame that could be used as an input to all common classification methodologies. Fusing these feature vectors yields to high dimensionality problem called curse of dimensionality (COD) [77][76]. To reduce the problem's complexity, we apply feature selection as discussed earlier in section 1.2.

Data Preparation

We prepare a standard dataset of speaker images, along with speaker audios trimmed to four seconds, about which we discuss further in the dataset subsection under the experiment section. The dataset of the speakers' face images can be created quite easily; Nevertheless, as previously mentioned, we should convert speaker voices into proper formats as well. To tackle this issue, we have tried various available methods to generate meaningful features out of input audio signals. There are a couple of choices which we have briefly mentioned previously and will investigate more in this section.

The first approach is to directly convert the audio files into wave form diagrams. To create such images, the main hurdle that we face is the frequency variation of the speakers' voices. To solve this issue, we plot them with the same y-axis range to have an identical axes for all the plots. Obviously, the y-axis length must be such that all wave form charts fall within its range. The generated images can directly be fed

into VGGNET; Nonetheless, later we will see that this approach is not really useful because of lacking sufficiently descriptive features.

Another approach is to extract meaningful and descriptive features of the audio streams instead of just drawing their waveform diagrams. Here, we have multiple options to examine; Mel-frequency cepstral coefficients (MFCCs), Differential Mel-frequency cepstral coefficients DMFCCs, and Filter Banks (F-Banks) are the algorithms reported to be effective for audio streams. MFCCs and DMFCCs can each extract a vector of 5,200 features from the input audio file, while F-Banks can extract 10,400 features. Now we have the option of either using these feature vectors directly and concatenating them with the extracted feature vectors of the face images coming from VGGNET Fully-Connected Layer 7 (so-called FC7 layer) or mapping them first to images and then, feeding them into the VGGNET. In the latter, we first need to fetch the flattened FC7 layer feature vector; afterwards, we have to perform the concatenation of the resultant vector with the previously learned face features. Spectrograms are another meaningful set of features we use in this work. A spectrogram is a visual representation of the spectrum of frequencies of a signal (audio signal here) as it varies with time. We feed such images into VGGNET directly and extract the features from the FC7 layer. Later, we will see how beneficial each of the aforementioned approaches is to predict the speakers' identity.

Feature selection

The more features we have, the higher is the probability of occurring overfitting problems which is also known as Curse of Dimensionality. This can be resolved to some extent by making use of feature selection (FS) algorithms. Feature selection approaches carefully select the most relevant and important subset of features with respect to the classes (speakers' identities). We choose a wrapper-based feature selection by exploiting lib-SVM kernel to evaluate the subsets of features. After applying feature selection, the dimensionality of the dataset decreases significantly. In our work, we apply FS two times in the model; once for each of the modals separately, and one again after concatenating them together (before feeding the resultant integrated feature vector to the SVM classifier.

6.2.3 Extended video Speaker Recognition

To do the task of video speaker recognition we have divided our architecture into three main steps as presented in fig. 6.7. The following sections explain each step in a closer scrutiny.

Step One

Inevitably, learning to differentiate between genders is notably more straightforward for the network compared to distinguishing between the identities. The former is a binary classification problem while the latter is a multi-label classification problem with 40 labels (in our dataset). On the other hand, facial expressions and audio frequencies of the two genders differ remarkably in some aspects. For instance, a woman usually has longer haircuts, a smaller skull, jewelries, and makeup. Men, on the contrary, occasionally have a mustache and/or beard, colored ties, and tattoos. Other than facial characteristics, males mostly have deep, low pitched voices while females have high, flute-like vocals. Such differences triggered the notion of designing the first step of our framework to distinguish the speakers' genders.

Basically, the objective of this step is to classify the input into Male and Female labels. This will greatly assist in training specialized and accurate models for each class. Since the gender classification is an easy task for the VGGNET, we just use the facial features in this step. In fact, we pass the speakers' images to the VGGNET and based on the resultant identified gender class extracted from the model, we decide whether to use the network for Male speakers or Female ones. In our dataset, such a binary classification yields 100 % accuracy on the test set. Later we will see the effect of this filtering step on our results.

Step Two

In this step, we take the separated datasets of men and women as inputs to the networks. For each category, we apply two VGGNETs, one for speaker images and another for their voices. Thus, in total, we have five VGGNETs in the first and second steps. Indeed, we had one singular modality VGGNET for filtering out the speakers' genders in the first step, and we have two VGGNETs for each gender (four VGGNETs in total) in the second step. Note that the pipeline always uses the VGGNET specified for the gender recognition in the first step; Afterwards, based on the output gender, it chooses whether to use the parallel VGGNET model for women or men, but not both simultaneously.

In the given dataset, we have 20 unique females and 20 unique males. Following this, we train the images and audios of males and females separately on two parallel VGGNETs and extract the result of each network's FC7 layer. Each extracted feature vector consists of 4,096 features which is passed to the next step.

The second step may change a little if we use the non-visualized vocal features of either MFCCs, DMFCCs, or F-Bank approach. In this case, we only need VGGNET for the speakers' face images to extract the dense features; Following this, we concatenate the resultant feature vector with the one we already have from vocal feature extractors to generate the final unified dataset for each gender.

Step Three

After receiving the feature vectors for each modality of each gender, we apply a classifier to recognize the speaker. Since the built-in neural network of VGGNET is not powerful enough for identity detection, we try a couple of classifiers on the resultant feature vector of the previous step to find the best classifier for our architecture. Nonetheless, before we feed the data into the classifiers, we need to ensure the amount of contribution each modality makes to the final result. As the contribution of each modality on the final result can vary according to the density and descriptivity of its features, we need to filter out the unnecessary features from each modality. To do so, we apply feature selection on each modality separately to allocate appropriate number of features for each of them. Afterwards, we concatenate them together as a unified 1-D vector. We apply feature selection again on the unified vector and use the final selected features as input to the classifiers. The specific number of data samples, the number of epochs for each stage and the results at each checkpoint are discussed in the Experiments section (section 4).

6.2.4 Experiments

Dataset

We have used VoxCeleb2 dataset proposed in [42] which originally has more than 7,000 speakers, 2,000 hours of videos, and more than one million utterances. We use an unbiased sub-sample [221] of that with 20,000 video samples in total, with almost 10,000 sample speakers per gender. The metadata of VoxCeleb2 dataset has gender and id labels; the id label is connected to the VGGFace2 dataset. The first step we have

to go through is to bind the two metadata sets together and segregate the labels correctly. The way their dataset is arranged is that a celeb id is assigned to multiple video clips extracted from several YouTube videos which is almost unusable. Hence, we unfolded this design and assigned a unique id to each video to make them meet our needs.

As mentioned earlier, we selected 40 random speakers from the dataset which included 20 male and 20 female speakers. Thereupon, one frame per video was extracted where the speaker's face was clearly noticeable. The voice was also extracted from a 4-second clip of the video. Finally, the image-voice pairs shuffled to create training, validation, and test sets of 14,000, 3,000 and 3,000 samples respectively for the whole dataset, i.e. both genders together.

Classifiers

Random Forests

Random forests [104] or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests prevent the overfitting which is common in regular decision tree models.

Gaussian Naive Bayes

Naïve Bayes [120] was first introduced in 1960s (though not under that name) and it is still a popular (baseline) method for classification problems. With appropriate pre-processing, it is competitive in the domain of text categorization with more advanced methods including support vector machines. It could also be used in automatic medical diagnosis and many other applications.

Logistic Regression

Logistic regression is a powerful statistical model that basically utilizes a logistic function to model a binary dependent variable, while much more complicated versions exist. In regression analysis, logistic regression [128] (or logit regression) is estimating the parameters of a logistic model. Mathematically, a binary logistic

model has a dependent variable with two possible values, such as pass/fail which is represented by an indicator variable, where the two values are labeled "0" and "1". In the logistic model, the log-odds (the logarithm of the odds) for the value labeled "1" is a linear combination of one or more independent variables ("predictors"); the independent variables can each be a binary variable (two classes, coded by an indicator variable) or a continuous variable (any real value). The corresponding probability of the value labeled "1" can vary between 0 (certainly the value "0") and 1 (certainly the value "1"), hence the labeling; the function that converts log-odds to probability is the logistic function, hence the name. The unit of measurement for the log-odds scale is called a logit, from logistic unit, hence the alternative names. Some applications of logits are presented in [9].

Support Vector Machine

A Support Vector Machine [100] is an efficient tool that helps to create a clear boundary among data clusters in order to aid with the classification. The way this is done is by adding an additional dimension in cases of overlapping data points to obtain a clear distinction and then projecting them back to the original dimensions to break them into clusters. These transformations are called kernels.

6.2.5 VGGNET Architecture

This section briefly explains the layers of our VGGNET architecture. Among the available VGGNET architectures, we have chosen the one containing the total of 13 convolutional and 3 Dense layers, famed as VGG-16 [227]. The architecture includes an input layer of size $224 \times 224 \times 3$ equal to a 2-D image with 224 pixels width and the same height including RGB channels. The input layer is followed by two convolutional layers with 64 filters each and a max pooling layer with a window of size 2×2 and the stride of 2. Then another pair of convolutional layers of size 112×112 with 128 filters each and a max pooling layer are implemented. Afterwards, in the next three stages, the architecture uses three convolutional layers and one pooling layer at the end of each stage. The dimension of the convolutional layers for these steps are $56 \times 56 \times 256$, $28 \times 28 \times 512$, and $14 \times 14 \times 512$, respectively. Finally, it has three dense layers of size $1 \times 1 \times 4096$ followed by a softmax layer. Since the output of the softmax layer specifies the output label (e.g. the speaker's name), its size must be equal to the number of classes. Also, notice that all convolutional and dense layers are followed by a ReLU function to protect the network from having

negative values. Moreover, the first Dense layer is usually referred as FC7 layer (Fully Connected layer 7) that contains an extracted flattened feature vector of the input.

6.2.6 Implementation

In section 4.1, we explained how we create the dataset and now we elucidate the steps taken to produce the results. In order to train the parallel VGGNET for each gender, we divide the dataset into two parts; the samples of the 20 Male speakers and the samples of the 20 Female speakers. Thereafter, each of the two partitions is fed into a dual-channel VGGNET consisting the image and audio streams. When the training process finishes, the architecture learns to extract meaningful features from the input data. Now, we can generate a new dataset for each gender by passing the face and Spectrogram's train, validation, and test images through their corresponding VGGNETs and fetch their FC7 layers' feature vectors.

Afterwards, using the linear SVM feature extractor library in Scikit learn - Python, we are able to extract almost 1,053 number of features for Male images, 798 features for Male voices, 1,165 features for Female images and 792 features for Female voices. Then, we concatenate the resultant feature vectors for each gender and feed it again to the same feature extractor to summarize it once more. The final size of the merged feature vectors for the Males is 1,262 and for Females is 1,383. Note that the reported number of voice features are related to the Spectrogram feature extractor which is the one we elected among the available options that were discussed earlier in section 3.3. The last step is to train the Linear SVM classifier and to get its result.

The very first baseline architecture that we are going to compare our results with, does not segregate genders, uses only one modality (i.e. either the face or the voice data), uses the plotted wave form of the voice data, and does not use any feature selection approach. To compare the effect of any changes to the baseline, we have accomplished an extremely dense ablation study process. The ablation study results are discussed in the next section.

6.2.7 Ablation Study

To check the effect of each contribution, we perform ablation study by training and testing the dataset in various conditions. The following sections briefly discuss the impact of each contribution on the final result.

Feature Extraction and Selection

Feature Extraction (FE) is highly crucial in the learning process. The main contribution of deep learning pipelines over the classical machine learning algorithms is their ability to extract rich meaningful features out of a high dimensional input. Here, VGGNET plays this role for the face images of the speakers and also for the visualized vocal features. On the other hand, Feature Selection (FS) can prevent the model to be misled by irrelevant features. As previously mentioned, we have used linear-SVM feature selection in this work.

To evaluate the advantage of using FE and FS when dealing with audial data and also to compare the performance of diverse FE algorithms, we apply each algorithm on Male, Female, and the whole dataset. Then, we apply FS on top of it; then, we examine each algorithm with four different classification methods, including Random Forests, Naive Bayes, Logistic Regression, and Support Vector Machines. Finally, we compare the results for both cases of either using or not using FS. Table 6.4 shows the result for all the situations. As the results represent, the best test accuracy is achieved when we utilize Spectrogram feature extractor combined with linear-SVM feature selection approach.

To analyze the efficacy of FS on the face frames, we train VGGNET and extract the FC7 layer feature vector. We then apply FS and eventually, train on four different classifiers. Table 6.5 represents the test accuracy for each classifier with and without FS. As the results demonstrate, the highest accuracy for each dataset is achieved for the case in which we have used FS on top of VGGNET and for the SVM classifier.

Gender Detection

As discussed in section 3.3.1 in details, the first step of our pipeline is to segregate speakers by their gender. Instead, we could train a model with 40 classes consisting of all men and women speakers. To see how the first step improves the overall performance of the model, we examined both cases and compared their

Table 6.4: The accuracy for single/multi modality with/without feature selection

classifier / **FE Algorithm**	RF	NB	LR	SVM
Spectogram(M) (%)	45.4	19.06	54.33	50.53
Spectogram(M) + FS (%)	45.93	25.93	52.86	**56.26**
Spectogram(F)(%)	44.26	21.53	52.26	48.46
Spectogram(F) + FS (%)	42.66	29.4	51.2	**53.3**
Spectogram(all)(%)	37.16	14.96	48.4	43.6
Spectogram(all) + FS (%)	38.03	21.6	46.5	**49.3**
Waveform(M)(%)	30.53	16.6	32.26	29.26
Waveform(M)(%) +FS	30.46	17.2	29.93	32.13
Waveform(F)(%)	22.06	14.08	27.73	21.4
Waveform(F)(%) +FS	22	13.93	23.26	23.6
MFCC(M)(%)	11.93	25.33	5.46	9.46
MFCC(M)(%) +FS	11.46	24.2	5.33	9.2
MFCC(F)(%)	10.8	21.86	5.93	9.46
MFCC(F)(%) +FS	10.8	21.53	5.93	9.73
Filter bank(M)(%)	32.33	19.13	42.06	36.6
Filter bank(M)(%) +FS	33	24	42.26	40.46
Filter bank(F)(%)	33.66	18.06	43.2	38.06
Filter bank(F)(%) +FS	33	25.46	41.93	41.06

results. The test accuracy of Male speakers, Female speakers, the average test accuracy of Male and Female speakers, and the test accuracy of the whole dataset (containing both genders) are reported in the table 6.6. The results show that the average accuracy increases when we perform gender segregation regardless of whether we use feature selection before and/or after concatenating the face and audio modalities or not. Also, notice that according to the table 6.6, we can achieve the highest accuracy when we perform feature selection, specifically before the concatenation step. According to the table, the average accuracy has been improved by almost 4 percent using our proposed method (the last row) compared to the baseline approach (the first row).

Table 6.5: The accuracy of the speakers' face images for four different classifiers associated with different feature extractors with/without Feature Selection (FS)

FE \ **classifiers**	RF	NB	LR	SVM
VGG(M) (%)	91	49.66	93.6	91.66
VGG(M) + FS (%)	91.26	66.73	92.53	**94.2**
VGG(F)(%)	86.26	55.33	90.93	87.33
VGG(F) + FS (%)	85.66	62.2	88.13	**91.26**
VGG(total) (%)	88.03	50.1	91.53	88.7
VGG(total) + FS (%)	88.06	58.43	90.4	**91.9**

Table 6.6: The accuracy of the whole dataset for SVM classifier associated with Spectrogram feature extractor combined with feature selection

approach \ **samples**	Male	Female	Avg.	Total(Genderless)
Simple concatenation	91.2	87.87	89.54	89.27
FS + concatenation	**95.13**	91.87	93.5	92.97
concatenation + FS	94.67	91.87	93.27	92.53
FS + concatenation + FS	95.07	**91.93**	**93.5**	**93.03**

Single modality Vs. Multimodality

One of the greatest contributions of DeepMSRF is taking advantage of more than one modality to recognize the speaker efficiently. Each modality comprises of unique features that lead the model to distinguish different individuals. To show how multimodality can overcome the limitations of single modality, we carry out a comparison between the two, reported in table 6.7. According to the results, using both visual and auditory inputs together can improve the accuracy of the task of speaker recognition.

Table 6.7: The Accuracy for single/multi modality with/out feature selection

Result \ **Modality**	Face Frames	Audio (Spectrogram)	Multimodality
Total (%)	88.7	43.6	89
Total + FS (%)	91.9	49.3	93.03

6.2.8 Time Complexity

In section 4.5, we saw the benefit of utilizing feature selection on the model's accuracy. Additionally, there exist one more criteria to consider which is the training time. Although the training process is being performed offline and in the worst case, training the SVM classifier over our dataset finishes in almost 20 minutes (for the whole dataset), it is noteworthy to see how feature selection can influence the training time. Figure 6.8 depicts the training time required for the SVM in the last step (step3) of our pipeline for the experiments shown in table 6.6. According to fig. 6.8, the required training time for each gender is approximately one third of the corresponding time required to train the whole dataset.

