NATIONAL TECHNICAL UNIVERSITY OF ATHENS

SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

DIVISION OF INFORMATION TRANSMISSION SYSTEMS AND MATERIAL TECHNOLOGY

Classification of cough recordings for Covid-19 detection: Development of Machine Learning and Deep Learning models incorporating Concept Drift Adaptation

DIPLOMA THESIS

CHRYSOVALANTIS-KONSTANTINOS ANDREAS

Supervisor: Konstantina Nikita
Professor NTUA

Athens, June 2024

NATIONAL TECHNICAL UNIVERSITY OF ATHENS
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING
DIVISION OF INFORMATION TRANSMISSION SYSTEMS AND MATERIAL TECHNOLOGY

« Classification of cough recordings for Covid-19 detection: Development of Machine Learning and Deep Learning models incorporating Concept Drift Adaptation »

DIPLOMA THESIS

CHRYSOVALANTIS-KONSTANTINOS ANDREAS

Advisory Board: Konstantina Nikita
Professor NTUA

Approved by the review board on 09/07/2024.

....	....	....
Konstantina Nikita Professor NTUA	Giorgos Stamou Professor NTUA	Athanasios Voulodimos Assistant Professor NTUA

Athens, June 2024

....

Chrysovalantis – Konstantinos Andreas
Graduate Electrical and Computer Engineering N.T.U.A.

....

Χρυσοβαλάντης – Κωνσταντίνος Ανδρεάς
Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Περίληψη

Η παρούσα διπλωματική εργασία εξετάζει την αποτελεσματικότητα της ανίχνευσης COVID - 19 από καταγραφές βήχα, χρησιμοποιώντας τεχνικές μηχανικής και βαθιάς μάθησης, σε μια προσπάθεια να μειωθεί το κόστος και ο χρόνος που απαιτείται για τη διάγνωση του ασθενούς. Επιπλέον, εξετάζεται η εφαρμογή μεθόδων προσαρμογής σε μετατοπίσεις εννοιών, λόγω των συνεχώς μεταβαλλόμενων χαρακτηριστικών του ιού, με στόχο τη διατήρηση της επίδοσης των μοντέλων που αναπτύσσονται. Για το σκοπό αυτό, διερευνάται η χρήση διαφορετικών μεθόδων μηχανικής (αλγόριθμος τυχαίων δασών, πολύ-επίπεδα δίκτυα perceptron) και βαθιάς μάθησης (συνελικτικά τεχνητά νευρωνικά δίκτυα), καθώς και προσεγγίσεων μεταφοράς μάθησης μέσω της αξιοποίησης προ-εκπαιδευμένων μοντέλων. Η ανάπτυξη και αξιολόγηση των μοντέλων βασίζεται στη χρήση του συνόλου δεδομένων Coswara. Για την αντιμετώπιση της μη ισορροπημένης φύσης του συνόλου δεδομένων αξιοποιούνται τεχνικές παραγωγής συνθετικών δεδομένων (SMOTE), μάθησης με ευαισθησία κόστους και βελτιστοποίησης των κατωφλίων ταξινόμησης. Η υψηλότερη επίδοση με βάση το κριτήριο AUROC (80,21%) επιτυγχάνεται από μια αρχιτεκτονική συνελικτικών νευρωνικών δικτύων που χρησιμοποιεί το προ-εκπαιδευμένο VGG-16 ως μοντέλο βάσης. Για την προσαρμογή στην μετατόπιση των εννοιών, τα τελευταία πυκνά στρώματα του μοντέλου επανεκπαιδεύονται με χρήση κατάλληλης μεθόδου κανονικοποίησης που οδηγεί σε βελτίωση της επίδοσης του μοντέλου ως προς το κριτήριο AUROC έως και 5%.

Λέξεις Κλειδιά

COVID - 19, Ταξινόμηση βήχα, Προσαρμογή μετατόπισης έννοιας, Συνελικτικά Νευρωνικά Δίκτυα, Φασματογράφημα Mel

Abstract

This thesis examines the effectiveness of COVID-19 detection from cough recordings using machine and deep learning techniques in an attempt to reduce the cost and time required to diagnose the patient. In addition, the application of methods to adapt to concept drifts due to the ever-changing characteristics of the virus is examined, with the aim of maintaining the performance of the developed models. To this end, the use of different machine learning (random forests, multi-layer perceptron) and deep learning (convolutional neural networks) methods, as well as transfer learning approaches through the exploitation of pre-trained models are explored. The development and evaluation of the models is based on the use of the Coswara dataset. To address the unbalanced nature of the dataset, techniques for synthetic data generation (SMOTE), cost-sensitive learning and classification threshold optimization are exploited. The highest performance based on the AUROC metric (80.21%) is achieved by a convolutional neural network architecture that uses the pre- trained VGG-16 as the base model. To adapt to the concept drift, the last dense layers of the model are retrained using an appropriate normalization method, which leads to an improvement of the model's performance with respect to the AUROC metric by up to 5%.

Key Words

COVID – 19, Cough classification, Concept drift adaptation, CNN, Mel – Spectrogram

Acknowledgements

First of all, I would like to thank my advisor Professor Konstantina Nikita for giving me the opportunity to work on such an interesting research topic, as well as for her support throughout this time. I would also like to express my gratitude to PhD candidate Theofanis Gantidis for his constant support and guidance throughout this journey. On a personal note, I would like to express my heartfelt appreciation to my friends, family and especially my mother for their continuous support during my academic endeavor at N.T.U.A.

Contents

Εκτεταμένη ελληνική περίληψη 16

1 Covid-19 36

1.1 Introduction 36

1.2 Transmission and Personal Infection Prevention Methods 37

1.3 Covid–19 Symptoms 39

1.4 Available Treatments...39

1.5 Motivation of the current study 41

1.5.1 Audio Signal Classification...42

1.5.2 Cough Classification..43

1.5.3 Non–stationary Data..46

2 Theoretical Background Information 48

2.1 Audio Signals 48

2.1.1 Mel-Frequency Cepstral Coefficients – MFCCs 48

2.1.2 MFCCs Deltas and Delta – deltas 48

2.1.3 Kurtosis 49

2.1.4 Root Mean Square Value - RMS 49

2.1.5 Zero-Crossing Rate – ZCR..49

2.1.6 Audio to Image Transformation - Mel-Spectrograms.....................................50

2.2 Machine Learning & Deep Learning...50

2.2.1 Decision Trees & Random Forests 51

2.2.2 Multi–Layer Perceptron..53

2.2.3 Convolutional Neural Networks – CNNs...55

2.2.4 Transfer Learning 57

2.2.5 Zero – Shot Learning 58

2.3 Concept Drift 60

2.4 Optuna Hyperparameter Optimization Framework...62

2.5 Evaluation Metrics...63

3 Data Analysis - Preprocessing - Methods Implemented 67

3.1 Coswara Dataset 67

3.1.1 Data Cleaning 69

3.1.2 Feature Extraction 73

3.1.3 Mel – Spectrograms 74

3.2 Classification Methods 74

3.2.1 Multilayer Perceptron – MLP 74

3.2.2 Random Forests 75

3.2.3 Transfer Learning 76

3.2.4 Zero – Shot Learning 82

3.3 Concept Drift Adaptation Methods 83

3.3.1 Retraining only the final dense layer 84

3.3.2 Retraining the entire model 85

3.4 Classification Threshold Optimization 85

4 Results – Discussion 86

4.1 Models trained and tested using Extracted Features 86

4.2 Models trained and tested using Mel – Spectrograms 88

4.3 Zero – Shot Learning using Mel – Spectrograms 92

4.4 Concept Drift Adaptation results 93

5 Conclusion – Feature Work 99

5.1 Conclusion 99

5.2 Future Work 101

6 Bibliography 103

List of Figures

Figure 1.1: Structure of SARS-Cov-2..37
Figure 1.2: All reported adverse effects per 1M vaccine doses [18] 41
Figure 2.1: Example of a Mel-Spectrogram 50
Figure 2.2: Types of Machine Learning 51
Figure 2.3: Decision Tree structure 52
Figure 2.4: Random Forest classifier structure 52
Figure 2.5: Multilayer Perceptron example 53
Figure 2.6: Tanh activation function...54
Figure 2.7: ReLu activation function..55
Figure 2.8: CLIP 59
Figure 2.9: Types of concept drift 61
Figure 2.10: Confusion Matrix for binary classification 64
Figure 2.11: ROC curve..65
Figure 3.1: Covid-19 status distribution of the recordings in the Coswara dataset....68
Figure 3.2: Metadata Statistics for the Coswara dataset..69
Figure 3.3: Label distribution 70
Figure 3.4: Processed metadata Statistics distribution...70
Figure 3.5: Distribution of labels over the months 71
Figure 3.6: The reconstructed Positive Cough-Shallow recording............................72
Figure 3.7: Detection of non-silent segments. The red line depicts the start of the non-silent event and the green line the end. 72
Figure 3.8: Final data split...73
Figure 3.9: CNN with VGG-16 base model and hidden GS layer 77
Figure 3.10: CNN with VGG-16 base model and input GS layer 78
Figure 3.11: CNN withResNet-50 base model and hidden GS layer 79
Figure 3.12: CNN withResNet-50 base model and input GS layer 80
Figure 3.13: CNN with Inception-ResNet-V2 base model and hidden GS layer 81
Figure 3.14: CNN with Inception-ResNet-V2 base model and input GS layer..........82
Figure 3.15: Drift data split for retraining 84

List of Tables

Table 3.1: MLP Hyperparameter Search Space..75

Table 3.2: Random Forest Hyperparameter Search Space 76

Table 3.3: CLIP Text - label pairs tested on the validation set..................................83

Table 3.4: Regularization methods 84

Table 4.1:Performance metrics for the RF and MLP models 86

Table 4.2: Performance metrics for the RF and MLP models with SMOTE..............87

Table 4.3: Performance metrics for the RF and MLP models with new thresholds...87

Table 4.4: Performance metrics for the RF and MLP models with SMOTE and new thresholds 88

Table 4.5: Performance metrics for the hidden GS - CNN models 89

Table 4.6: Performance metrics for the input GS - CNN models 90

Table 4.7: Performance metrics for the hidden GS - CNN models with new threshold 91