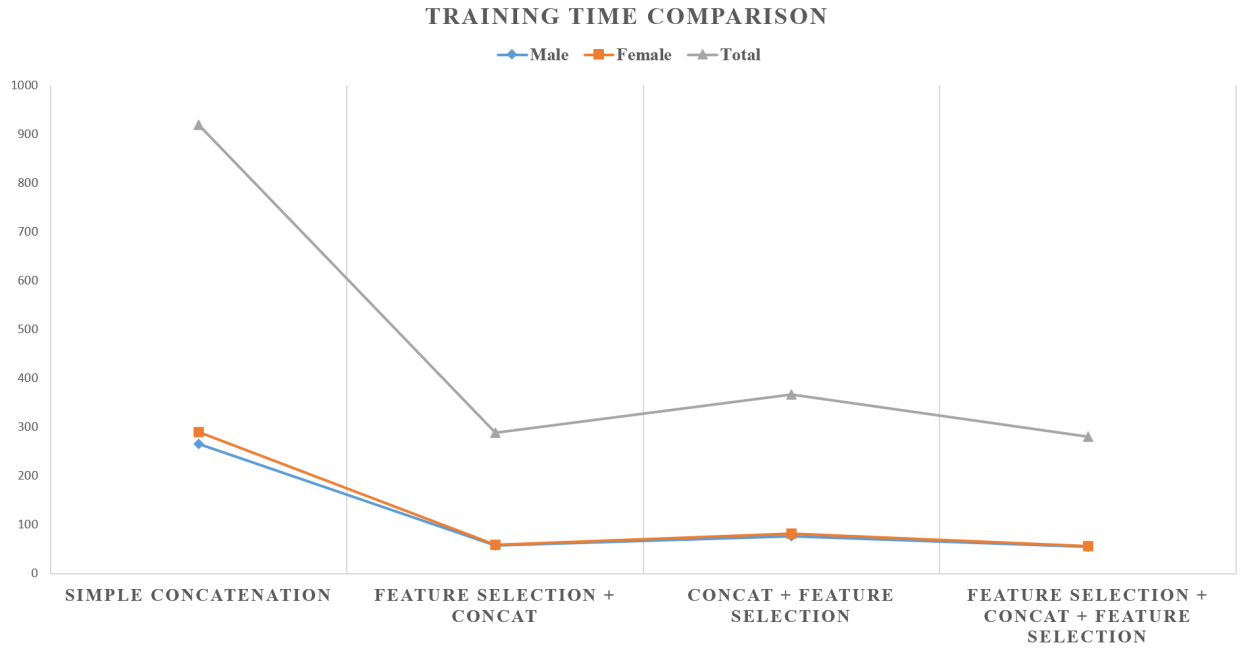


Figure 6.8: Training time Comparison (Male Vs. Female) for DeepMSRF with SVM classifier

Nextly, the time required to accomplish the step 3 of DeepMSRF, including the time needed for feature selection and the training time, is reported in table 6.8. The results can be discussed from two different points of view: (i) Among the examined methodologies, the least training duration is for the case in which we apply Feature Selection (FS) after the concatenation of the two modalities' feature vectors. On the contrary, the worst time performance is for the situation of applying FS before and after concatenation. (ii) The time is significantly shorter when we segregate the genders. Each gender's dataset needs less than

one third of the required time for the whole dataset. In fact, training two separate models (one per gender), together, requires lesser time than training a general model which contains both genders.

Table 6.8: The accuracy of the whole dataset for SVM classifier associated with Spectrogram feature extractor combined with feature selection

samples / **approach**	Male	Female	Total(Genderless)
Simple concatenation	265.37	290.94	919.46
FS + concatenation	209.91	195.49	883.23
concatenation + FS	179.61	197.27	729.6
FS + concatenation + FS	296.93	291.56	1,208.52

6.2.9 Conclusion

This work takes a trip down the novelty lane by adding multimodality to improve robustness of the recognition and overcomes the limitations of single modality performance. From the results of the experiments above, we can infer that the hypothesis made about the multimodality improving over the single modality results for person recognition using deep neural networks was nearly conclusive. Among other challenges, this work also solves the dimensionality challenge arising from using multimodality input streams. Exploiting feature extraction has provided a deep insight into how significant features to train the network are to be extracted to obtain a well-trained model. We can see that although the images provide a high accuracy over speaker recognition, audio stream input reinforces the performance and provides an additional layer of robustness to the model. In conclusion, we state that the unique framework used in this chapter, DeepMSRF, provides an efficient solution to the problem of speaker recognition from video streams. At last, DeepMSRF is a highly recommended framework for those researchers who deal with video analysis.

6.3 Final Remarks

In this chapter, we designed, implemented, and evaluated two case studies to demonstrate how machine learning and deep learning algorithms can be employed in healthcare to analyze big data, and also show that these algorithms can be tweaked to accept multi streams of data as input to improve the overall accuracy.

We will employ these as building blocks for our main contribution which we will start to discuss in the next three chapters.

Chapter 7

DRDr: Automatic Masking of Exudates and Microaneurysms Caused By Diabetic Retinopathy Using Mask R-CNN and Transfer Learning [1]

[1] F Shenavarmasouleh and HR Arabnia. 2021. Advances in Computer Vision and Computational Biology. 307-318. Reprinted here with permission of the publisher.

Diabetic retinopathy is a major cause of vision impairment, and eventually vision loss in the world; especially among working-aged individuals. Its diagnosis can be done by analyzing color fundus images by experienced clinicians to identify its presence and the significance of the damages that it has caused. Fundus images are the results of screenings. The procedure is easy and it can easily and safely be done via retinal photography in every clinic with the proper tools. If detected soon enough, diabetic retinopathy (DR) can be treated via laser surgeries. But, the demand is increasing much more rapidly than the supply. The annual number of patients is growing and each patient requires frequent screenings. Each of these images needs to be carefully analyzed by doctors. The task is innately time-consuming since the deficiencies are usually extremely small and require careful examination and the doctors need to find and weight countless features for each image. The thing is from all the patients being screened annually, only 25.2% have diabetic retinopathy and are referred to ophthalmologist [160] and it begs the question as to whether it is possible to use the time and resources more sufficiently?

Luckily, the answer to the above question is yes. During the past few years, Machine Learning and notably Deep Learning have shown high potential in helping health care and they can be used to aid doctors in detecting and predicting the development of various illnesses. In fact, machine learning and deep learning approaches have already helped numerous researchers to overcome some of the difficulties in the aforementioned task at hand. However, the majority of the previous work in this part lies in the machine learning section, and classification and especially binary classification was the primary subject of interest. In other words, previous work was mostly devoted to finding a way to automatically identify whether a patient has diabetic retinopathy or not.

Decencière et al. [49] leveraged a big dataset extracted from OPHDIAT [160], a teleophthalmology network, during 2008-2009 and the help of three experts to tackle this issue. They merged features extracted from images with the patients' contextual data such as age, weight, diabetic type, and the number of years of DR and altogether could predict whether the patient needs to be referred or not. Bhatia et al. [25] and Antal et al. [11] used ensembles of machine learning techniques, namely Decision Trees, Support Vector Machines (SVM), Adaboost, Naïve Bayes, and Random Forests, on Messidor dataset. Usher et al. [250] and Gardner et al. [72] employed Neural Networks to perform the task of classification for them. The former, utilized candidate lesions, their position, and their type as the inputs of the Neural Network and the latter used Neural Networks and pixel intensity values.

In Priya et al. [198], the authors explored Probabilistic Neural Networks (PNN) along with Naïve Bayes and SVM. They employed Adaptive Histogram Equalization, Discrete Wavelet Transform, Matched Filter Response, Fuzzy C-Means Segmentation, and Morphological Processing on top of the Green channel of the images for their preprocessing phase. The train/test split was questionable though, as out of 350 total images, the authors used 250 of them for test and only 100 for the training.

Sopharak et al. [234] made use of Naïve Bayes, SVM and K Nearest Neighbors (KNN) classifiers to detect exudates in pixel level. This task was traditionally being dealt with by using region growing and thresholding [57], [150]. They selected 15 handpicked features and operated on them. However, their dataset was extremely small with only 39 images.

Several attempts were made to extend the level of classification as well and drag the level of severity to this task too. Lachure et al. [134] utilized SVM and KNN to classify images of Messidor and DB-reet into 3 classes. Their model could tell whether a fundus image is normal or not; and if abnormal, whether it is grade 1 or 3. Roychowdhury et al. [205] used Gaussian Mixture Model (GMM), KNN, SVM, Adaboost along with feature selection to classify images of Messidor in 4 classes. Acharya et al. [1] and Adarsh et al. [2] also used SVM to deal with this problem and classified patients into 5 classes.

All of the forenamed approaches required an external feature extraction phase. Authors needed to manually perform multiple morphological operations, apply various filters to the images, and use techniques such as region growing and thresholding to extract features one by one and then fuse them with other contextual data, if any, and then use the resulting files as inputs for the different classifiers.

With the advancement of deep learning and specifically Convolutional Neural Networks (CNN), network architectures solely designed to enhance working with images, the task of feature extraction turned into an implicit phase instead. Gargeya et al. [73] made use of CNN to identify healthy and unhealthy patients using a huge dataset with 75 thousand images. Gulshan et al. [90] employed Inception V3, a more complex type of CNN, to classify images of EyePACS-1 and Messidor-2 into two categories. And finally, Pratt et al. [197] harnessed CNN and a huge publicly available Kaggle dataset to classify the fundus images into 5 classes.

In this chapter, we approach the problem from another angle and address the problem of automatically identifying the deficiencies caused by diabetic retinopathy in fundus images together with their exact shape and location. We modify and leverage a CNN-based model that can identify and use the intricate features

in the available images to detect, locate, and most importantly mask and label two important lesion types, namely microaneurysms and exudates. Due to the automatic behavior of our approach, it can easily fit into clinical systems and aid clinicians in the process of identifying unhealthy patients while saving them plenty of time. It has the potential to both be incorporated into retinal cameras and/or be used as a post-photography tool for optometrists and ophthalmologists.

The remainder of the paper is organized as follows: First, we touch base with the related works that are aligned with our interests. Then, we fully explain our methodology, including how we handle our limited available data effectively. Next, we show the experiments that we have performed and illustrate our results. Finally, we conclude with the discussion and future work.

7.1 Related Work

7.1.1 Convolutional Neural Networks

Computer Vision is the branch of computer science which its ultimate goal is to imitate the behavior and functionality of human eyes. It assists computers to analyze images and videos and ultimately understand objects that are present in them. Thanks to the advancement of Deep Learning in the past few years, we are now able to handle this task very well. Since AlexNet [133] won the ImageNet image classification competition in 2012, Convolutional Neural Networks (CNN) have become the go-to approach for any task that required dealing with images. In fact, nowadays, CNNs are so powerful that they even surpass humans' performance on the ImageNet challenge. Learning from Observation [231], and Embodied Question Answering [46] ,to name a few, are some of the very high-level tasks that implicitly use CNNs as one of their core components. Besides, researchers in the field of Meta-Learning and Neural Architecture Search (NAS) are constantly trying to find an innovative way to further improve the performance of these systems [174].

Image classification, Object Localization, Object Detection, Semantic Segmentation, and Instance Segmentation are 5 main Computer Vision problems; sorted by their level of difficulty in ascending order. In Image Classification problem, usually exists an image with a single main object in it and the goal is to predict what category that image belongs to. A tad more challenging task is Object Localization. In object

localization, the image usually contains one or more objects from the same category and the model's goal is to output the location of those objects as bounding boxes, a rectangular box around the object, besides predicting the category that they all are affiliated with.

As impressive as they look like, these tasks are not remotely as complex as what humans visual understanding and eyes are capable of doing.

Next is Object detection and recently a huge breakthrough has happened in it. CNNs work similarly to human eyes and they are able to detect edges and consequently define boundaries of the objects. Hence, they could be used to detect objects of different kinds in a given image. However, to do so, it is required to apply them to a massive number of locations with varieties of scales on each image, making it extremely time-consuming. As a result, an extensive amount of research has been done to tackle this issue.

R-CNN (Region-based CNN) [81] solves the aforementioned issue by making use of a region proposal module. This module proposes a collection of candidate bounding boxes, also known as Regions of Interests (ROIs), using the Selective Search technique. The pixels corresponding to each of these boxes are then fed into a pre-trained modified version of AlexNet to check if any object is present inside that box. On the very last layer of this CNN lies an SVM that judges whether the pixels represent an object or not and if yes, what is the category that goes with them. At last, if an object is found in the box, the box is tightened to best fit the object dimensions.

Training an R-CNN model is hard and time-consuming because approximately 2,000 ROIs are proposed for every single image and all of them need to be fed to the CNN individually. Besides, three different networks are ought to be trained separately.

Fast R-CNN [80] solved both of these problems. Normally, many of the regions that are proposed for further examination overlap with each other and hence this causes the CNN phase to do so many redundant computations. Fast R-CNN overcomes this issue by using only one CNN per image to compute all the features at once. The result is then shared and used by all the 2000 proposals, reducing the computational time significantly. This technique is called Region of Interest Pooling (RoIPool) in the original paper.

To tackle the second issue, Fast R-CNN united all the three models into one single network to enable jointly training of all of them. The SVM classifier was exchanged with a Softmax layer to handle the task

of classification and a regression layer was added in parallel to that to find and yield the best bounding box for each object.

However, Fast R-CNN still used the Region Proposal method using selective search to find the ROIs, which turned out to be the bottleneck of the overall process.

Faster R-CNN [200] was proposed to resolve this issue. The authors' main intention was to replace the selective search phase with something more efficient. They argued that the image feature maps that were already calculated with the forward pass of the CNN could be fed directly to a Fully Convolutional Network (FCN) on top of them to perform the task of region proposal instead of running a separate selective search algorithm. This newly added FCN was called Region Proposal Network (RPN) in the paper and this way, ROIs could be proposed almost for free, fixing the last problem present in the system.

As wonderful as object detection is, it still could not understand and provide us the actual shape of the objects and stops at delivering the bounding box only. This task, however, is where Image Segmentation comes into play and tackles the issue by creating a pixel-wise mask for each object. The task of image segmentation itself can be done in two main ways. In Semantic Segmentation, every pixel in the image needs to be assigned to a predefined class. In addition, all the pixels corresponding to a class are treated the same and are given an identical color and thus the differences among different object instances that belong to one class are disregarded. By contrast, in Instance Segmentation, each instance of the same class is treated discretely and given a unique color and label.

Mask R-CNN [97] is a model developed for the task of image instance segmentation. It extends Faster R-CNN, goes one step further, and specializes in generating pixel-level masks for each object in excess of finding the bounding box and the class label. It, creatively, adds another FCN on top of RPN, altogether creating a new parallel branch to the Fast R-CNN model which outputs a binary mask for the object found in a given region. It is also worth noting that the authors needed to slightly modify the RoIPool to fix the problem of location misalignment caused by its quantization behavior. They called the modified technique RoIAlign.

7.1.2 Transfer Learning

Humans are really good at transferring their knowledge across tasks. We rarely learn a certain task from scratch and instead, we tend to leverage our previous knowledge that we have acquired in the past in some

similar activity or topic. By doing so, we utilize and accelerate our new learning process. Traditional machine learning and deep learning algorithms are designed to work in insulation and learn to handle only a domain-specific task. To make a model work for another task or domain, the entire model has to be retrained from scratch, and thus not taking advantage of the previously learned task will result in consuming so much more time and resources than required, as plenty of redundant operations are needed to take place. Besides, often a huge amount of labeled data are required to learn a specific task in a supervised manner. Preparing such datasets are innately hard as it takes a long time to collect and then label them manually. And sometimes constructing them are nearly impossible for some domains.

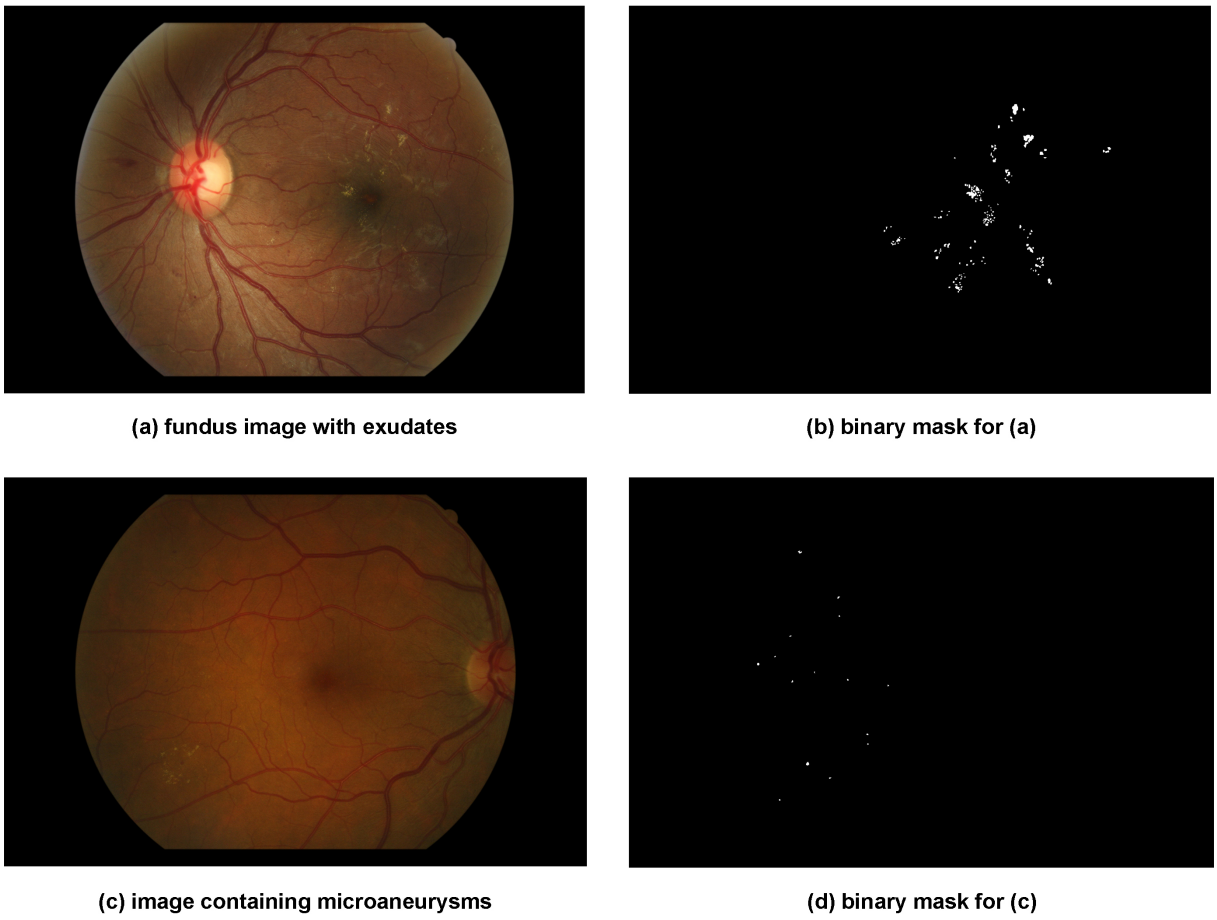
Transfer learning is the proposed solution to overcome this issue and facilitate the knowledge sharing process among different tasks. It suggests reusing parts of the model which have been already trained on a similar task as the foundation for the new task at hand. This approach has proved to be extremely helpful in cases where no good or large enough dataset is available for the target domain, but a fairly good one exists for the source domain. In addition, it saves a lot of time and computational power as the pre-trained weights are employed and the model only needs to learn the last few layers and barely fine-tune the other ones if necessary.

7.2 Methodology

7.2.1 Dataset

Diabetic retinopathy causes different types of deficiencies in the patients' eyes such as exudates, hemorrhages, aneurysms, cotton wool spots, and abnormal growth of blood vessels to name a few. There are many publicly available datasets that could be found online, some small and some really huge. Often, big datasets are used for learning complex tasks to avoid overfitting and ensure the reproducibility of the research result [221]. However, we needed a dataset that could offer masks for the type of defect which it corresponds to. We came across the e-ophtha [49] which had two separate masked datasets; one for exudates and one for aneurysms; and both were manually annotated by ophthalmology experts. E-ophtha EX contains 47 images with exudates and 35 images with no lesion, while e-ophtha MA provides 148 images with microaneurysms or small hemorrhages and 233 images with no lesion.

We only made use of the images with lesions present in them, which altogether summed up to a total of 195 fundus pictures, all having another black and white image as their mask. We shuffled and splitted them in to train, validation, and test sets with 155, 20, 20 images in each of them respectively.



(a) fundus image with exudates

(b) binary mask for (a)

(c) image containing microaneurysms

(d) binary mask for (c)

Figure 7.1: (a) an image from e-ophtha EX containing exudates. (b) binary mask showing the position of exudates in image (a). (c) an image from e-ophtha MA containing microaneurysms. (d) binary mask showing the location of microaneurysms in picture (c).

7.2.2 Preprocessing

The images in the datasets were collected from different clinics with different fundus photography facilities and this had resulted in having varying lighting and pixel intensity values in them, collectively creating unimportant differences among pictures that would have been misleading for the CNN model if left unaltered. Hence, a preprocessing phase was required to counterbalance this issue.

First, we employed OpenCV [31] to crop the images and trim the extra blank space from them. Next, to make the dataset even more homogeneous, we morphed the eyes into perfect circles and removed the extra margins once more. The colors needed to be normalized and enhanced as well, so we performed a weighted sum, and for each image, we applied Gaussian blur (sigma X = 20) on it and added it to its original version. We assigned weights of 4 and -4 to the original and blurred images respectively. The gamma was also set to 128. Finally, the images were resized to 1024x1024 pixels.

All of the above operations were concurrently applied to the masks as well, to preserve the exact scale and position that they signify in the image. But, the masks, especially the ones which corresponded to microaneurysms were extremely small and only consisted of a handful of pixels (between 1 to 5) as opposed to the total image dimension of 1024x1024. This would have been a tremendously hard task for the model to learn. To tackle this issue, the masks were dilated 2 times with a kernel size of 5x5 to make them big enough to be identifiable by the network.

Also, all the instances of a certain defect were shown in one single mask in the dataset. To make them usable, we needed to create a separate binary mask for each instance present in the image. Thus, first, we found all the contours in a given mask. Then, we detached the instances and constructed an exclusive binary mask for each of them. Finally, class ids were assigned to the masks to indicate which defect each of them represents.

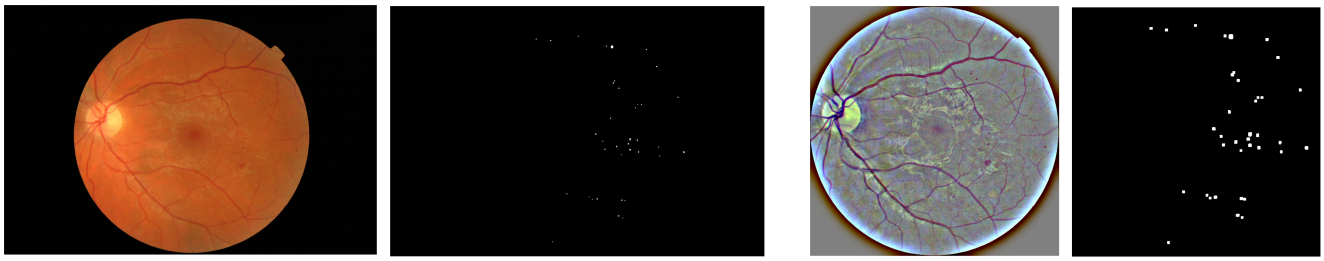


Figure 7.2: (a) an example of the original image from e-ophtha MA and its mask. (b) resulting image and its mask after the preprocessing phase.

7.2.3 Training and Implementation Details

We started with the original implementation of Mask R-CNN for Keras which was made publicly available by its authors. By default, the model's hyperparameters were configured to find the medium to large objects in the image. However, as mentioned before, even after dilating the masks, they only consisted of a few pixels and were really small relative to the complete picture. Hence, we had to make alterations to the model to make it suitable for our task at hand. RPN anchor sizes had to be decreased to enable the model to find deficits as small as 8 pixels in size. Since lesions were small and could be found anywhere in the image, the number of anchors to be trained were increased from 256 to 512, the number of ROIs per image was raised to 512, and the maximum number of final detections was set to 256. Also, we needed to reduce the minimum confidence and threshold required for the model to accept a detection. We disabled the mini-mask feature to avoid any mask resize as we had enough memory and didn't have to sacrifice accuracy for the memory load. Also, we defined the number of classes to be three; one for the background, and two more for exudates and microaneurysms. Adam optimizer was found to be more effective than the default stochastic gradient descent as well. So, it was employed instead to help the model converge to the optimal point faster.

We found out that Mask R-CNN can easily overfit the training set if used naively. As for our first way out, We made use of data augmentation. It comprised random vertical and horizontal flips, 90 degrees clockwise and counterclockwise rotations, and translations and scalings along x and y axes; all of which been applied to the input training images on the fly with the help of the CPU, in parallel to the main training which was being done on our Nvidia 2080Ti GPU to accelerate the process even more.

Our dataset was small, and thus, because of the aforementioned reasons, the best way to counteract this was to employ transfer learning. We put the pre-trained weights of ResNet101 [98] which were originally been trained on Microsoft COCO dataset [148] into service. The model was then trained for 65 epochs with the learning rates of 0.0001, 0.00001, and 0.000001 for two 25, and one 15 epochs respectively to fine-tune all the initial weights and make them suitable for our new task. The process in total took about 15 hours to finish.

7.3 Experiments and Results

It is conventional to evaluate and measure the performance of segmentation models with IoU and mAP. Intersection over Union (IoU) calculates the area of the overlap that happens between the predicted bounding box and the actual mask and then divides it to the area of the union of those two. A completely correct bounding box will result in IoU of 1. A threshold is also set to accept IoUs above it as correct predictions. The percentage of correct predictions out of all predicted bounding boxes is called precision. Recall, on the other hand, is the percentage of correct predictions out of all objects present in the image. As more and more predictions are made the precision will decrease due to false positives, but the recall will increase. These two are calculated for different thresholds and then averaged to find out the AP (average precision) for a given image. The mean of APs across all the images in the dataset is referred to as mAP (mean average precision).

To calculate the mAP for our model, we first needed to fix an issue. Our model was deliberately trained on both tasks simultaneously; giving it the ability to find both types of lesions in a given fundus image at the same time. However, the masks that we had from the datasets were only associated with one type of lesion and for the most part, there was no overlap between the two datasets. If used this way, it would have caused our model to get a lower mAP. The reason behind it was that in the test phase, given a fundus image, our model would have predicted and masked both types of lesions, but the mask that it was being compared to was only showing one type, altogether making the evaluation agent think that the lesions from the other type are all false positives and hence reduce the precision.

To fix this, when the predictions were made for a given image by our model, we passed them through a filter to only keep the ones that are associated with the type that is marked in the corresponding mask image. This enabled us to test our model in a fair way and see how accurate the exudates and microaneurysms are being predicted individually.

Due to the innate complexity of the task and the extremely small size of the lesions that had to be found, we had previously decreased our model's minimum prediction confidence hyperparameter to 35. Hence, we used it along with two more standard thresholds that are usually used in the task of instance segmentation. As a result, 35, 50, and 75 were chosen as our three thresholds to calculate the results with. Results of the evaluation can be found in Table 7.1.

Table 7.1: mAP for train, validation, and test sets created from e-ophtha EX and e-ophtha MA datasets

	mAP_{35}	mAP_{50}	mAP_{75}
Train	0.5408	0.5217	0.3032
Validation	0.5113	0.4780	0.2563
Test	0.4562	0.4370	0.2071

To the best of our knowledge, our work is the first to employ instance segmentation models to identify and mask the lesions in the eye and help to diagnose the infamous diabetic retinopathy. Given the complexity of the task, our model performed extremely well and we call our result a success.

7.4 Conclusion and Final Remarks

We have presented a simple, yet efficient approach to detect, locate, and generate segmentation masks for exudates and microaneurysms which are two types of lesions that diabetic retinopathy causes in eyes. Unlike most of the previous work, our model is capable of automatically extracting useful features related to the scale of our work, and learn to perform the task in an end-to-end manner. Moreover, due to its fast predictions, it has the potential to be easily incorporated into health care facilities. In the next chapter, we will employ our model as a feature extractor to identify the severity of DR in patients' eyes.

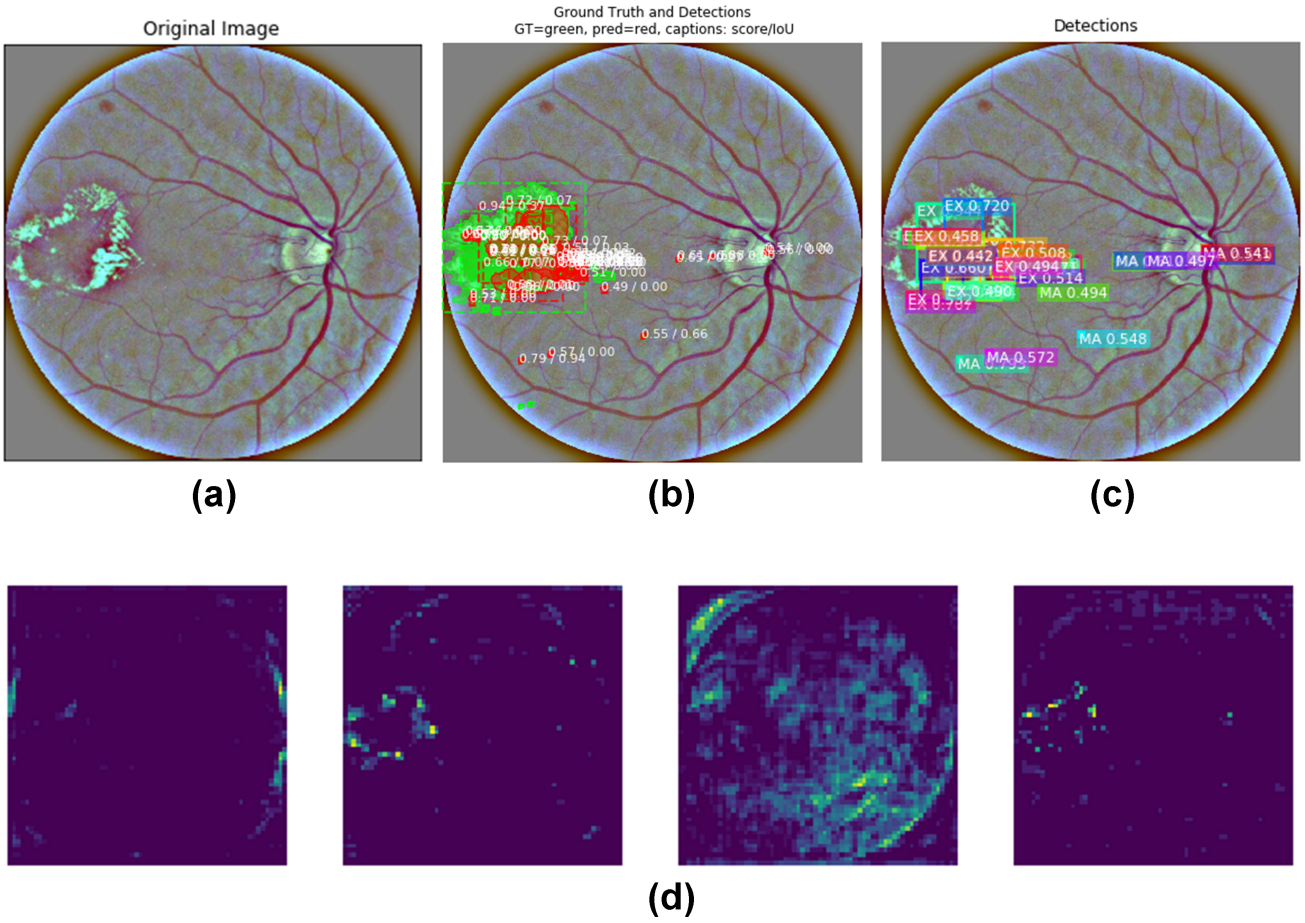


Figure 7.3: (a) original image (b) predicted masks, bounding boxes, their score and IoU (c) Types of the lesions detected and their scores. (d) sample activations of a few layers of the model

Chapter 8

DRDr II: Detecting the Severity Level of Diabetic Retinopathy Using Mask RCNN and Transfer Learning [1]

[1] F Shenavarmasouleh, FG Mohammadi, MH Amini, and HR Arabnia. 2020. International Conference on Computational Science and Computational Intelligence (CSCI). 788-792.

Reprinted here with permission of the publisher.

Motivation: Diabetic is among the most infamous diseases in the world and it affects millions of people every year. It can cause problems in several organs of the body, one of which is the eyes. When it does, it can cause vision impairment and if not treated professionally it can eventually lead to vision loss. Diabetic Retinopathy (DR), is the name that the experts have given to this subsection of the disease. Patients need to undergo a simple screening to get their fundus images taken by the retinal photography devices in clinics. Experienced clinicians will then need to analyze the pictures and decide on the presence and the significance level of the case by carefully examining countless features in them. Those features happen to be extremely small in many cases (e.g. less than 5 pixels) and can be very difficult and time-consuming to find even for the best-trained eyes. But, the good news is once detected soon enough, laser surgeries can be used as the treatment. Hence, the sooner DR is diagnosed, the more chance the patient has to recover from it and get the most out of his/her eyes.

As the number of diabetic patients is growing annually and each patient needs frequent screenings, more and more images need to be analyzed by experts every day and since us, humans, are prone to mistakes in repetitive tasks, it increases the risk of not diagnosing DR in time that can be crucial to patients health. Besides, research has shown that out of all the screenings conducted annually, only 25.2% show trays of DR in them and the patient has to be referred to ophthalmologists [160].

Looking at this from a wider perspective, we can see that the whole process is being taken care of sub-optimally on both sides. Doctors spend nearly three-fourths of their time analyzing healthy patients and patients with mild cases of DR are always at the risk of being left undiagnosed. Fortunately, machine learning and deep learning can come to the rescue and help in many areas with their speed and accuracy. Biomedical imaging is not an exception and has always been an area of interest for researchers in these fields.

As mentioned earlier, doctors need to analyze fundus images and decide on the presence and the significance level of DR. In our preceding paper [222], we proposed DRDr, a deep learning model derived from a complex Convolutional Neural Network, namely Mask RCNN, that could take a fundus image as the input, and output all the instances of microaneurysms and exudates - two types of lesions that could appear in the eyes of DR patients - present in it, their position in addition to their exact shapes as separate black and white binary masks, and a confidence score for each of them, all in near real-time. Hence, taking care of the former problem and helping doctors in coming to a conclusion about the presence of the DR.

Contribution: In this chapter, we propose DRDr II, a hybrid of machine learning and deep learning approaches that uses DRDr as a feature extractor in the core of its pipeline, and with that, it becomes able to tackle the latter problem and classify patients into three severity groups. An overview of the system can be found in Figure 8.1.

Organization: The remainder of the paper is organized as follows: First, we start with the related works. Then, we fully explain our several preprocessing steps and feature extraction phase. Next, we show the experiments that we conducted and illustrate the results. Finally, we conclude with the discussion and future work.