Table 4.8: Performance metrics for the input GS - CNN models with new threshold91

Table 4.9: CLIP performance metrics of the text-label pairs on the validation data ..92

Table 4.10: CLIP performance metrics of the text-label pairs on the test set.............93

Table 4.11: Performance metrics for the hidden GS VGG - 16 model trained for 200 epochs on the drift set 94

Table 4.12: Concept drift adaptation results from retraining the last dense layer of VGG-16 - 200 epoch - hidden GS 95

Table 4.13: Concept drift adaptation results from retraining the last two dense layers of VGG-16 - 200 epoch - hidden GS 95

Table 4.14: Concept drift adaptation results from retraining the last dense layer of VGG-16 - 200 epoch - hidden GS with new thresholds 96

Table 4.15: Concept drift adaptation results from retraining the last two dense layers of VGG-16 - 200 epoch - hidden GS with new thresholds...97

Εκτεταμένη ελληνική περίληψη

COVID – 19

Οι κορονοϊοί είναι μια μεγάλη οικογένεια μονόκλωνων, θετικού νοήματος RNA ιών με τέσσερις δομικές πρωτεΐνες που μολύνουν τον άνθρωπο και ένα ευρύ φάσμα ζώων. Μεταξύ των υπο-τύπων των κορονοϊών που μπορούν να μολύνουν τον άνθρωπο, ο παράγοντας κινδύνου ποικίλλει, καθώς προκαλούν λοιμώξεις της αναπνευστικής οδού που κυμαίνονται από ήπιες έως θανατηφόρες. Οι ήπιες ασθένειες στον άνθρωπο περιλαμβάνουν ορισμένες περιπτώσεις κοινού κρυολογήματος, ενώ θανατηφόρα περιστατικά προκαλούνται από τις λοιμώξεις του σοβαρού οξέος αναπνευστικού συνδρόμου (SARS), το οποίο ξεκίνησε στην Κίνα το 2002, και του αναπνευστικού συνδρόμου της Μέσης Ανατολής (MERS) το 2012 με ποσοστό θνησιμότητας περίπου 40% [1]. Η COVID-19, ο οποίος προκαλείται από τον νέο κορονοϊό του σοβαρού οξέος αναπνευστικού συνδρόμου 2 (SARS-CoV2), εμφανίστηκε για πρώτη φορά στη Wuhan, τη πρωτεύουσα της επαρχίας Hubei της Λαϊκής Δημοκρατίας της Κίνας, στις 27 Δεκεμβρίου 2019.

Μετάδοση και Πρόληψη

Η COVID-19 μπορεί να εξαπλωθεί μεταξύ των ατόμων με διάφορους τρόπους. Η κύρια μέθοδος μετάδοσης είναι μέσω σωματιδίων αέρα, τα οποία μπορούν να μεταδοθούν με δραστηριότητες όπως η ομιλία, ο βήχας και το φτέρνισμα. Τα σωματίδια αυτά μπορούν να παραμείνουν στον αέρα έως και τρεις ώρες και κυμαίνονται σε μέγεθος από μεγαλύτερα αναπνευστικά σταγονίδια έως μικροσκοπικά αερολύματα. Τα μολυσμένα αερολύματα ή σταγονίδια μπορούν να εισέλθουν στο αναπνευστικό σύστημα ενός ατόμου μέσω της μύτης, του στόματος ή των ματιών και να προκαλέσουν λοίμωξη. Ο ιός μπορεί να διανύσει μεγαλύτερες αποστάσεις σε πολυσύχναστα ή ανεπαρκώς αεριζόμενα εσωτερικά περιβάλλοντα, αλλά μεταδίδεται κυρίως μεταξύ ατόμων που βρίσκονται σε κοντινή απόσταση μεταξύ τους.

Είτε ένα άτομο με τον ιό έχει συμπτώματα είτε όχι, ο ιός μπορεί να μεταδοθεί από αυτό. Τα άτομα με ήπια συμπτώματα μπορούν να μεταδώσουν τη λοίμωξη σε άλλους για μεγαλύτερο χρονικό διάστημα, αλλά τα άτομα με σοβαρά συμπτώματα φαίνεται να είναι πιο μεταδοτικά λίγο πριν εμφανιστούν τα συμπτώματα [2].

Παρά τους διάφορους τρόπους μόλυνσης και το γεγονός ότι οι άνθρωποι μπορεί να έχουν μολυνθεί από τον ιό και να τον μεταδίδουν χωρίς οι ίδιοι να εμφανίζουν συμπτώματα, υπάρχουν διάφορα μέτρα που μπορούν να σταματήσουν τη μετάδοση της νόσου. Πρώτα απ' όλα, η χρήση μάσκας ήταν μία από τις πρώτες προτεινόμενες μεθόδους κατά τη διάρκεια της επιδημίας COVID-19 [3]. Επιπλέον, η βελτίωση του εξαερισμού και του φιλτραρίσματος του αέρα μπορεί να συμβάλει στην αποτροπή της συσσώρευσης σωματιδίων του ιού σε εσωτερικούς χώρους. Ορισμένες ενέργειες για την αποφυγή της υψηλής συγκέντρωσης σωματιδίων αέρα μολυσμένων με τον ιό SARS-CoV-2, όπως αναφέρεται από τα Κέντρα Ελέγχου και Πρόληψης Νοσημάτων (CDC), είναι η συχνή αλλαγή και η χρήση φίλτρων που είναι κατάλληλα τοποθετημένα και παρέχουν υψηλότερη διήθηση στο σύστημα θέρμανσης, εξαερισμού και κλιματισμού (HVAC), καθώς και το άνοιγμα των παραθύρων για να εισέρχεται όσο το δυνατόν περισσότερος εξωτερικός αέρας.