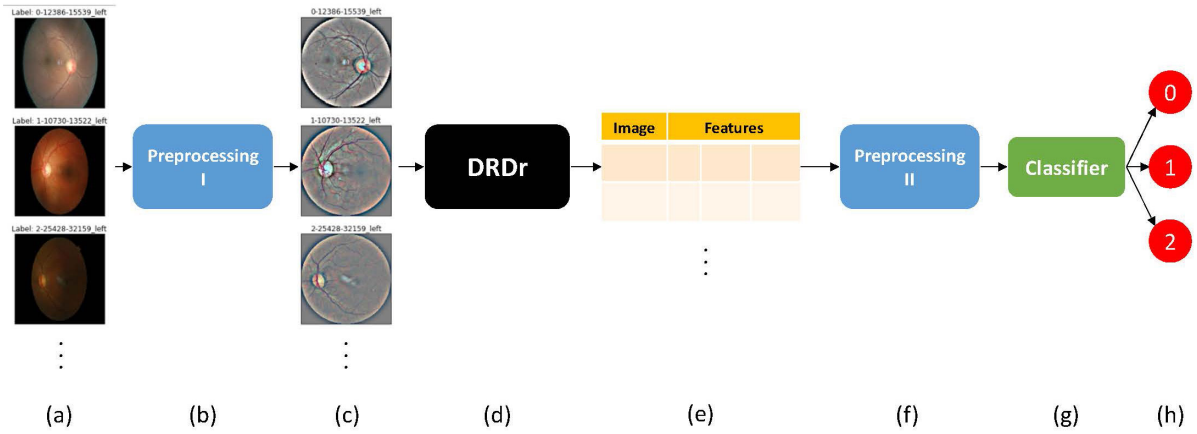


Figure 8.1: (a) original images in the dataset. (b) perform all the Preprocessing I steps explained in section 3.2. (c) preprocessed images. (d,e) pump all the preprocessed images into the DRDr model and use the output generated by it for all images to create a data frame for features. The steps are explained in section 3.3. (f) apply the Preprocessing II steps explained in section 3.4. (g,h) feed the data frame to an arbitrary classifier and get the severity levels as explained in section 4.

8.1 Related Work

An extensive amount of research has been done to find the severity of diabetic retinopathy in patients and classify images into several categories. The majority of them use pure image processing and computer graphics techniques such as Adaptive Histogram Equalization, Discrete Wavelet Transform, Matched Filter Response, Fuzzy C-Means Segmentation, Region Growing, Thresholding [57], [150] along with

other Morphological Processings as feature extractors and pass the result to machine learning classifiers. More than often, authors have decided to go with an ensemble of classifiers in order to make their model more robust and get a better result. Meta-learning has also shown promising results and researchers could benefit from its simplicity and speed as it tries to learn using the least amount of features [172], [173].

Traditionally, the problem definition revolved around binary classification, and models were trained to label patients as healthy vs unhealthy. To that end, some merged image features with the contextual data that were available for the patients [49] and a few used Decision Trees, Naïve Bayes, Random Forests, Adaboost, Probabilistic Neural Networks (PNN), K Nearest Neighbors (KNN), and Support Vector Machines (SVM) as ensembles to tackle this issue [25] [11] [198] [234].

Gardner et al. [72] were one of the firsts who used feed-forward neural networks as the classifier. They converted the fundus images into grayscale and created small tiles out of the entire picture for each of them. The 20x20 tiles were then converted to the vectors and were fed into the neural network to find the deficiency class. Usher et al. [250] also employed neural networks and fed candidate lesions, their position, and their type as the inputs of the Neural Network. DRDr II is in line with this work in terms of the employed features.

Releasing of better datasets that provided more severity classes was good news for researchers as they started to level up their models. Same as before, traditional machine learning models were the default option here and SVM, KNN, Gaussian Mixture Model (GMM), and Adaboost were used to classify fundus images into three [134], four [205], and five [2] groups. But, as of the past few years, with the rise of deep learning models, researchers could get promising results solely by using image features extracted via Convolutional Neural Networks (CNN) and group patients into two [73], and five [197] classes.

Inevitably, pure CNN models have the advantage of taking an image as a whole and use it completely as their available features. They try to figure out which parts of the image are best to use and what to look for in images for the sole purpose of getting a better accuracy in the classification task (severity level). So, in theory, the model can converge to a point that it can cognitively understand all the types of lesions and deficiencies, such as cotton wools spots, abnormal growth of blood vessels, hemorrhages, and much more in addition to the two (exudates and microaneurysms) that we were bound to use due to lack of proper databases. With all that said, we will show that with proper techniques, two is all it takes to yield an acceptable 92.55% classification accuracy.

One big drawback of all the aforementioned related works is that they each used different datasets for their work. Some were public datasets but some were private ones created only for their labs. Also, the size of the datasets varied by a big margin, as they could contain as few images as 39 in total (extremely prone to overfitting), all the way up to the order of a few thousand. As a result, unfortunately, it is not possible to compare the previous work and their achieved accuracy with each other objectively.

8.2 Methodology

8.2.1 Dataset

Unlike DRDr, DRDR II did not enforce any mask requirement as a limitation for our dataset selection. And since we intended to use our pre-trained DRDr model as our primary feature extractor to generate the masks ourselves, we had the freedom to choose from all the huge datasets available out there for diabetic retinopathy that did not provide masks as a part of their data. To that end, we selected a public Kaggle dataset [201] that has more than 35 thousands fundus images with the size of 1024x1024. It was large enough for our task to ensure that overfitting would not occur and that our result could be reproducible [221]. The dataset architecture was simple, it only provided the original fundus images, the eye that the picture was taken from (left vs right) along with a single integer indicating the severity of the case.

8.2.2 Preprocessing I

This new dataset, also, suffered from the same issues that were present in e-ophta [49]; the dataset which we had used for training DRDr. The images had been collected from different devices located in various photography facilities around the world and as a result, they had different pixel intensity values, lightings, and zoom levels. To counteract this issue, we employed the same preprocessing steps that we had used in DRDr as the starting point for our new preprocessing pipeline. For short, we employed OpenCV [31] to crop the images and remove the extra blank space around them. Then, we transformed the eyes into perfect circles and removed the extra margins once more. To normalize the contrast level, we applied Gaussian blur (sigma X = 20) on each image and added the output to the original version. We assigned weights of 4 and -4 to the original and blurred images respectively. The gamma was also set to 128.

Table 8.1: Hyper-parameters for Mask RCNN used for DRDr

MASK RCNN	
Attribute	Value
BACKBONE	resnet101
BACKBONE_STRIDES	[4, 8, 16, 32, 64]
DETECTION_MAX_INSTANCES	256
DETECTION_MIN_CONFIDENCE	0.35
IMAGE_MAX_DIM	1024
IMAGE_RESIZE_MODE	square
LEARNING_MOMENTUM	0.9
LEARNING_RATE (Value, (#Epochs))	10^{-4} (25), 10^{-5} (25), 10^{-6} (15)
MASK_SHAPE	[28, 28]
MAX_GT_INSTANCES	100
NUM_CLASSES	3 (BG, EX, MA)
RPN_ANCHOR_RATIOS	[0.5, 1, 2]
RPN_ANCHOR_SCALES	(8, 16, 32, 64, 128)
RPN_ANCHOR_STRIDE	1
RPN_TRAIN_ANCHORS_PER_IMAGE	512
STEPS_PER_EPOCH	500
TOP_DOWN_PYRAMID_SIZE	256
TRAIN_BN	FALSE
TRAIN_ROIS_PER_IMAGE	512
USE_MINI_MASK	FALSE

8.2.3 Data Frame Creation

After preparing the images it was time to bring our DRDr model into play. We decided to use our pre-trained model as the main feature extractor for the task at hand. The original DRDr was a Mask RCNN model that we had trained on the e-optha EX and e-optha MA datasets that contained images for exudates and microaneurysms or small hemorrhages respectively. However, we ended up having to change many hyper-parameters of the original implementation to make it usable for our task. An overview of the updated model hyper-parameters can be found in table 8.1.

Hence we geared up the model and initialized it with our pre-trained weights. Then we pumped all the images in the new dataset into it to get the masks for each instance of microaneurysm and exudate

along with their confidence scores and bounding boxes. We used this info to conduct a data frame with the following details about each instance of the lesions found in an image: name of the file as the id, the eye that the image was taken from, the bounding box, its center position, the area of the masked lesion, type of the lesion, severity, and its confidence score. That left us with about 445 thousand instances collectively found for a total of 35 thousand images.

It is also worth mentioning that DRDr normally takes a fundus image and the original mask as the input. But, at the test/production phase, the goal of the latter would only be to help the model illustrate the differences between the predicted mask and the original one. Since this fancy feature is not needed for our new task at hand, we used a fixed black binary mask and passed it alongside all of the images in our dataset.

8.2.4 Preprocessing II

In our preceding paper [222], we have elaborated on why we had to tweak the confidence score to get the most out of our model. DRDr had originally been trained to output all the instances that it is more than 35% confident about. It worked well for our previous task, however, in DRDr II, we realized that it is not good enough and it can add unwanted noise to the dataset. This was mainly an issue for the exudate instances since their areas were typically several folds larger than microaneurysms. Hence, as the first step towards putting the data frame created in the previous phase into use, we pruned all the exudate instances with a confidence score of below 65%. Next, we converted the eye column (left vs right) into categorical entities and then calculated weighted areas (confidence score multiplied by area) to counteract the big areas with low probabilities.

Originally, the images in the dataset were separated into five categories. But, based on the case study conducted by Google [35], the are so many overlaps in the adjacent categories in terms of severity that even experts were not sure which category to pick as the final choice. Hence, we grouped the top two most and bottom two most groups and reduced the number of classes to three to make up for it.

Next, to bundle all the various instances of lesions found in one image together to form one single row in the data frame for each image found in the dataset, we grouped all the rows based on their name and then calculated the following attributes for each of them: number of instances per lesion type (EX and MA), the sum of weighted area for each type, and mean and standard deviation for the center of

bounding boxes for each lesion type. To further improve our model and make the system robust to the outliers we then proceeded and calculated the z-score for the aforementioned columns and dropped the instances that were not within two standard deviations.

We started using the data frame at that point, but then we realized that the dataset was not well balanced as the majority of images did not have any issue (the patient was healthy) and that caused the model to get a very good accuracy just by outputting 0 all the time without learning any useful relationship among the features. Hence, we decided to under-sample the 0 class to make the final dataset well balanced.

We then normalized all the attributes in order to make them be in the range of 0 to 1 that is proven to help the model to converge faster. At this point, the data frame was ready and each row in it corresponded to an image in the original dataset. We then shuffled the whole thing and since the data frame was large enough, we decided to use 80% of the rows for the training phase and the remainder for validation and testing phases equally.

8.3 Experiments and Results

As for our main classifier, we used a feed-forward neural network with 2 hidden states each with 75 nodes followed by a softmax layer that outputted the corresponding severity level. We used Adam optimizer with a learning rate of 0.001 and trained the model for 100 epochs. To compare it with other classifiers and get a sense of where we stand among other machine learning approaches mentioned in the related works, we also trained Linear SVM, Kernel SVM (polynomial and RBF), Logistic Regression, Decision Tree, AdaBoost, Naïve Bayes, and KNN. You can find our results in table 8.2. Unlike most of the related work, we did not ensemble any of our classifiers together. However, for the most part, the margins of differences were significantly narrow and consequently, it is unlikely that ensembling could lead to any significant positive difference.

We also performed a brief ablation study over our data frame to find out the amount of influence each feature has over the final accuracy score. According to our findings, all columns played a positive role and had a positive correlation with the final result, however, the most dominant one proved to be the sum of weighted areas for each type of the lesions.

Table 8.2: DRDr II results for different classifiers

Classifier	Accuracy (%)
Neural Network	92.55
Decision Tree	91.17
Naïve Bayes	85.25
SVM - Polynomial	79.10
Logistic Regression	78.99
SVM - Linear	78.65
SVM - RBF	77.20
KNN	75.86
AdaBoost	60.00

This was also a great test for our DRDr model since the performance of segmentation models are often measured with Intersection over Union (IoU) and mean Average Precision (mAP), and although they are both great measures, they are not as intuitive as the traditional accuracy score. It can be perceived that DRDr proved itself once again and showed that in addition to being able to find lesions in the fundus images, it can also be used as a solid and reliable feature extractor to help find the severity of infamous diabetic retinopathy.

8.4 Conclusion and Final Remarks

In this chapter, we presented DRDr II, a proceeding work built on top of DRDr, to aid doctors and clinicians in the diagnosis of Diabetic Retinopathy and help them identify the severity of the cases. In the next chapter, we plan to unify DRDr and DRDr II and morph them into a single monolithic deep learning model so that it can produce binary masks for the lesions and find the severity level at the same time.

Chapter 9

DRDrV3: Complete Lesion Detection in Fundus Images Using Mask R-CNN, Transfer Learning, and LSTM [1]

[1] F Shenavarmasouleh, FG Mohammadi, MH Amini, T Taha, K Rasheed, and HR Arabnia. Accepted by 2021 International Conference on Health Informatics and Medical Systems (HIMS).

One of the infamous causes for eye damage is Diabetic Retinopathy (DR) which usually starts as vision impairment and ends up causing vision loss if not cured on time [37]. Diabetic retinopathy causes different types of vision problems in the patients' eyes, such as cotton wool spots, exudates, microaneurysms, hemorrhages, and abnormal growth of blood vessels. In this study, we aim to focus on two of them: exudates and microaneurysms. Figure 9.1 presents a sample fundus picture including exudates and microaneurysms in the eye.

Detecting eye damages on eyes as early as possible helps specialists to have the chance to stop further injury and take advantage of laser surgeries to cure it. As the annual number of such patients is proliferated significantly and each patient needs frequent screenings, the process of analyzing the screened images is getting more important day by day. Each of the captured images requires to be technically analyzed by doctors. However, it has some limitations like time and cost. Doctors have to go through a hard and time-consuming process as the deficiencies are not always visible clearly, due to their inherent tiny sizes and unstructured shapes; hence requiring detailed assessment. Thus, the doctors have to look for an infinite number of features within each screened image. To address this inefficiency, a large number of research studies and Kaggle challenges are posted globally to make this process more automatic. Traditional accuracy, Intersection Over Union (IOU) and mean Average Precision (mAP) are examples of sample criteria that are considered in the research studies.

Machine learning, especially deep learning plays a pivotal role in eye damage detection. Needless to say, to apply machine learning, we need to have proper datasets and leverage feature extractors, especially the ones which are compatible with the task at hand. In other words, we need to apply feature extractors which can extract important features that help machine learning algorithms to forge a model and eventually learn to predict properly. Additionally, most feature extractors are problem/task-oriented which means we cannot use an extractor trained on a certain domain for another. For instance, a feature extractor developed for detecting skin cancer [238] is not useful for detecting any other diseases.

9.0.1 Goals

Following our work in DRDr [222] and DRDrII [223], our main goal here is to combine the previous networks and use the new model crafted by unifying them to 1. detect everything about lesions via a single model: lesions masks, bounding boxes, lesion types, along with the overall severity of the instances

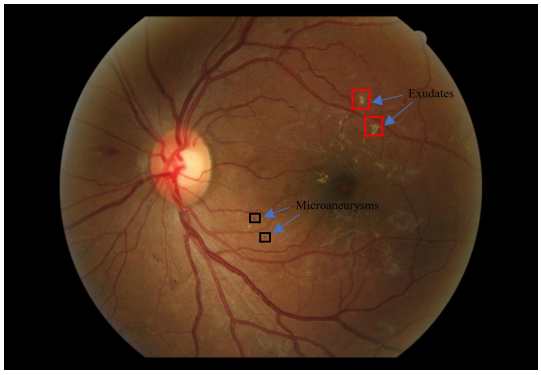


Figure 9.1: The lesions caused by diabetic retinopathy, namely exudates (red rectangles) and microaneurysms (black rectangles) in the eye

per image. 2. increase the accuracy of the severity classification that we achieved in DRDrII. We aspire to combine technically different image segmentation and machine learning algorithms, such as Mask R-CNN [97], LSTM [105], and transfer learning, to detect lesions' attributes. We extend and evolve our Mask R-CNN model with LSTM that enables the artifact to learn more and hence tackle more tasks simultaneously.

9.0.2 Challenges

The first problem, we face in this research study is the nature of fundus pictures. They are screened and captured in different zoom levels, lighting conditions by different laboratory cameras. The images have different pixel intensity values and are collected from separate clinics with a variety of fundus photography facilities. Having these heterogeneous types of pictures, in general, negatively affect machine learning algorithms, like CNNs, and prevent the algorithm from learning and generating proper and abstract features [60]. Hence, we need to deal with this challenge before proceeding with machine learning algorithms.

The rest of this chapter is organized as follows. We first discuss the related work including state of the art work including history of CNN and the latest improvement in the field. Next, we address the proposed method and problem description stating the application of transfer learning and LSTM followed by evaluation and results.

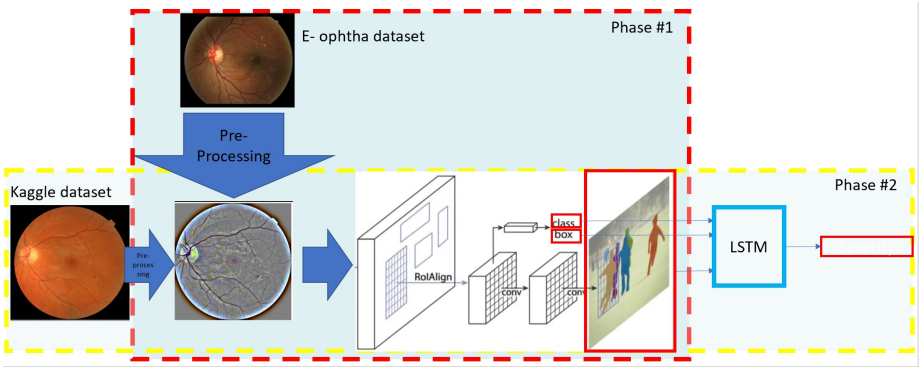


Figure 9.2: Proposed method in two phases

9.1 Related Works

Sopharak *et al* [234] used a few conventional classifiers, such as Naive Bayes, Support Vector Machine (SVM), and K Nearest Neighbors (KNN) to detect one of the lesion types like exudates in the spatial level. Authors took advantage of popular methods for segmentation such as region growing [135] [169], and thresholding [187]. The researchers in [234] extracted 15 features and assessed them. Additionally, the dataset they used was small with a few (39) images.