Συμπτώματα

Η συντριπτική πλειονότητα των ασθενών, σύμφωνα με μια μελέτη που διεξήχθη από τους Talukder et.al [4], παρουσιάζουν ήπια αναπνευστικά συμπτώματα. Τα πιο τυπικά συμπτώματα είναι ο πυρετός, ο ξηρός βήχας, η κόπωση και η απώλεια της γεύσης ή/και της όσφρησης- τα συμπτώματα του ανώτερου αναπνευστικού συστήματος μπορεί να περιλαμβάνουν φαρυγγαλγία, πονοκεφάλους και μυαλγίες [5]. Τα σοβαρά συμπτώματα της COVID-19 περιλαμβάνουν δύσπνοια, απώλεια λόγου ή κινητικότητας, σύγχυση και θωρακικό πόνο. Τα συμπτώματα μπορεί να εμφανιστούν 2-14 ημέρες μετά τη μόλυνση, με το μέσο χρονικό διάστημα να είναι 5-6 ημέρες.

Διαθέσιμες Θεραπείες

Τέσσερα εμβόλια που έχουν παραχθεί από διαφορετικές εταιρείες, την Pfizer/BioNTech, τη Moderna, την Johnson & Johnson/Janssen και την AstraZeneca, έχουν εγκριθεί από τον Ευρωπαϊκό Οργανισμό Φαρμάκων (EMA) και ανήκουν σε έναν από τους τρεις διαθέσιμους τύπους εμβολίων, το mRNA, τον αδενοϊό και το μη αναπαραγόμενο ιικό εμβόλιο. Τα τρία πρώτα εμβόλια αναπτύχθηκαν στις ΗΠΑ, προχωρούν σε κλινική δοκιμή φάσης 3 και χορηγούνται ενδομυϊκά (IM), ενώ το τελευταίο αναπτύχθηκε στο Ηνωμένο Βασίλειο (UK) [1].

Κίνητρο της Διπλωματικής

Τα συστήματα υγειονομικής περίθαλψης παγκοσμίως αντιμετώπισαν σημαντικές προκλήσεις λόγω της πανδημίας COVID-19. Λόγω της μεγάλης ζήτησης σε γρήγορες και εύκολα προσβάσιμες διαγνωστικές μεθόδους, οι εταιρείες ανέπτυξαν κιτ ταχείας εξέτασης που θα μπορούσαν να αγοραστούν από ιδιώτες για εξέταση στο σπίτι, παρέχοντας παράλληλα μια εναλλακτική λύση για τις δοκιμές RT-PCR στα ιατρικά εργαστήρια. Αν και οι ταχείες δοκιμές δεν είναι τόσο ακριβείς όσο οι δοκιμές RT-PCR, παράγουν αποτελέσματα σε λιγότερο από μία ώρα, ενώ οι τελευταίες μπορεί να χρειαστούν έως και δύο ημέρες. Ωστόσο, προκειμένου ένα άτομο να παρακολουθεί την κατάστασή του, τόσο σε περίπτωση έκθεσης όσο και σε περίπτωση μόλυνσης από τον ιό, πρέπει να διενεργείται σημαντικός αριθμός εξετάσεων, οι οποίες μακροπρόθεσμα είναι δαπανηρές, χρονοβόρες και επιρρεπείς σε ανακριβή αποτελέσματα λόγω της μη ορθής χρήσης του εξοπλισμού. Ως τρόπος αντιμετώπισης αυτού του προβλήματος, στη παρούσα διπλωματική εργασία, τα δεδομένα που χρησιμοποιούνται είναι ηχογραφήσεις ασθενών με COVID-19 και υγιών ατόμων από το σύνολο δεδομένων πλήθους Coswara.

Οι ασθένειες του αναπνευστικού μαζί με τα αναπνευστικά προβλήματα γίνονται όλο και πιο συχνές με την πάροδο των ετών και οι άνθρωποι πρέπει να επισκέπτονται νοσοκομεία και να εξετάζονται σωματικά από γιατρό. Για τους λόγους αυτούς, η ανίχνευση με βάση τον ήχο από τεχνικές ML και DL [6],[7], [8], [9] μπορεί να μειώσει σημαντικά το κόστος για τους ασθενείς και να εξοικονομήσει πολύτιμο χρόνο τόσο για τον ασθενή όσο και για τον επαγγελματία υγείας.