Chen *et al* [37] proposed a method, so-called LFPN, to detect tiny lesion cases in diabetic retinopathy images. The LFPN architecture has two important features; first, one can use large CNN feature maps, which are an identical size to the DR image, including details of small lesion features. So, LFPN is an efficient algorithm for object classification in the second stage of RCNN. Second, one can use the top layer for region proposal, computing resources more efficiently.

Sahoo *et al.* [210] established a new algorithm for Retinal Fluid Automatic Detection (RFAD) by extracting the affected retinal segment automatically and claimed to have obtained promising results. Furthermore, researchers [156] used hybrid color, texture features, and customized CNNs (MCNN) for a multi-class lesion classification system.

Moreover, researchers have used color to detect lesions [273]. In [273] researchers applied machine learning algorithms to identify microaneurysms (MAs) in fundus images using e-ophtha dataset. Among all ML classifiers, they used decision trees, Support Vector Machine (SVM) and Logistic Regression (LR),

k-nearest neighbors (k-NN), Random Forest (RF), and Naive Bayes (NB) classifiers to distinguish MAs from healthy fundus images.

Researchers have used different algorithms of CNNs and schemes to tackle this problem [273] [97]. Several research studies are accomplished to extend the level of classification and drag the level of severity to this task too. For example, researchers in [222] presented a new method called DRDr to generate segmentation masks for two popular types of eye lesions: exudates and microaneurysms. Additionally, Shenavarmasouleh *et al* in [223] established DRDr2 to extend the lesion detection process by generating instance masks and classifying the fundus images with respect to the severity cases. DRDr2 seeks for lesions severity manually which is considered as its main drawback. Unlike most of the previous work, in this work, we present DRDrV3 in which we automatically extract useful and important features and take advantage of LSTM to learn to perform the task of classification entirely.

9.2 Proposed Method and Problem Description

Researchers [97] in 2017 established a method for object instance segmentation, namely Mask R-CNN. This method detects object instances in a given input picture efficiently and creates precise segmentation masks for each of them. We find that Mask R-CNN can be generalized to other tasks with careful fine-tuning.

Mask R-CNN returns the class, bounding box, and mask information. In state-of-the-art works, researchers use Mask R-CNN for different purposes. In this study, we aim to extend and customize this architecture to make it yield the severity of the damage caused by diabetic retinopathy to the eye in addition to the previously mentioned info. Following our work in [14] we hypothesize that this can be achieved by designing a new pipeline and expand the underlying main model. The new model will also have the benefit of being able to be trained in an end-to-end manner.

In this section, we aim to propose an optimal solution to take advantage of a combination of image segmentation and machine learning algorithms. Scientists in research studies like [124] leveraged Mask R-CNN to do image segmentation using masked dataset and get required mask information. In this study, we propose two main phases. Figure 9.2 presents two combined phases. In the first phase, we train Mask R-CNN to learn based on the masked images. Next, in the second phase, we use the result of the first phase

to carefully predict the severity of damages in each case. In general, we mainly focus on Mask R-CNN to obtain these results: class/damage type (MA vs EX), bounding box, and mask. Then, we redirect all these into an LSTM model to predict the severity of the damage as well. We discuss and elaborate details of each phase clearly in the following subsections.

In this study, we use two publicly available datasets: The first dataset we use is E-ophtha [49] offering separate masked datasets consist of exudates and aneurysms in which the images of each dataset were manually labeled and classified by ophthalmology experts. The second dataset is a public Kaggle dataset [201] that includes 35 thousand fundus images taken from either left or right eye, together with a single numeric value stating the severity of the instance. We use the first dataset for training masks, bounding boxes, and labels. Then we include the severity of images from the second dataset and perform the procedures explained in the second phase.

9.2.1 Pre-processing

To address the challenges discussed above, we need to apply pre-processing algorithms and convert all pictures into a homogeneous dataset. Additionally, we morph the pictures to make all the eyes in the fundus pictures look like a circle and get rid of any unwanted noises. The sizes and contrasts of eyes in the picture should never be considered a positive or negative feature and contribute on objects detection since our goal is to only look for some tiny objects in the images. Hence, we normalize these two by resizing the images to 1024x1024 pixels and also perform the same contrast normalization technique used in DRDr [222]. By doing these steps, we ensure that the feature extractors like CNN can properly begin to learn the features in our model without having to struggle with additional complexities.

Furthermore, since microaneurysms were extremely tiny and only consisted of a handful of pixels (between 1 to 5), we dilate the corresponding masks 2 times with a kernel size of 5x5 to make sure that masks are big enough to be noticeable by the artificial network.

Since all the instances of a certain defect were shown in one single mask in the first dataset (E-ophtha), to make them usable for our algorithm, we need to generate a separate binary mask for each instance shown in the mask image. To that end, we detect and detach the instances and construct an exclusive binary mask for all of them. Let's consider if one image has M instances, we generate M separate binary masks where the ones illustrate the pixels which correspond to the defect and the zero pixels show the

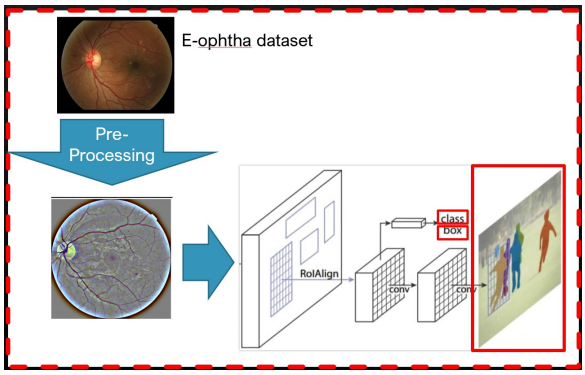


Figure 9.3: The architecture of phase one

background. Additionally, we assign class ids to the binary masks to indicate which defect each of them depicts.

9.2.2 Phase One

In the first phase, we aim to use E-ophtha dataset to train and evaluate Mask R-CNN performance.

Training Mask R-CNN

In order to make Mask R-CNN work best on the dataset, we need to configure it by updating its main hyperparameters like region proposal network (RPN) sizes. For example, we decrease the size of RPN and replace the size number with as small as eight pixels. More details can be found in [222]. Furthermore, training Mask R-CNN from scratch requires a huge dataset and is time-hungry. Instead, we take advantage of transfer learning first introduced by Pan and Yang [186], which is one of the most interesting features of deep neural networks, to address this problem with a much smaller annotated dataset.

Applying transfer learning

Due to the lack of training samples during the training phase (in phase one) which may lead out model to overfit, we employ transfer learning to address the aforementioned problem. The already existing edges in the model graph can be initialized via the weights that are publicly available via the authors of the COCO dataset [148]. Then we train the remaining edges (mostly edges in the last layer) while fine-tuning the

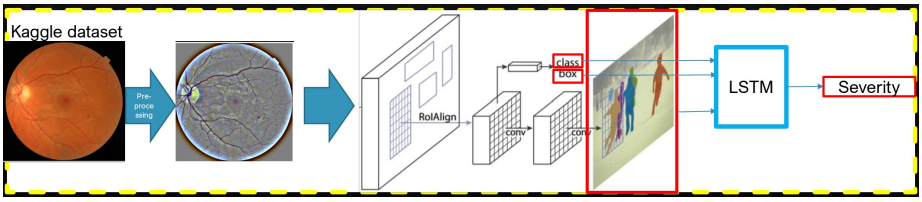


Figure 9.4: The architecture of phase two

previous ones if necessary. After training, our Mask R-CNN model can generate bounding boxes, masks, and assign deficit labels (EX vs MA). The performance of the model trained in this phase can be found in the Evaluation and Results section.

9.2.3 Phase Two

In this step, we take advantage of the trained Mask R-CNN in the first step, build on it to enable the model to detect the severity of damage as well, and finally evaluate this extension on a new dataset borrowed from Kaggle [201].

Applying Mask R-CNN

After executing pre-processing step, we plug in the fundus images from Kaggle dataset into the Mask R-CNN to generate the same information (bounding boxes, classes, masks for each image) we have in phase one. Furthermore, we want to add one more feature, severity, to the phase two to plug in LSTM [105].

LSTM

We aim to take advantage of LSTM [105] due to various reasons. LSTM does not require tuning main parameters due to the large number of its parameters such as learning rates, input, and output biases. More importantly, LSTM can handle dynamic input sizes and since we will have a different number of deficit instances generated for each image in our task, it comes to the rescue. Furthermore, LSTM helps us monitor and analyze the possible relation between sliding windows to manage the lesions associated with the locations in the image.

In order to enable LSTM to yield the severity of each image, we need to prepare its input layer properly to combine layers of information that we have obtained from phase one. Figure 9.5 presents how input layers are connected to each other. *input_13: Input_layer* is the concatenation of bounding boxes and class types that is being added to *permute_7: Permute* that is the mask layers that have been fed to 3 phases of CNN in order to extract any possible information from the shape of the masks and then reshaped to match the dimension of the first branch.

9.3 Evaluation and Results

To evaluate our proposed method, we need to determine relevant datasets, together with evaluation criteria. In this section, we discuss both followed by the results.

9.3.1 Datasets

In this study, we use two publicly available datasets: The first dataset we use is e-ophtha [49] offering separate masked datasets consist of exudates and aneurysms where each dataset was manually labeled and classified by ophthalmology experts. E-ophtha EX contains 47 pictures with EXudates and 35 pictures are lesion-free, while e-ophtha MA provides 148 pictures with microaneurysms or small hemorrhages and 233 pictures are lesion-free.

The second dataset is a public Kaggle dataset [201] that includes 35 thousand fundus images taken from either left or right eye, together with a single numeric value stating the severity of the instance. It is large enough for our task to ensure that overfitting would not occur and that our result could be reproducible [221]. It is important to note that this dataset does not come with any kind of mask and hence it could not be directly used in our paper without the help of the first phase.

9.3.2 Evaluation criteria

◇ **Intersection over Union (IOU):** IOU is the fraction of the area of the overlap that happens between the actual mask and the generated (predicted) bounding box, over the area of the union of those two. When IOU=1, it means that we have a completely correct bounding box and mask. In practice, to accept

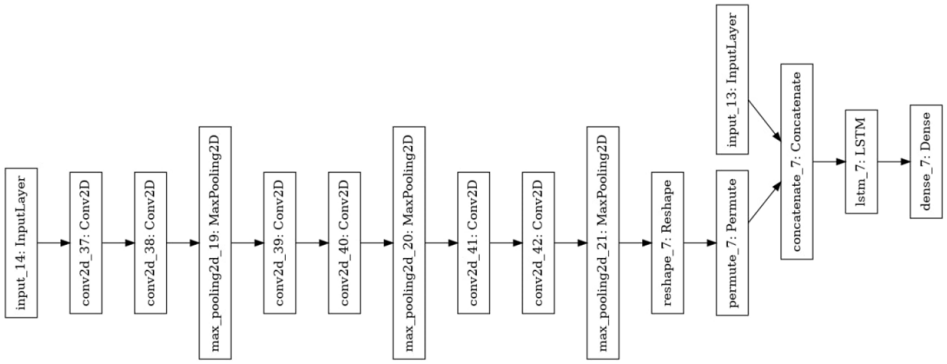


Figure 9.5: The input layers of LSTM

the predictions a threshold is set and if the IOU positions above it then the predictions are marked as correct and incorrect the other way around [118].

◇ **mean Average Precision (mAP):** Let's first define precision (P) which calculates the percentage of correct predictions out of all predicted bounding boxes in the image. Average Precision (AP) presents the performance of the model for each class while detecting objects in a dataset. Finally, Mean average precision is examined as the average or mean of the AP across all of the images in a dataset [27].

◇ **Accuracy:** calculates the percentage of correct predictions out of all predictions. The accuracy formula is defined in Eq. 1 [213].

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

where TP is the true positive, TN is the true negative, (TP+FN) is the number of total positive predictions, FP is the False Positive, and (TN+FP) is the number of total negative predictions.

9.3.3 Phase One Results

To evaluate the performance of phase one, as stated in [222], we use mAP to compare the results with different thresholds. We use three different thresholds for IOU: 35, 50, and 75. 35 is the minimum value for IOU to mark true predictions as the model is configured previously to yield masks that have a confidence score of more than %35. These thresholds are considered as evaluation criteria to compute mAP for each instance in the images. Table 9.1 presents

Table 9.1: result of phase one with three different thresholds. The numbers (35,50 and 75) are the threshold of Intersection over Union (IOU)

test option	mAP_35	mAP_50	mAP_75
train	0.5408	0.5217	0.3032
Validation	0.5113	0.4780	0.2563
test	0.4562	0.4370	0.2071

9.3.4 Phase Two Results

In this phase, we aimed to predict the overall severity of the case in the image. We have three classes: 0 which indicates healthy, 1 that suggests medium damages, and 2 which denotes severe damages. Table 9.2 presents the accuracy of the test scenarios with their associated confusion matrices. The results show that the test option with classes, bounding boxes, and masks yields the best result in comparison with others and we achieve slightly better performance than our previous result in DRDrII [223].

Table 9.2: result of phase two accuracy and the corresponding confusion matrices

test option	Accuracy(%) test	confusion matrix
Classes, bounding boxes, without normalization	92.67	[2900 8 1] [72 790 20] [17 175 16]
Classes, bounding boxes, with normalization	84.37	[2904 4 1] [407 455 20] [48 145 15]
Classes, bounding boxes, masks	93.47	[2880 24 5] [29 771 82] [7 111 87]

9.3.5 Further Analysis

Following the successful integration of LSTM into our model, we discovered that there exists a specialized padding masking layer in Keras that can be used instead of the naive layer. Up until this point, we were padding the data with zeros and were hoping for the model to learn to ignore those values on its own,

Table 9.3: Accuracy of DRDrV3 after incorporating new masking layer versus benchmarks

Model	Accuracy
Inception V3 [241]	94.70
DenseNet [117]	38.82
EfficientNet [242]	77.03
DRDr V3 - Normal Masking Layer	93.47
DRDr V3 - New Masking Layer	**94.89**
DRDr V3 - New Masking Layer with mask images	92.65

inevitably introducing some noise to the model. However, the specialized layer enabled us to have a unique value for padding and then it internally ignores those values before feeding them to the next layer. The approach in return refined our model even more and the results can be found in table 9.3.

In the process of creating baseline benchmarks, we kept all the parameters the same as our original model. Hence, all the different models were trained and tested with similar subsets of the dataset, and the same preprocessing techniques were applied to the data for all of them.

The results showcase the power of DRDrV3 as it achieves not only a competitive accuracy score versus our baselines, but can also provide much more information than just the malignancy stage which is all that other benchmarks can provide. This includes but is not limited to the masks for two different kinds of lesions, their locations, a corresponding confidence score for each deficiency, bounding boxes, and an attention mask. As a result, DRDrV3 separates itself from baseline and proves itself as an interpretable all-in-one model for Diabetic Retinopathy.

9.4 Conclusion and Final Remarks

In this study, we bundled several machine learning and deep learning approaches and models and created a complex multi-purpose interpretable framework that is capable of aiding doctors to better face Diabetic Retinopathy disease. Given a fundus image, our model can find all the instances of lesions, more specifically exudates and microaneurysms; and generate masks, bounding boxes, lesion types and finally a single severity score in an image. Our framework can be incorporated into medical facilities as a real-time

or post-processing assistant. The architecture itself is dynamic and can be easily expanded with parallel branches for processing new features.

Chapter 10

Conclusion

Machine Learning and Deep Learning have proven to be extremely good at extracting complex patterns and features from the raw data and hence are now being used in a wide range of applications and fields as one or a combination of Computer Vision, Natural Language Processing, and Reinforcement Learning algorithms. They are exceedingly good at scenarios where a huge amount of data exists as they can effectively avoid biases that could possibly be present in the data collection. Hence, the Internet of Things (IoT) and Embodied AI are the fields that are heavily invested in using these models in their workflows and are designing, developing, and testing new pipelines for collection of data, local and/or centralized processing of them, safe transmission, and proper storage of data for future reusability.

One of the major fields that can benefit from this foundation is the healthcare industry as it produces a huge amount of data annually. However, it is more challenging there. In healthcare, the collection of data and labeling them is often very expensive and time-consuming and as a result, not many big and properly labeled datasets exist for Deep Learning researchers. Different clinics use different devices with varying configurations and if the data is collected from varying sources, then adequate preprocessing techniques need to be applied to make the data standard and usable. Also, the decisions about the malignancy of the disease are often very subjective and depend on the doctors, hence even the hardly collected labels need to be further analyzed for possible outliers.

Transfer Learning along with Data Augmentation is the current best solution for tackling the aforementioned issues, as it can help produce high accuracy results even with small datasets, thanks to its ability

to benefit from the knowledge it has previously collected from a similar task and on a different but huge dataset.

Furthermore, Machine Learning and more specifically Deep Learning models are extremely difficult to trace and due to their complexity, understanding the reasoning behind their decisions is challenging. That's why they are often referred to as black boxes. This might be alright in many fields, but in healthcare, there is no room for errors, and understanding the reasoning behind choices or predictions, especially if they are in conflict with the medical team is a must, and failing to do so can lead to biases and incorrect outcomes, which can be disastrous in healthcare.

In this dissertation, we first elaborated on the common practices that can lead to biases and misleading results. Then we explored Embodied AI, its paradigm, and its recent advances. Next, we discussed the Internet of Things and how healthcare can benefit from it, what are its challenges and what possible solutions can be employed to address them. Then, we employed all that knowledge to design, implement, and evaluate a multi-purpose interpretable framework for preprocessing, analyzing, masking, and diagnosing Diabetic Retinopathy in diabetic patients that has the ability to be incorporated into medical facilities as a real-time or post-processing assistant. We used all the currently available datasets for developing our model, but it can benefit greatly from better data down the road. The architecture itself is dynamic and can be easily expanded with parallel branches for processing new features, and we hope that other researchers can contribute to this field by providing better data in the future to make this framework even more robust.

Bibliography

[1] R. Acharya, C. K. Chua, E. Ng, W. Yu, and C. Chee, "Application of higher order spectra for the identification of diabetes retinopathy stages," *Journal of medical systems*, vol. 32, no. 6, pp. 481–488, 2008.

[2] P. Adarsh and D. Jeyakumari, "Multiclass svm-based automated diagnosis of diabetic retinopathy," in *2013 International Conference on Communication and Signal Processing*, IEEE, 2013, pp. 206–210.

[3] F. Ahamed and F. Farid, "Applying internet of things and machine-learning for personalized healthcare: Issues and challenges," in *2018 International Conference on Machine Learning and Data Engineering (iCMLDE)*, IEEE, 2018, pp. 19–21.

[4] M. A. Ahmad, A. Patel, C. Eckert, V. Kumar, and A. Teredesai, "Fairness in machine learning for healthcare," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 3529–3530.

[5] U. Akhtar, A. M. Khattak, and S. Lee, "Challenges in managing real-time data in health information system (his)," in *International Conference on Smart Homes and Health Telematics*, Springer, 2016, pp. 305–313.

[6] B. Amarapur *et al.*, "Computer-aided diagnosis applied to mri images of brain tumor using cognition based modified level set and optimized ann classifier," *Multimedia Tools and Applications*, vol. 79, no. 5, pp. 3571–3599, 2020.

[7] M. H. Amini, H. Arasteh, and P. Siano, "Sustainable smart cities through the lens of complex interdependent infrastructures: Panorama and state-of-the-art," in *Sustainable interdependent networks II*, Springer, 2019, pp. 45–68.

[8] M. H. Amini, A. Imteaj, and P. M. Pardalos, "Interdependent networks: A data science perspective," *Patterns*, p. 100 003, 2020.

[9] P. V. AMINI, A. SHAHABINIA, H. Jafari, O. Karami, and A. Azizi, "Estimating conservation value of lighvan chay river using contingent valuation method," 2016.

[10] P. Anderson, A. Chang, D. S. Chaplot, *et al.*, "On evaluation of embodied navigation agents," *arXiv preprint arXiv:1807.06757*, 2018.

[11] B. Antal and A. Hajdu, "An ensemble-based system for automatic screening of diabetic retinopathy," *Knowledge-based systems*, vol. 60, pp. 20–27, 2014.

[12] S. Antol, A. Agrawal, J. Lu, *et al.*, "Vqa: Visual question answering," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2425–2433.

[13] R. Arandjelovic and A. Zisserman, "Look, listen and learn," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 609–617.

[14] E. Asali, F. Shenavarmasouleh, F. G. Mohammadi, P. S. Suresh, and H. R. Arabnia, "Deepmsrf: A novel deep multimodal speaker recognition framework with feature selection," in *Advances in Computer Vision and Computational Biology*, Springer, 2021, pp. 39–56.

[15] K. Ashton *et al.*, "That internet of things thing," *RFID journal*, vol. 22, no. 7, pp. 97–114, 2009.

[16] A. Auler, F. Cássaro, V. da Silva, and L. Pires, "Evidence that high temperatures and intermediate relative humidity might favor the spread of covid-19 in tropical climate: A case study for the most affected brazilian cities," *Science of The Total Environment*, p. 139 090, 2020.

[17] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[18] M. Baker, "1,500 scientists lift the lid on reproducibility," *Nature News*, vol. 533, no. 7604, p. 452, 2016.

[19] D. H. Ballard, "Animate vision," *Artificial intelligence*, vol. 48, no. 1, pp. 57–86, 1991.

[20] M. Basingab, "Distributed simulation with multi-agents for iot in a retail pharmacy facility," *Information*, vol. 11, no. 11, p. 527, 2020.

[21] S. Bauer, C. May, D. Dionysiou, G. Stamatakos, P. Buchler, and M. Reyes, "Multiscale modeling for image analysis of brain tumor studies," *IEEE transactions on biomedical engineering*, vol. 59, no. 1, pp. 25–29, 2011.

[22] A. G. Bedeian, S. G. Taylor, and A. N. Miller, "Management science on the credibility bubble: Cardinal sins and various misdemeanors," *Academy of Management Learning & Education*, vol. 9, no. 4, pp. 715–725, 2010.

[23] J. D. Berman and K. Ebisu, "Changes in us air pollution during the covid-19 pandemic," *Science of the Total Environment*, vol. 739, p. 139 864, 2020.

[24] J. Best, "Missing children, misleading statistics," *The Public Interest*, vol. 92, p. 84, 1988.

[25] K. Bhatia, S. Arora, and R. Tomar, "Diagnosis of diabetic retinopathy using machine learning classification algorithm," in *2016 2nd International Conference on Next Generation Computing Technologies (NGCT)*, IEEE, 2016, pp. 347–351.

[26] P. Bhatkoti and M. Paul, "Early diagnosis of alzheimer's disease: A multi-class deep learning framework with modified k-sparse autoencoder classification," in *2016 International Conference on Image and Vision Computing New Zealand (IVCNZ)*, IEEE, 2016, pp. 1–5.

[27] H. M. A. Bhatti, J. Li, S. Siddeeq, A. Rehman, and A. Manzoor, "Multi-detection and segmentation of breast lesions based on mask rcnn-fpn," in *2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, IEEE, 2020, pp. 2698–2704.

[28] D. Blalock, J. J. G. Ortiz, J. Frankle, and J. Guttag, "What is the state of neural network pruning?" *arXiv preprint arXiv:2003.03033*, 2020.

[29] J. M. BLAND and D. G. ALTMAN, "Misleading statistics: Errors in textbooks, software and manuals," *International journal of epidemiology*, vol. 17, no. 2, pp. 245–247, 1988.

[30] G. Bledt, M. J. Powell, B. Katz, J. Di Carlo, P. M. Wensing, and S. Kim, "Mit cheetah 3: Design and control of a robust, dynamic quadruped robot," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2018, pp. 2245–2252.

[31] G. Bradski, "The opencv library," *Dr. Dobb's Journal of Software Tools*, 2000.

[32] R. A. Brooks, "New approaches to robotics," *Science*, vol. 253, no. 5025, pp. 1227–1232, 1991.

[33] S. Camazine, P. K. Visscher, J. Finley, and R. S. Vetter, "House-hunting by honey bee swarms: Collective decisions and individual behaviors," *Insectes Sociaux*, vol. 46, no. 4, pp. 348–360, 1999.

[34] L. M. Casanova, S. Jeon, W. A. Rutala, D. J. Weber, and M. D. Sobsey, "Effects of air temperature and relative humidity on coronavirus survival on surfaces," *Applied and environmental microbiology*, vol. 76, no. 9, pp. 2712–2717, 2010.

[35] *Case study: Tensorflow in medicine - retinal imaging*, https://www.youtube.com/watch?v=oOeZ7IgEN40, 2017.

[36] A. Chang, A. Dai, T. Funkhouser, *et al.*, "Matterport3d: Learning from rgb-d data in indoor environments," *arXiv preprint arXiv:1709.06158*, 2017.

[37] Q. Chen, X. Sun, N. Zhang, Y. Cao, and B. Liu, "Mini lesions detection on diabetic retinopathy images via large scale cnn features," in *2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*, IEEE, 2019, pp. 348–352.

[38] Y. Chen, Y. Li, R. Narayan, A. Subramanian, and X. Xie, "Gene expression inference with deep learning," *Bioinformatics*, vol. 32, no. 12, pp. 1832–1839, 2016.

[39] Y. Chen, X. Qin, J. Wang, C. Yu, and W. Gao, "Fedhealth: A federated transfer learning framework for wearable healthcare," *IEEE Intelligent Systems*, vol. 35, no. 4, pp. 83–93, 2020.

[40] G. Chetty and M. Wagner, "Robust face-voice based speaker identity verification using multilevel fusion," *Image and Vision Computing*, vol. 26, no. 9, pp. 1249–1260, 2008.

[41] K. Cho, B. Van Merriënboer, C. Gulcehre, *et al.*, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[42] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," *arXiv preprint arXiv:1806.05622*, 2018.

[43] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3d u-net: Learning dense volumetric segmentation from sparse annotation," in *International conference on medical image computing and computer-assisted intervention*, Springer, 2016, pp. 424–432.

[44] E. Csoeregi, C. P. Quinn, D. W. Schmidtke, *et al.*, "Design, characterization, and one-point in vivo calibration of a subcutaneously implanted glucose electrode," *Analytical Chemistry*, vol. 66, no. 19, pp. 3131–3138, 1994.

[45] A. Ed-daoudy and K. Maalmi, "Application of machine learning model on streaming health data event in real-time to predict health status using spark," in *2018 International Symposium on Advanced Electrical and Communication Technologies (ISAECT)*, IEEE, 2018, pp. 1–4.

[46] A. Das, S. Datta, G. Gkioxari, S. Lee, D. Parikh, and D. Batra, "Embodied question answering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 2054–2063.

[47] A. Das, G. Gkioxari, S. Lee, D. Parikh, and D. Batra, "Neural modular control for embodied question answering," *arXiv preprint arXiv:1810.11181*, 2018.

[48] S. Datta, O. Maksymets, J. Hoffman, S. Lee, D. Batra, and D. Parikh, "Integrating egocentric localization for more realistic point-goal navigation agents," *arXiv preprint arXiv:2009.03231*, 2020.

[49] E. Decencière, G. Cazuguel, X. Zhang, *et al.*, "Teleophta: Machine learning and image processing methods for teleophthalmology," *Irbm*, vol. 34, no. 2, pp. 196–203, 2013.

[50] D. DeLia, "Annual bed statistics give a misleading picture of hospital surge capacity," *Annals of Emergency Medicine*, vol. 48, no. 4, pp. 384–388, 2006.

[51] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, Ieee, 2009, pp. 248–255.

[52] P. Dhakal, P. Damacharla, A. Y. Javaid, and V. Devabhaktuni, "A near real-time automatic speaker recognition architecture for voice-based user interface," *Machine Learning and Knowledge Extraction*, vol. 1, no. 1, pp. 504–520, 2019.

[53] D. DiFrancesco, "Pacemaker mechanisms in cardiac tissue," *Annual review of physiology*, vol. 55, no. 1, pp. 455–472, 1993.

[54] A. Dosovitskiy and V. Koltun, "Learning to act by predicting the future," *arXiv preprint arXiv:1611.01779*, 2016.

[55] O. J. Dunn and V. A. Clark, "Applied statistics: Analysis of variance end regression," John Wiley & Sons, Tech. Rep., 1987.

[56] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: Part i," *IEEE robotics & automation magazine*, vol. 13, no. 2, pp. 99–110, 2006.

[57] B. M. Ege, O. K. Hejlesen, O. V. Larsen, *et al.*, "Screening for diabetic retinopathy using computer based image analysis and statistical classification," *Computer methods and programs in biomedicine*, vol. 62, no. 3, pp. 165–175, 2000.

[58] H. Elayan, M. Aloqaily, and M. Guizani, "Deep federated learning for iot-based decentralized healthcare systems," in *2021 International Wireless Communications and Mobile Computing (IWCMC)*, IEEE, 2021, pp. 105–109.

[59] P. R. Epstein, "West nile virus and the climate," *Journal of Urban Health*, vol. 78, no. 2, pp. 367–371, 2001.

[60] A. Farahani, S. Voghoei, K. Rasheed, and H. R. Arabnia, "A brief review of domain adaptation," *arXiv preprint arXiv:2010.03978*, 2020.

[61] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 6202–6211.

[62] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1933–1941.

[63] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Detect to track and track to detect," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3038–3046.

[64] A. R. Feinstein, D. M. Sosin, and C. K. Wells, "The will rogers phenomenon: Stage migration and new diagnostic techniques as a source of misleading statistics for survival in cancer," *New England Journal of Medicine*, vol. 312, no. 25, pp. 1604–1608, 1985.

[65] M. Feng, S. Hu, M. H. Ang, and G. H. Lee, "2d3d-matchnet: Learning to match keypoints across 2d image and 3d point cloud," in *2019 International Conference on Robotics and Automation (ICRA)*, IEEE, 2019, pp. 4790–4796.

[66] C. Finn, A. Rajeswaran, S. Kakade, and S. Levine, "Online meta-learning," in *International Conference on Machine Learning*, PMLR, 2019, pp. 1920–1930.

[67] D. Floreano, P. Husbands, and S. Nolfi, "Evolutionary robotics," Springer Verlag, Tech. Rep., 2008.

[68] Ó. Fontenla-Romero, B. Guijarro-Berdiñas, D. Martinez-Rego, B. Pérez-Sánchez, and D. Peteiro-Barral, "Online machine learning," in *Efficiency and Scalability Methods for Computational Intellect*, IGI Global, 2013, pp. 27–54.

[69] D. A. Freedman, "Ecological inference and the ecological fallacy," *International Encyclopedia of the social & Behavioral sciences*, vol. 6, no. 4027-4030, pp. 1–7, 1999.

[70] R. Gao, R. Feris, and K. Grauman, "Learning to separate object sounds by watching unlabeled video," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 35–53.

[71] Y. Gao, J. M. Phillips, Y. Zheng, R. Min, P. T. Fletcher, and G. Gerig, "Fully convolutional structured lstm networks for joint 4d medical image segmentation," in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, IEEE, 2018, pp. 1104–1108.

[72] G. Gardner, D. Keating, T. H. Williamson, and A. T. Elliott, "Automatic detection of diabetic retinopathy using an artificial neural network: A screening tool.," *British journal of Ophthalmology*, vol. 80, no. 11, pp. 940–944, 1996.

[73] R. Gargeya and T. Leng, "Automated identification of diabetic retinopathy using deep learning," *Ophthalmology*, vol. 124, no. 7, pp. 962–969, 2017.

[74] A. El-Gazzar, M. Quaak, L. Cerliani, P. Bloem, G. van Wingen, and R. M. Thomas, "A hybrid 3dcnn and 3dc-lstm based model for 4d spatio-temporal fmri data: An abide autism classification study," in *OR 2.0 Context-Aware Operating Theaters and Machine Learning in Clinical Neuroimaging*, Springer, 2019, pp. 95–102.

[75] A. Gelman and P. N. Price, "All maps of parameter estimates are misleading," *Statistics in medicine*, vol. 18, no. 23, pp. 3221–3234, 1999.

[76] F. Ghareh Mohammadi and M. H. Amini, "Applications of nature-inspired algorithms for dimension Reduction: Enabling efficient data analytics," in *Optimization, Learning and Control for Interdependent Complex Networks*, Springer, 2019.

[77] F. Ghareh Mohammadi and M. H. Amini, "Evolutionary computation, optimization and learning algorithms for data science," in *Optimization, Learning and Control for Interdependent Complex Networks*, Springer, 2019.

[78] F. C. Ghesu, B. Georgescu, and J. Hornegger, "Efficient medical image parsing," in *Deep Learning for Medical Image Analysis*, Elsevier, 2017, pp. 55–81.

[79] F.-C. Ghesu, B. Georgescu, Y. Zheng, *et al.*, "Multi-scale deep reinforcement learning for real-time 3d-landmark detection in ct scans," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 1, pp. 176–189, 2017.

[80] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[81] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[82] O. N. Gofrit, K. C. Zorn, G. D. Steinberg, G. P. Zagaja, and A. L. Shalhav, "The will rogers phenomenon in urological oncology," *The Journal of urology*, vol. 179, no. 1, pp. 28–33, 2008.

[83] A. Gondalia, D. Dixit, S. Parashar, V. Raghava, A. Sengupta, and V. R. Sarobin, "Iot-based healthcare monitoring system for war soldiers using machine learning," *Procedia computer science*, vol. 133, pp. 1005–1013, 2018.

[84] I. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[85] D. Gordon, A. Kembhavi, M. Rastegari, J. Redmon, D. Fox, and A. Farhadi, "Iqa: Visual question answering in interactive environments," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4089–4098.

[86] S. Gössling, D. Scott, and C. M. Hall, "Pandemics, tourism and global change: A rapid assessment of covid-19," *Journal of Sustainable Tourism*, pp. 1–20, 2020.

[87] R. Greenaway-McGrevy, B. Grimm, and D. Fixler, "The revisions to gdp, gdi, and their major components," 2014.

[88] "Gross domestic product, 2nd quarter 2015 (advance estimate)," *U.S. Bureau of Economic Analysis news release*, July 30, 2015.

[89] "Gross domestic product, second quarter 2019 (third estimate)," *U.S. Bureau of Economic Analysis news release*, September 26, 2019.

[90] V. Gulshan, L. Peng, M. Coram, *et al.*, "Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs," *Jama*, vol. 316, no. 22, pp. 2402–2410, 2016.

[91] A. Gupta, T. Stewart, N. Bhulani, *et al.*, "Feasibility of wearable physical activity monitors in patients with cancer," *JCO clinical cancer informatics*, vol. 2, pp. 1–10, 2018.

[92] S. Gupta, J. Davidson, S. Levine, R. Sukthankar, and J. Malik, "Cognitive mapping and planning for visual navigation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2616–2625.

[93] B. M. ter Haar Romeny, "Multi-scale and multi-orientation medical image analysis," in *Biomedical image processing*, Springer, 2010, pp. 177–196.

[94] M. Haghi, K. Thurow, and R. Stoll, "Wearable devices in medical internet of things: Scientific research and commercially available devices," *Healthcare informatics research*, vol. 23, no. 1, pp. 4–15, 2017.

[95] F. Hassan, M. E. Shaheen, and R. Sahal, "Real-time healthcare monitoring system using online machine learning and spark streaming," *(IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 11, no. 9, 2020.

[96] A. He, C. Luo, X. Tian, and W. Zeng, "A twofold siamese network for real-time object tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4834–4843.