Μια ειδική κατηγορία δεδομένων ηχητικών σημάτων που χρησιμοποιούνται για ταξινόμηση είναι αυτή των καταγραφών βήχα. Ο αντίκτυπος του βήχα στο αναπνευστικό σύστημα ποικίλλει και αποτελεί κοινό σύμπτωμα σε πάνω από 100 ασθενειών και άλλων καταστάσεων ιατρικής σημασίας [10], όπως της COVID-19. Έχουν υλοποιηθεί αρκετοί αλγόριθμοι που πραγματοποιούν διάγνωση COVID-19 [11] από ηχητικές καταγραφές βήχα [12] με χρήση συνελικτικών νευρωνικών δικτύων, [13], [14], [15], [16].

Μια κυρίαρχη πρόκληση στις εφαρμογές ML και DL, όπως, η ανίχνευση βλαβών, η διάγνωση, η πρόβλεψη της εναπομένουσας ωφέλιμης ζωής σε βιομηχανικά εξαρτήματα κ.λπ. έγκειται στη μη σταθερή φύση του περιβάλλοντος συλλογής ροών δεδομένων. Οι παρεκκλίσεις εννοιών, γνωστές και ως αιτίες μη στάσιμων συμπεριφορών, οφείλονται σε φαινόμενα όπως η εποχικότητα, η υποβάθμιση αισθητήρων ή εξαρτημάτων, οι θερμικές μεταβολές και οι αλλαγές στους τρόπους λειτουργίας ή στα ενδιαφέροντα των χρηστών. Το ξέσπασμα της πανδημίας COVID-19, είναι ένα χαρακτηριστικό παράδειγμα παρεκκλίσεων δεδομένων με πλήθος ερευνών να έχουν πραγματοποιηθεί για την αντιμετώπιση του προβλήματος απόκλισης εννοιών στον COVID-19 [17], [18], [19], [20], [21], [22], [23], [24].

Θεωρητικό Υπόβαθρο

Μηχανική Μάθηση και Βαθιά Μάθηση

Το 1956, μια ομάδα επιστημόνων πληροφορικής έθεσε τα θεμέλια για την ιδέα ότι οι υπολογιστές μπορούν να μιμηθούν την ανθρώπινη σκέψη και συλλογισμό. Υποστήριξαν ότι "κάθε πτυχή της μάθησης ή οποιοδήποτε άλλο χαρακτηριστικό της νοημοσύνης [θα μπορούσε], κατ' αρχήν, να περιγραφεί με τόση ακρίβεια ώστε μια μηχανή [να] μπορεί να την προσομοιώσει". [25]. Αυτή η αρχή έγινε γνωστή ως "τεχνητή νοημοσύνη" (ΤΝ). Στην ουσία, η ΤΝ αποτελεί έναν τομέα αφιερωμένο στην αυτοματοποίηση των διανοητικών εργασιών που συνήθως εκτελούνται από τον άνθρωπο. Στο πλαίσιο αυτού του τομέα, η ML και η DL αναδύονται ως συγκεκριμένες μεθοδολογίες που αποσκοπούν στην επίτευξη αυτού του στόχου με τη διάκριση μοτίβων από τα δεδομένα για τη βελτίωση της απόδοσης σε μια ποικιλία εργασιών.

Το τυχαίο δάσος ή RF είναι επομένως μια μέθοδος συνόλου που επεκτείνει τη μεθοδολογία των δέντρων απόφασης με τη δημιουργία πολλαπλών δέντρων απόφασης. Σε αντίθεση με τη χρήση όλων των χαρακτηριστικών για την κατασκευή κάθε δέντρου απόφασης, ένα τυχαίο δάσος χρησιμοποιεί ένα υποσύνολο χαρακτηριστικών για τη δημιουργία μεμονωμένων δέντρων. Στη συνέχεια, τα δέντρα προβλέπουν συλλογικά τα αποτελέσματα της κλάσης και η πρόβλεψη της επικρατούσας κλάσης μεταξύ των δέντρων καθορίζει την ταξινόμηση του τελικού μοντέλου.

Ένα πολυεπίπεδο perceptron ή MLP, (Rosenblatt 1958), είναι ένα μοντέλο τεχνητού νευρωνικού δικτύου με πρόωση, το οποίο αποτελείται από πολλαπλά στρώματα νευρώνων που συνδέονται πλήρως με τους επόμενους νευρώνες σε κάθε στρώμα. Ένας αριθμός διασυνδεδεμένων perceptrons συνθέτουν το MLP.

Για εργασίες που αφορούν την αναγνώριση εικόνων, κάθε είσοδος σε ένα ANN με τροφοδότηση αντιστοιχεί σε ένα εικονοστοιχείο - pixel μέσα στην εικόνα. Ωστόσο, αυτή η προσέγγιση έχει ένα σημαντικό μειονέκτημα: οι διασυνδέσεις μεταξύ των κόμβων είναι ανύπαρκτες και, επομένως, χάνεται το χωρικό πλαίσιο των χαρακτηριστικών. Προκειμένου να αντιμετωπιστεί αυτός ο περιορισμός των ANN με τροφοδότηση, εισάγονται τα Νευρωνικά Δίκτυα Συνελικτικής Δικτύωσης (ΣΝΝ) ως εξειδικευμένη κατηγορία των πρώτων, ικανή να διατηρεί τη χωρική συσχέτιση μεταξύ των εικονοστοιχείων μιας εικόνας. Σε αντίθεση με τα ANN που επεξεργάζονται μεμονωμένα εικονοστοιχεία, τα ΣΝΝ επεξεργάζονται και μεταδίδουν περιοχές μιας εικόνας σε συγκεκριμένους κόμβους σε επόμενα στρώματα, διατηρώντας έτσι το χωρικό πλαίσιο από το οποίο εξήχθη το χαρακτηριστικό.