[97] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[98] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[99] M. L. Head, L. Holman, R. Lanfear, A. T. Kahn, and M. D. Jennions, "The extent and consequences of p-hacking in science," *PLoS biology*, vol. 13, no. 3, e1002106, 2015.

[100] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their applications*, vol. 13, no. 4, pp. 18–28, 1998.

[101] R. Held and A. Hein, "Movement-produced stimulation in the development of visually guided behavior.," *Journal of comparative and physiological psychology*, vol. 56, no. 5, p. 872, 1963.

[102] K. M. Hermann, F. Hill, S. Green, *et al.*, "Grounded language learning in a simulated 3d world," *arXiv preprint arXiv:1706.06551*, 2017.

[103] L. Herschdorfer, W. M. Negreiros, G. O. Gallucci, and A. Hamilton, "Comparison of the accuracy of implants placed with cad-cam surgical templates manufactured with various 3d printers: An in vitro study," *The Journal of Prosthetic Dentistry*, vol. 125, no. 6, pp. 905–910, 2021.

[104] T. K. Ho, "Random decision forests," in *Proceedings of 3rd international conference on document analysis and recognition*, IEEE, vol. 1, 1995, pp. 278–282.

[105] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[106] M. Hoffmann and R. Pfeifer, "The implications of embodiment for behavior and cognition: Animal and robotic case studies," *arXiv preprint arXiv:1202.0440*, 2012.

[107] K. Hosoda, "Robot finger design for developmental tactile interaction," in *Embodied Artificial Intelligence*, Springer, 2004, pp. 219–230.

[108] F. Hussain, S. A. Hassan, R. Hussain, and E. Hossain, "Machine learning for resource management in cellular and iot networks: Potentials, current solutions, and open challenges," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 2, pp. 1251–1275, 2020.

[109] J. L. Hutton, "Misleading statistics," *Pharmaceutical Medicine*, vol. 24, no. 3, pp. 145–149, 2010.

[110] J. Hutton, "Number needed to treat: Properties and problems," *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, vol. 163, no. 3, pp. 381–402, 2000.

[111] F. Iida and R. Pfeifer, "Cheap rapid locomotion of a quadruped robot: Self-stabilization of bounding gait," in *Intelligent autonomous systems*, IOS Press Amsterdam, The Netherlands, vol. 8, 2004, pp. 642–649.

[112] A. Imteaj, U. Thakker, S. Wang, J. Li, and M. H. Amini, "Federated learning for resource-constrained iot devices: Panoramas and state-of-the-art," *arXiv preprint arXiv:2002.10610*, 2020.

[113] M. Jaderberg, V. Mnih, W. M. Czarnecki, *et al.*, "Reinforcement learning with unsupervised auxiliary tasks," *arXiv preprint arXiv:1611.05397*, 2016.

[114] M. A. Jan, F. Khan, M. Alam, and M. Usman, "A payload-based mutual authentication scheme for internet of things," *Future Generation Computer Systems*, vol. 92, pp. 1028–1039, 2019.

[115] D. Jayaraman, R. Gao, and K. Grauman, "Shapecodes: Self-supervised feature learning by lifting views to viewgrids," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 120–136.

[116] D. Jayaraman and K. Grauman, "Learning image representations tied to egomotion from unlabeled video," *International Journal of Computer Vision*, vol. 125, no. 1-3, pp. 136–161, 2017.

[117] S. Jégou, M. Drozdzal, D. Vazquez, A. Romero, and Y. Bengio, "The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017, pp. 11–19.

[118] C. Jiang, S. Wang, X. Liang, H. Xu, and N. Xiao, "Elixirnet: Relation-aware network architecture adaptation for medical lesion detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 11 093–11 100.

[119] H. Jiang, C. Cai, X. Ma, Y. Yang, and J. Liu, "Smart home based on wifi sensing: A survey," *IEEE Access*, vol. 6, pp. 13 317–13 325, 2018.

[120] G. H. John and P. Langley, "Estimating continuous distributions in bayesian classifiers," in *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, Morgan Kaufmann Publishers Inc., 1995, pp. 338–345.

[121] T. Jourdan, A. Boutet, A. Bahi, and C. Frindel, "Privacy-preserving iot framework for activity recognition in personal healthcare monitoring," *ACM Transactions on Computing for Healthcare*, vol. 2, no. 1, pp. 1–22, 2020.

[122] J. Kawahara and G. Hamarneh, "Multi-resolution-tract cnn with hybrid pretrained and skin-lesion trained layers," in *International workshop on machine learning in medical imaging*, Springer, 2016, pp. 164–171.

[123] L. U. Khan, W. Saad, Z. Han, E. Hossain, and C. S. Hong, "Federated learning for internet of things: Recent advances, taxonomy, and open challenges," *IEEE Communications Surveys & Tutorials*, 2021.

[124] M. A. Khan, T. Akram, Y.-D. Zhang, and M. Sharif, "Attributes based skin lesion detection and recognition: A mask rcnn and transfer learning-based deep learning framework," *Pattern Recognition Letters*, vol. 143, pp. 58–66, 2021.

[125] A. Khatami, A. Khosravi, T. Nguyen, C. P. Lim, and S. Nahavandi, "Medical image analysis using wavelet transform and deep belief networks," *Expert Systems with Applications*, vol. 86, pp. 190–198, 2017.

[126] Y. Kim, S. Lee, and S. Lee, "Coexistence of zigbee-based wban and wifi for health telemonitoring systems," *IEEE journal of biomedical and health informatics*, vol. 20, no. 1, pp. 222–230, 2015.

[127] M. R. Kinthada, S. Bodda, and S. B. K. Mande, "Emedicare: Mhealth solution for patient medication guidance and assistance," in *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, IEEE, 2016, pp. 657–661.

[128] D. G. Kleinbaum, K. Dietz, M. Gail, M. Klein, and M. Klein, *Logistic regression*. Springer, 2002.

[129] Y. Koda, Y. Yoshitomi, M. Nakano, and M. Tabuse, "A facial expression recognition for a speaker of a phoneme of vowel using thermal image processing and a speech recognition system," in *RO-MAN 2009-The 18th IEEE International Symposium on Robot and Human Interactive Communication*, IEEE, 2009, pp. 955–960.

[130] N. Komninos, E. Philippou, and A. Pitsillides, "Survey in smart grid and smart home security: Issues, challenges and countermeasures," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 4, pp. 1933–1954, 2014.

[131] J. Konevcny, H. B. McMahan, D. Ramage, and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," *arXiv preprint arXiv:1610.02527*, 2016.

[132] R. Krishna, Y. Zhu, O. Groth, *et al.*, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *International journal of computer vision*, vol. 123, no. 1, pp. 32–73, 2017.

[133] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[134] J. Lachure, A. Deorankar, S. Lachure, S. Gupta, and R. Jadhav, "Diabetic retinopathy using morphological operations and machine learning," in *2015 IEEE International Advance Computing Conference (IACC)*, IEEE, 2015, pp. 617–622.

[135] J. Lagergren, E. Rutter, and K. Flores, "Region growing with convolutional neural networks for biomedical image segmentation," *arXiv preprint arXiv:2009.11717*, 2020.

[136] J. Larson-Hall, "Our statistical intuitions may be misleading us: Why we need robust statistics," *Language Teaching*, vol. 45, no. 4, pp. 460–474, 2012.

[137] P. A. Leake, G. T. Hradek, and R. L. Snyder, "Chronic electrical stimulation by a cochlear implant promotes survival of spiral ganglion neurons after neonatal deafness," *Journal of Comparative Neurology*, vol. 412, no. 4, pp. 543–562, 1999.

[138] C. S. Lee and A. Y. Lee, "Clinical applications of continual learning machine learning," *The Lancet Digital Health*, vol. 2, no. 6, e279–e281, 2020.

[139] K. Lee, A. Agrawal, and A. Choudhary, "Real-time disease surveillance using twitter data: Demonstration on flu and cancer," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2013, pp. 1474–1477.

[140] K. Lee, L. Sunwoo, T. Kim, and K. J. Lee, "Spider u-net: Incorporating inter-slice connectivity using lstm for 3d blood vessel segmentation," *Applied Sciences*, vol. 11, no. 5, p. 2014, 2021.

[141] J. Lee Rodgers and W. A. Nicewander, "Thirteen ways to look at the correlation coefficient," *The American Statistician*, vol. 42, no. 1, pp. 59–66, 1988.

[142] N. Leiper, "A conceptual analysis of tourism-supported employment which reduces the incidence of exaggerated, misleading statistics about jobs," *Tourism Management*, vol. 20, no. 5, pp. 605–613, 1999.

[143] J. Li, J. Cai, F. Khan, *et al.*, "A secured framework for sdn-based edge computing in iot-enabled healthcare system," *IEEE Access*, vol. 8, pp. 135 479–135 490, 2020.

[144] L. Li, X. Hu, H. Huang, *et al.*, "Latent source mining of fmri data via deep belief network," in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, IEEE, 2018, pp. 595–598.

[145] S. Li, H. Lei, F. Zhou, J. Gardezi, and B. Lei, "Longitudinal and multi-modal data learning for parkinson s disease diagnosis via stacked sparse auto-encoder," in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, IEEE, 2019, pp. 384–387.

[146] W. Li, Y. Chai, F. Khan, *et al.*, "A comprehensive survey on machine learning-based big data analytics for iot-enabled smart healthcare system," *Mobile Networks and Applications*, pp. 1–19, 2021.

[147] W. Y. B. Lim, N. C. Luong, D. T. Hoang, *et al.*, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 2031–2063, 2020.

[148] T.-Y. Lin, M. Maire, S. Belongie, *et al.*, "Microsoft coco: Common objects in context," in *European conference on computer vision*, Springer, 2014, pp. 740–755.

[149] R. Liu, L. Wei, and P. Zhang, "A deep learning framework for drug repurposing via emulating clinical trials on real-world patient data," *Nature Machine Intelligence*, vol. 3, no. 1, pp. 68–75, 2021.

[150] Z. Liu, C. Opas, and S. M. Krishnan, "Automatic image analysis of fundus photograph," in *Proceedings of the 19th Annual International Conference of the IEEE Engineering in Medicine and Biology Society.'Magnificent Milestones and Emerging Opportunities in Medical Engineering'(Cat. No. 97CH36136)*, IEEE, vol. 2, 1997, pp. 524–525.

[151] Z. Liu, C. Yao, H. Yu, and T. Wu, "Deep reinforcement learning with its application for lung cancer detection in medical internet of things," *Future Generation Computer Systems*, vol. 97, pp. 1–9, 2019.

[152] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[153] J. Ma, J. Huang, Q. Feng, *et al.*, "Low-dose computed tomography image restoration using previous normal-dose scan," *Medical physics*, vol. 38, no. 10, pp. 5713–5731, 2011.

[154] A. Madabhushi, S. Agner, A. Basavanhally, S. Doyle, and G. Lee, "Computer-aided prognosis: Predicting patient and disease outcome via quantitative fusion of multi-scale, multi-modal data," *Computerized medical imaging and graphics*, vol. 35, no. 7-8, pp. 506–514, 2011.

[155] W. G. Madow, *Elementary sampling theory*, 1968.

[156] C. Mahiba and A. Jayachandran, "Severity analysis of diabetic retinopathy in retinal images using hybrid structure descriptor and modified cnns," *Measurement*, vol. 135, pp. 762–767, 2019.

[157] R. D. Manzanedo and P. Manning, "Covid-19: Lessons for the climate change emergency," *Science of the Total Environment*, vol. 742, p. 140 563, 2020.

[158] K. Mao, R. Tang, X. Wang, W. Zhang, and H. Wu, "Feature representation using deep autoencoder for lung nodule image classification," *Complexity*, vol. 2018, 2018.

[159] Z. Mao, Y. Su, G. Xu, *et al.*, "Spatio-temporal deep learning method for adhd fmri classification," *Information Sciences*, vol. 499, pp. 1–11, 2019.

[160] P. Massin, A. Chabouis, A. Erginay, *et al.*, "Ophdiat©: A telemedical network screening system for diabetic retinopathy in the île-de-france," *Diabetes & metabolism*, vol. 34, no. 3, pp. 227–234, 2008.

[161] D. Maturana and S. Scherer, "Voxnet: A 3d convolutional neural network for real-time object recognition," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2015, pp. 922–928.

[162] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," in *Psychology of learning and motivation*, vol. 24, Elsevier, 1989, pp. 109–165.

[163] D. L. McGuinness, F. Van Harmelen, *et al.*, "Owl web ontology language overview," *W3C recommendation*, vol. 10, no. 10, p. 2004, 2004.

[164] C. Mellis, "Lies, damned lies and statistics: Clinical importance versus statistical significance in research," *Paediatric respiratory reviews*, vol. 25, pp. 88–93, 2018.

[165] E. Miller, "An introduction to the resource description framework," *Bulletin of the American Society for Information Science and Technology*, vol. 25, no. 1, pp. 15–19, 1998.

[166] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *2016 fourth international conference on 3D vision (3DV)*, IEEE, 2016, pp. 565–571.

[167] D. Mishkin, A. Dosovitskiy, and V. Koltun, "Benchmarking classic and learned navigation in complex 3d environments," *arXiv preprint arXiv:1901.10915*, 2019.

[168] T. Mitchell, W. Cohen, E. Hruschka, *et al.*, "Never-ending learning," *Communications of the ACM*, vol. 61, no. 5, pp. 103–115, 2018.

[169] F. G. Mohammadi and H. Sajedi, "Region based image steganalysis using artificial bee colony," *Journal of Visual Communication and Image Representation*, vol. 44, pp. 214–226, 2017.

[170] F. G. Mohammadi and M. S. Abadeh, "Image steganalysis using a bee colony based feature selection algorithm," *Engineering Applications of Artificial Intelligence*, vol. 31, pp. 35–43, 2014.

[171] F. G. Mohammadi and M. S. Abadeh, "A new metaheuristic feature subset selection approach for image steganalysis," *Journal of Intelligent & Fuzzy Systems*, vol. 27, no. 3, pp. 1445–1455, 2014.

[172] F. G. Mohammadi and M. H. Amini, "Promises of meta-learning for device-free human sensing: Learn to sense," in *Proceedings of the 1st ACM International Workshop on Device-Free Human Sensing*, 2019, pp. 44–47.

[173] F. G. Mohammadi, M. H. Amini, and H. R. Arabnia, "An introduction to advanced machine learning: Meta-learning algorithms, applications, and promises," in *Optimization, Learning, and Control for Interdependent Complex Networks*, Springer, 2020, pp. 129–144.

[174] F. G. Mohammadi, H. R. Arabnia, and M. H. Amini, "On parameter tuning in meta-learning for computer vision," in *2019 International Conference on Computational Science and Computational Intelligence (CSCI)*, IEEE, 2019, pp. 300–305.

[175] F. G. Mohammadi, F. Shenavarmasouleh, M. H. Amini, and H. R. Arabnia, "Impact of weather conditions on the covid-19 pandemic in the united states: A big data analytics approach," in *2020 International Conference on Computational Science and Computational Intelligence (CSCI)*, IEEE, 2020, pp. 418–423.

[176] F. G. Mohammadi, F. Shenavarmasouleh, M. H. Amini, and H. R. Arabnia, "Data analytics for smart cities: Challenges and promises," *arXiv preprint arXiv:2109.05581*, 2021.

[177] F. G. Mohammadi, F. Shenavarmasouleh, M. H. Amini, and H. R. Arabnia, "Evolutionary algorithms and efficient data analytics for image processing," in *2021 15th International Conference on Ubiquitous Information Management and Communication (IMCOM)*, IEEE, 2021, pp. 1–8.

[178] S. P. Mudunuri and S. Biswas, "Low resolution face recognition across variations in pose and illumination," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 5, pp. 1034–1040, 2015.

[179] S. Murata, A. Kamimura, H. Kurokawa, E. Yoshida, K. Tomita, and S. Kokaji, "Self-reconfigurable robots: Platforms for emerging functionality," in *Embodied Artificial Intelligence*, Springer, 2004, pp. 312–330.

[180] F. Navarro, A. Sekuboyina, D. Waldmannstetter, J. C. Peeken, S. E. Combs, and B. H. Menze, "Deep reinforcement learning for organ localization in ct," in *Medical Imaging with Deep Learning*, PMLR, 2020, pp. 544–554.

[181] R. Nobles and D. Schiff, "Misleading statistics within criminal trials: The sally clark case," *Significance*, vol. 2, no. 1, pp. 17–19, 2005.

[182] B. A. Nosek, G. Alter, G. C. Banks, *et al.*, "Promoting an open research culture," *Science*, vol. 348, no. 6242, pp. 1422–1425, 2015.

[183] M. Otoom, N. Otoum, M. A. Alzubaidi, Y. Etoom, and R. Banihani, "An iot-based framework for early identification and monitoring of covid-19 cases," *Biomedical signal processing and control*, vol. 62, p. 102 149, 2020.

[184] Ş. Öztürk, "Stacked auto-encoder based tagging with deep features for content-based medical image retrieval," *Expert Systems with Applications*, vol. 161, p. 113 693, 2020.

[185] P. Padilla, M. López, J. M. Górriz, J. Ramirez, D. Salas-Gonzalez, and I. Alvarez, "Nmf-svm based cad tool applied to functional brain images for the diagnosis of alzheimer's disease," *IEEE Transactions on medical imaging*, vol. 31, no. 2, pp. 207–216, 2011.

[186] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2009.

[187] S. Pare, A. Kumar, G. Singh, and V. Bajaj, "Image segmentation using multilevel thresholding: A research review," *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, vol. 44, no. 1, pp. 1–29, 2020.

[188] S. Parekh, S. Essid, A. Ozerov, N. Q. Duong, P. Pérez, and G. Richard, "Guiding audio source separation by video object information," in *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, IEEE, 2017, pp. 61–65.