Όπως αναφέρθηκε προηγουμένως, για την εκπαίδευση ενός μοντέλου βαθιάς μάθησης απαιτούνται μεγάλα, σχολιασμένα σύνολα δεδομένων που προετοιμάζονται από κλινικούς ιατρούς ή εμπειρογνώμονες. Ειδικά σε τομείς όπως η ιατρική απεικόνιση, όπου επαρκείς ποσότητες δεδομένων δεν είναι άμεσα διαθέσιμες ή απλώς δεν υπάρχουν ακόμη (π.χ. ξέσπασμα της COVID - 19), καθίσταται αναγκαία η καθιέρωση μιας εναλλακτικής μεθόδου που απαιτεί λιγότερα ή καθόλου σχολιασμένα δεδομένα για την παροχή προβλέψεων σχετικά με τη νέα έννοια. Αυτή η συγκεκριμένη πρόκληση της εκμάθησης μιας νέας έννοιας χωρίς να λαμβάνουμε εκ των προτέρων παραδείγματα, αποφεύγοντας έτσι την απαίτηση για κοπιαστική συλλογή δεδομένων και επαγγελματικό σχολιασμό, ονομάζεται Zero-Shot Learning (ZSL) [26], [27], [28].

Το μοντέλο που χρησιμοποιείται σε αυτή την προσέγγιση ονομάζεται CLIP , το οποίο σημαίνει Contrastive Language-Image Pre-training. Το CLIP είναι ένα ΝΔ που εκπαιδεύτηκε σε 400 εκατομμύρια ζεύγη (εικόνα, κείμενο). Δεδομένης μιας εικόνας, είναι σε θέση να προβλέψει, σε φυσική γλώσσα, το πιο σχετικό απόσπασμα κειμένου χωρίς να χρειάζεται να βελτιστοποιηθεί άμεσα για την εργασία.

Παρέκκλιση Εννοιών

Ο όρος "μετατόπιση εννοιών" αναφέρεται σε απρόβλεπτες μετατοπίσεις της υποκείμενης κατανομής των δεδομένων ροής με την πάροδο του χρόνου. Ως αποτέλεσμα, οι προβλέψεις των μοντέλων που εκπαιδεύτηκαν στο παρελθόν μπορεί να γίνουν λιγότερο ακριβείς καθώς περνάει ο καιρός ή μπορεί να χαθούν ευκαιρίες βελτίωσης της ακρίβειας. Προτάθηκε αρχικά από τους Schlimmer και Granger [29] το 1986, οι οποίοι είχαν ως στόχο να επισημάνουν ότι τα θορυβώδη δεδομένα μπορεί τελικά να γίνουν μη θορυβώδης πληροφορία σε διαφορετικό χρόνο. Επομένως, τα μοντέλα μάθησης πρέπει να διαθέτουν μηχανισμούς για συνεχή διάγνωση της απόδοσης και να μπορούν να προσαρμόζονται στις αλλαγές των δεδομένων με την πάροδο του χρόνου. Η έρευνα σχετικά με την παρέκκλιση εννοιών περιλαμβάνει τη δημιουργία μεθόδων για την ανίχνευση, την κατανόηση και την προσαρμογή της παρέκκλισης. η παρέκκλιση εννοιών διακρίνεται συνήθως σε τρεις κατηγορίες:

- Εικονικές διολισθήσεις στις οποίες η κατανομή των δεδομένων εισόδου PT (X) αλλάζει με το χρόνο, ενώ η εκ των υστέρων πιθανότητα της εξόδου PT (Y|X), που αντιπροσωπεύει τη σχέση χαρτογράφησης μεταξύ XT και YT, δεν αλλάζει.
- Πραγματικές μετατοπίσεις στις οποίες η εκ των υστέρων κατανομή πιθανότητας PT (Y|X) μεταβάλλεται με την πάροδο του χρόνου, ανεξάρτητα από τις μεταβολές στο PT (X).
- Υβριδικές διολισθήσεις στις οποίες συμβαίνουν ταυτόχρονα εικονικές και πραγματικές διολισθήσεις (οι υβριδικές διολισθήσεις είναι οι πιο συνηθισμένες σε βιομηχανικές εφαρμογές).

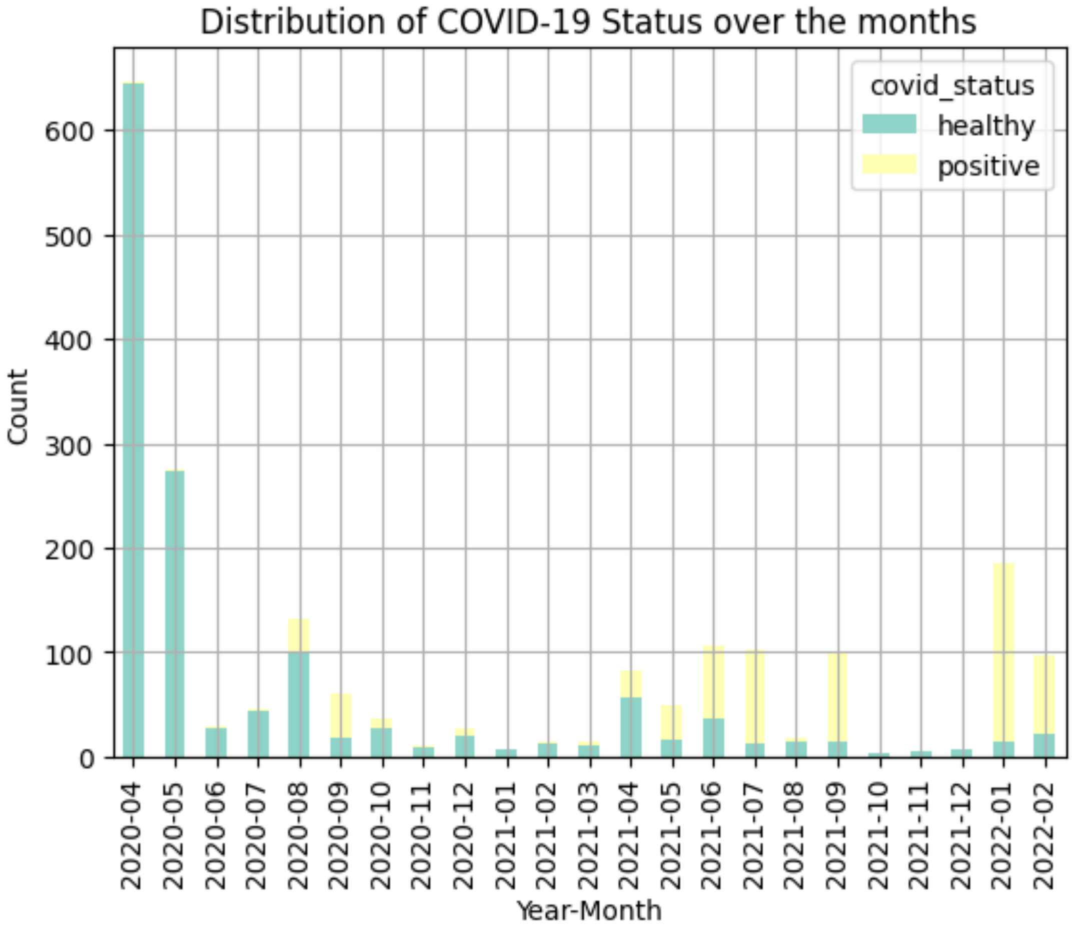
Δεδομένα – Μεθοδολογία

Δεδομένα Coswara

Το σύνολο δεδομένων Coswara [30], [31], [32], είναι ένα σύνολο δεδομένων από το πλήθος που περιέχει 3 είδη αναπνευστικών ήχων: ήχους βήχα, αναπνοής και ομιλίας, καθώς και πληροφορίες μετα-δεδομένων για κάθε χρήστη. Αποτελείται από δείγματα ήχου που παρέχονται από 2746 διαφορετικά αναγνωριστικά χρηστών (user ids). Κάθε χρήστης υπέβαλε τις ακόλουθες 9 διαφορετικές ηχογραφήσεις, δύο είδη ήχων βήχα, βαριά και ρηχά, δύο είδη ήχων αναπνοής, ρηχά και βαθιά, δύο είδη απαρίθμησης ενός έως είκοσι ψηφίων, κανονικό και γρήγορο και 3 διαφορετικούς φθόγγους φωνηέντων με παρατεταμένη φωνή. Κάθε ηχητικό δείγμα συνοδεύεται από πληροφορίες μεταδεδομένων που περιλαμβάνουν δημογραφικές πληροφορίες όπως, ηλικία, φύλο, χώρα προέλευσης κ.λπ., τύπο δοκιμής Covid-19 και την τρέχουσα κατάσταση υγείας του χρήστη. Όλα τα ηχητικά αρχεία έχουν αξιολογηθεί χειροκίνητα, όσον αφορά την ποιότητα του ηχητικού δείγματος και την κατηγορία στην οποία ανήκει, από 13 σχολιαστές με κάθε αρχείο να σχολιάζεται μία φορά.