[189] J. H. Park, M. Younas, H. R. Arabnia, and N. Chilamkurti, *Emerging ict applications and services big data, iot, and cloud computing*, 2021.

[190] K. Pearson, "Vii. mathematical contributions to the theory of evolution. iii. regression, heredity, and panmixia," *Philosophical Transactions of the Royal Society of London. Series A, containing papers of a mathematical or physical character*, no. 187, pp. 253–318, 1896.

[191] X. Peng and C. Schmid, "Multi-region two-stream r-cnn for action detection," in *European conference on computer vision*, Springer, 2016, pp. 744–759.

[192] J. Pérez, M. Arenas, and C. Gutierrez, "Semantics and complexity of sparql," *ACM Transactions on Database Systems (TODS)*, vol. 34, no. 3, pp. 1–45, 2009.

[193] W. H. Pinaya, A. Gadelha, O. M. Doyle, *et al.*, "Using deep belief network modelling to characterize differences in brain morphometry in schizophrenia," *Scientific reports*, vol. 6, no. 1, pp. 1–9, 2016.

[194] D. Pivoto, P. D. Waquil, E. Talamini, C. P. S. Finocchio, V. F. Dalla Corte, and G. de Vargas Mores, "Scientific development of smart farming technologies and their application in brazil," *Information processing in agriculture*, vol. 5, no. 1, pp. 21–32, 2018.

[195] I. Portugal, P. Alencar, and D. Cowan, "The use of machine learning algorithms in recommender systems: A systematic review," *Expert Systems with Applications*, vol. 97, pp. 205–227, 2018.

[196] P. Pramod, "Gps based advanced soldier tracking with emergency messages & communication system," *International Journal of advance research in Computer science and management studies research Article*, vol. 2, no. 6, 2014.

[197] H. Pratt, F. Coenen, D. M. Broadbent, S. P. Harding, and Y. Zheng, "Convolutional neural networks for diabetic retinopathy," *Procedia Computer Science*, vol. 90, pp. 200–205, 2016.

[198] R. Priya and P. Aruna, "Diagnosis of diabetic retinopathy using machine learning techniques," *ICTACT Journal on soft computing*, vol. 3, no. 4, pp. 563–575, 2013.

[199] D. Ravi, A. B. Szczotka, S. P. Pereira, and T. Vercauteren, "Adversarial training with cycle consistency for unsupervised super-resolution in endomicroscopy," *Medical image analysis*, vol. 53, pp. 123–131, 2019.

[200] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 6, pp. 1137–1149, 2016.

[201] *Resized version of the diabetic retinopathy kaggle competition dataset*, https://www.kaggle.com/tanlikesmath/diabet retinopathy-resized, 2019.

[202] A. L. R. Ribeiro and N. W. A. Sousa, "Besides the climate model, other variables driving the covid-19 spread in brazil," *The Science of the Total Environment*, vol. 737, p. 140 211, 2020.

[203] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, Springer, 2015, pp. 234–241.

[204] N. Al-Rousan and H. Al-Najjar, "Data analysis of coronavirus covid-19 epidemic in south korea based on recovered and death cases," *Journal of Medical Virology*, 2020.

[205] S. Roychowdhury, D. D. Koozekanani, and K. K. Parhi, "Dream: Diabetic retinopathy analysis using machine learning," *IEEE journal of biomedical and health informatics*, vol. 18, no. 5, pp. 1717–1728, 2013.

[206] D. R.-.-J. G.-.-J. Rydning, "The digitization of the world from edge to core," *Framingham: International Data Corporation*, p. 16, 2018.

[207] F. Sadeghi and S. Levine, "Cad2rl: Real single-image flight without a single real image," *arXiv preprint arXiv:1611.04201*, 2016.

[208] A. K. Sagar, S. Singh, and A. Kumar, "Energy-aware wban for health monitoring using critical data routing (cdr)," *Wireless Personal Communications*, pp. 1–30, 2020.

[209] S. Sagiroglu and D. Sinanc, "Big data: A review," in *2013 international conference on collaboration technologies and systems (CTS)*, IEEE, 2013, pp. 42–47.

[210] M. Sahoo, S. Pal, and M. Mitra, "Automatic segmentation of accumulated fluid inside the retinal layers from optical coherence tomography images," *Measurement*, vol. 101, pp. 138–144, 2017.

[211] M. Salvi and F. Molinari, "Multi-tissue and multi-scale approach for nuclei segmentation in h&e stained images," *Biomedical engineering online*, vol. 17, no. 1, pp. 1–13, 2018.

[212] R. K. Samala, H.-P. Chan, L. M. Hadjiiski, M. A. Helvie, K. H. Cha, and C. D. Richter, "Multi-task transfer learning deep convolutional neural network: Application to computer-aided diagnosis of breast cancer on mammograms," *Physics in Medicine & Biology*, vol. 62, no. 23, p. 8894, 2017.

[213] G. A. Sandag *et al.*, "A prediction model of company health using bagging classifier," *JITK (Jurnal Ilmu Pengetahuan Dan Teknologi Komputer)*, vol. 6, no. 1, pp. 41–46, 2020.

[214] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[215] M. Savva, A. X. Chang, A. Dosovitskiy, T. Funkhouser, and V. Koltun, "Minos: Multimodal indoor simulator for navigation in complex environments," *arXiv preprint arXiv:1712.03931*, 2017.

[216] M. Savva, A. Kadian, O. Maksymets, *et al.*, "Habitat: A platform for embodied ai research," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 9339–9347.

[217] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[218] P. Sedgwick, "Pearson s correlation coefficient," *Bmj*, vol. 345, e4483, 2012.

[219] S. Selvaraj and S. Sundaravaradhan, "Challenges and opportunities in iot healthcare systems: A systematic review," *SN Applied Sciences*, vol. 2, no. 1, pp. 1–8, 2020.

[220] S. Shahrestani, "Assistive iot: Deployment scenarios and challenges," in *Internet of Things and Smart Environments*, Springer, 2017, pp. 75–95.

[221] F. Shenavarmasouleh and H. Arabnia, "Causes of misleading statistics and research results irreproducibility: A concise review," in *2019 International Conference on Computational Science and Computational Intelligence (CSCI)*, IEEE, 2019, pp. 465–470.

[222] F. Shenavarmasouleh and H. R. Arabnia, "Drdr: Automatic masking of exudates and microaneurysms caused by diabetic retinopathy using mask r-cnn and transfer learning," in *Advances in Computer Vision and Computational Biology*, Springer, 2021, pp. 307–318.

[223] F. Shenavarmasouleh, F. G. Mohammadi, M. H. Amini, and H. R. Arabnia, "Drdr ii: Detecting the severity level of diabetic retinopathy using mask rcnn and transfer learning," *arXiv preprint arXiv:2011.14733*, 2020.

[224] F. Shenavarmasouleh, F. G. Mohammadi, M. H. Amini, and H. R. Arabnia, "Embodied ai-driven operation of smart cities: A concise review," *arXiv preprint arXiv:2108.09823*, 2021.

[225] F. Shenavarmasouleh, F. G. Mohammadi, M. H. Amini, T. Taha, K. Rasheed, and H. R. Arabnia, "Drdrv3: Complete lesion detection in fundus images using mask r-cnn, transfer learning, and lstm," *arXiv preprint arXiv:2108.08095*, 2021. arXiv: 2108.08095 [eess.IV].

[226] H. A. Simon, "Spurious correlation: A causal interpretation," *Journal of the American statistical Association*, vol. 49, no. 267, pp. 467–479, 1954.

[227] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[228] A. Singh, V. Natarajan, M. Shah, *et al.*, "Towards vqa models that can read," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8317–8326.

[229] A. K. Singh, B. Ganapathysubramanian, S. Sarkar, and A. Singh, "Deep learning for plant stress phenotyping: Trends and future perspectives," *Trends in plant science*, vol. 23, no. 10, pp. 883–898, 2018.

[230] S. P. Singh and R. S. Sutton, "Reinforcement learning with replacing eligibility traces," *Machine learning*, vol. 22, no. 1-3, pp. 123–158, 1996.

[231] N. Soans, E. Asali, Y. Hong, and P. Doshi, "Sa-net: Robust state-action recognition for learning from observations," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2020, pp. 2153–2159.

[232] S. Song, F. Yu, A. Zeng, A. X. Chang, M. Savva, and T. Funkhouser, "Semantic scene completion from a single depth image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1746–1754.

[233] T.-H. Song, V. Sanchez, H. Eldaly, and N. M. Rajpoot, "Hybrid deep autoencoder with curvature gaussian for detection of various types of cells in bone marrow trephine biopsy images," in *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*, IEEE, 2017, pp. 1040–1043.

[234] A. Sopharak, M. N. Dailey, B. Uyyanonvara, *et al.*, "Machine learning approach to automatic exudate detection in retinal images from diabetic patients," *Journal of Modern optics*, vol. 57, no. 2, pp. 124–135, 2010.

[235] T. Sowell, *Economic facts and fallacies*. Basic Books, 2011.

[236] C. Spearman, "The proof and measurement of association between two things," *The American Journal of Psychology*, vol. 15, no. 1, pp. 72–101, 1904, ISSN: 00029556. [Online]. Available: http://www.jstor.org/stable/1412159.

[237] A. Speiser, S. C. Turaga, and J. H. Macke, "Teaching deep neural networks to localize sources in super-resolution microscopy by combining simulation-based learning and unsupervised learning," *arXiv*, 2019.

[238] B. Sreedhar, M. S. BE, and M. S. Kumar, "A comparative study of melanoma skin cancer detection in traditional and current image processing techniques," in *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*, IEEE, 2020, pp. 654–658.

[239] L. Steels, "Evolving grounded communication for robots," *Trends in cognitive sciences*, vol. 7, no. 7, pp. 308–312, 2003.

[240] C. Szegedy, W. Liu, Y. Jia, *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[241] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[242] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*, PMLR, 2019, pp. 6105–6114.

[243] I. Tenney, D. Das, and E. Pavlick, *Bert rediscovers the classical nlp pipeline*, 2019. arXiv: 1905.05950 [cs.CL].

[244] S. K. Thakur, D. P. Singh, and J. Choudhary, "Lung cancer identification: A review on detection and classification," *Cancer and Metastasis Reviews*, vol. 39, pp. 989–998, 2020.

[245] R. Tosepu, J. Gunawan, D. S. Effendy, H. Lestari, H. Bahar, P. Asfian, *et al.*, "Correlation between weather and covid-19 pandemic in jakarta, indonesia," *Science of The Total Environment*, p. 138 436, 2020.

[246] S. Truex, N. Baracaldo, A. Anwar, *et al.*, "A hybrid approach to privacy-preserving federated learning," in *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security*, 2019, pp. 1–11.

[247] H.-Y. F. Tung, R. Cheng, and K. Fragkiadaki, "Learning spatial common sense with geometry-aware recurrent networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2595–2603.

[248] M. Tye, "Qualia," 1997.

[249] A. Ullah, M. Azeem, H. Ashraf, A. A. Alaboudi, M. Humayun, and N. Jhanjhi, "Secure healthcare data aggregation and transmission in iot a survey," *IEEE Access*, vol. 9, pp. 16 849–16 865, 2021.

[250] D. Usher, M. Dumskyj, M. Himaga, T. H. Williamson, S. Nussey, and J. Boyce, "Automated detection of diabetic retinopathy in digital retinal images: A tool for diabetic retinopathy screening," *Diabetic Medicine*, vol. 21, no. 1, pp. 84–90, 2004.

[251] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[252] P. Verma and S. K. Sood, "A comprehensive framework for student stress monitoring in fog-cloud iot environment: M-health perspective," *Medical & biological engineering & computing*, vol. 57, no. 1, pp. 231–244, 2019.

[253] V. Vippalapalli and S. Ananthula, "Internet of things (iot) based smart health care system," in *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, IEEE, 2016, pp. 1229–1233.

[254] A. Wald, "A reprint of'a method of estimating plane vulnerability based on damage of survivors.," CENTER FOR NAVAL ANALYSES ALEXANDRIA VA OPERATIONS EVALUATION GROUP, Tech. Rep., 1980.

[255] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, "Glue: A multi-task benchmark and analysis platform for natural language understanding," *arXiv preprint arXiv:1804.07461*, 2018.

[256] C. Wang and C.-Y. Hsu, "Rankings correlation study: Brand search volume vs. brand sales volume," in *2020 5th IEEE International Conference on Big Data Analytics (ICBDA), IEEE*, 2020, pp. 6–10.

[257] F. Wang, M. Jiang, C. Qian, *et al.*, "Residual attention network for image classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3156–3164.

[258] J. Wang, Z. Feng, Z. Chen, *et al.*, "Bandwidth-efficient live video analytics for drones via edge computing," in *2018 IEEE/ACM Symposium on Edge Computing (SEC)*, IEEE, 2018, pp. 159–173.

[259] Y. Wang, H. Su, B. Zhang, and X. Hu, "Interpret neural networks by identifying critical data routing paths," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8906–8914.

[260] M. Ware and M. Mabe, "The stm report: An overview of scientific and scholarly journal publishing," 2015.

[261] E. W. Weisstein, "Bonferroni correction," 2004.

[262] S. M. Werz, S. Zeichner, B.-I. Berg, H.-F. Zeilhofer, and F. Thieringer, "3d printed surgical simulation models as educational tool by maxillofacial surgeons," *European Journal of Dental Education*, vol. 22, no. 3, e500–e505, 2018.

[263] R. R. Wilcox, *Introduction to robust estimation and hypothesis testing*. Academic press, 2011.

[264] C. Wiwie, J. Baumbach, and R. Röttger, "Comparing the performance of biomedical clustering methods," *Nature methods*, vol. 12, no. 11, pp. 1033–1038, 2015.

[265] Q. Wu, K. He, and X. Chen, "Personalized federated learning for intelligent iot applications: A cloud-edge based framework," *IEEE Open Journal of the Computer Society*, vol. 1, pp. 35–44, 2020.

[266] Y. Wu, Y. Wu, G. Gkioxari, and Y. Tian, "Building generalizable agents with a realistic and rich 3d environment," *arXiv preprint arXiv:1801.02209*, 2018.

[267] F. Xia, A. R. Zamir, Z. He, A. Sax, J. Malik, and S. Savarese, "Gibson env: Real-world perception for embodied agents," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9068–9079.

[268] Y. Xing, J. Wang, X. Chen, and G. Zeng, "2.5 d convolution for rgb-d semantic segmentation," in *2019 IEEE International Conference on Image Processing (ICIP)*, IEEE, 2019, pp. 1410–1414.

[269] J. Xu, B. S. Glicksberg, C. Su, P. Walker, J. Bian, and F. Wang, "Federated learning for healthcare informatics," *Journal of Healthcare Informatics Research*, vol. 5, no. 1, pp. 1–19, 2021.

[270] J. Xu, L. Xiang, Q. Liu, *et al.*, "Stacked sparse autoencoder (ssae) for nuclei detection on breast cancer histopathology images," *IEEE transactions on medical imaging*, vol. 35, no. 1, pp. 119–130, 2015.

[271] Y. Xu, A. Hosny, R. Zeleznik, *et al.*, "Deep learning predicts lung cancer treatment response from serial medical imaging," *Clinical Cancer Research*, vol. 25, no. 11, pp. 3266–3275, 2019.

[272] Y. Xue, F. G. Farhat, O. Boukrina, *et al.*, "A multi-path 2.5 dimensional convolutional neural network system for segmenting stroke lesions in brain mri images," *NeuroImage: Clinical*, vol. 25, p. 102 118, 2020.

[273] D. Yadav, A. K. Karn, A. Giddalur, A. Dhiman, S. Sharma, A. K. Yadav, *et al.*, "Microaneurysm detection using color locus detection method," *Measurement*, vol. 176, p. 109 084, 2021.

[274] S. S. Yadav and S. M. Jadhav, "Machine learning algorithms for disease prediction using iot environment," *International Journal of Engineering and Advanced Technology*, vol. 8, no. 6, pp. 4303–4307, 2019.

[275] J. Yang, Z. Ren, M. Xu, *et al.*, *Embodied visual recognition*, 2019. arXiv: 1904.04404 [cs.CV].

[276] X. Yang, P. Molchanov, and J. Kautz, "Multilayer and multimodal fusion of deep neural networks for video classification," in *Proceedings of the 24th ACM international conference on Multimedia*, 2016, pp. 978–987.

[277] W. Yao, A. Yahya, F. Khan, *et al.*, "A secured and efficient communication scheme for decentralized cognitive radio-based internet of vehicles," *IEEE Access*, vol. 7, pp. 160 889–160 900, 2019.

[278] W. Ying, Y. Zhang, J. Huang, and Q. Yang, "Transfer learning via learning to transfer," in *International conference on machine learning*, PMLR, 2018, pp. 5085–5094.

[279] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv preprint arXiv:1511.07122*, 2015.

[280] B. Yuan, S. Ge, and W. Xing, "A federated learning framework for healthcare iot devices," *arXiv preprint arXiv:2005.05083*, 2020.

[281] R. Zellers, Y. Bisk, R. Schwartz, and Y. Choi, "Swag: A large-scale adversarial dataset for grounded commonsense inference," *arXiv preprint arXiv:1808.05326*, 2018.

[282] X. Zhang, J. Zou, K. He, and J. Sun, "Accelerating very deep convolutional networks for classification and detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 10, pp. 1943–1955, 2015.

[283] P. Zhou, X. Han, V. I. Morariu, and L. S. Davis, "Two-stream neural networks for tampered face detection," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, IEEE, 2017, pp. 1831–1839.

[284] W. Zhu, C. Liu, W. Fan, and X. Xie, "Deeplung: Deep 3d dual path nets for automated pulmonary nodule detection and classification," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, 2018, pp. 673–681.