Προ-επεξεργασία

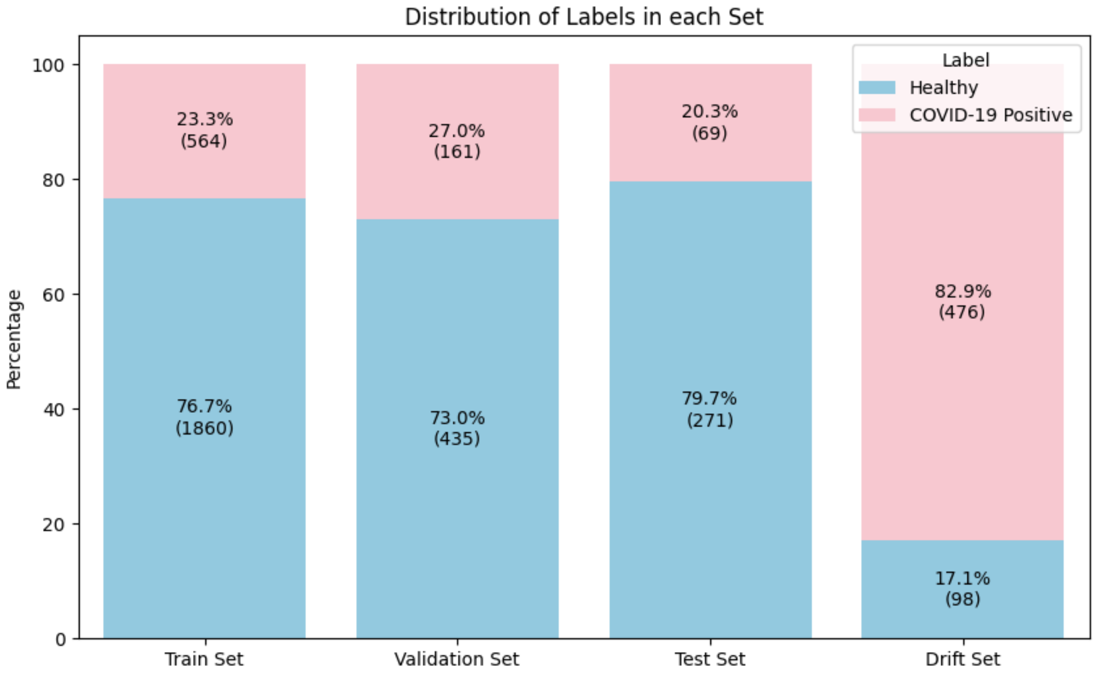
Για τους σκοπούς της παρούσας μελέτης, μόνο οι δύο τύποι δειγμάτων βήχα (βαριά, ρηχά) χρησιμοποιήθηκαν σε ένα ενιαίο σύνολο δεδομένων. Τα δείγματα με κατάσταση σε μία από τις τρεις κατηγορίες, δηλαδή θετικό ήπιο, θετικό μέτριο και θετικό ασυμπτωματικό, ταξινομούνται ως θετικά, τα δείγματα που δηλώνονται υγιή παραμένουν ως έχουν και τα υπόλοιπα δείγματα απορρίπτονται. Επιπλέον, το σχήμα 0.1 παρουσιάζει την κατανομή των συμμετεχόντων σε υγιείς - θετικές ετικέτες ανά μήνα. Προκειμένου να εφαρμοστούν οι μέθοδοι προσαρμογής της παρέκκλισης της έννοιας, τα δεδομένα που καταγράφηκαν από τον Οκτώβριο του 2021 πρόκειται να χρησιμοποιηθούν ως σύνολο παρέκκλισης και συνεπώς θα εξαιρεθούν από τα σύνολα εκπαίδευσης, επικύρωσης και δοκιμής.



Εικόνα 0.1: Κατανομή δειγμάτων ανά μήνα

Προκειμένου να μειωθεί η συνολική διάρκεια των ηχητικών δεδομένων για την εκπαίδευση, εφαρμόστηκε περικοπή της σιωπής. Αυτό επιτεύχθηκε με τον διαχωρισμό του ήχου κάθε ηχογράφησης στα μη σιωπηλά διαστήματά της, χρησιμοποιώντας ένα κατώφλι 30 dB ως κριτήριο για τη διάκριση των σιωπηλών από τα μη σιωπηλά διαστήματα, και τα υπόλοιπα τμήματα συνδέθηκαν μεταξύ τους για την ανακατασκευή της ηχογράφησης. Με τη χρήση αυτής της μεθόδου, απορρίφθηκε ο μη ουσιώδης ήχος και μειώθηκε η συνολική διάρκεια των ηχογραφήσεων. Τέλος, 66 καταγραφές απορρίφθηκαν λόγω του ότι είτε περιείχαν 0 δευτερόλεπτα ήχου (60 καταγραφές), είτε η διάρκειά τους ήταν μικρότερη από 0,35 δευτερόλεπτα με άσχετο ήχο (6 καταγραφές που περιείχαν φωνές ή δυσδιάκριτους ήχους).

Τα εναπομείναντα δεδομένα διαχωρίστηκαν στη συνέχεια σε 4 σύνολα, δηλαδή το σύνολο εκπαίδευσης, το σύνολο επικύρωσης, το σύνολο δοκιμής και το σύνολο παρέκκλισης. Η κατανομή των δεδομένων που προέκυψε παρουσιάζεται στο ακόλουθο σχήμα 0.2.



Εικόνα 0.2: Τελικός διαχωρισμός δεδομένων

Εξαγωγή Χαρακτηριστικών

Προκειμένου να αντιμετωπιστεί το πρόβλημα των ηχογραφήσεων με διαφορετικά μήκη διάρκειας, χρησιμοποιήθηκε η μέθοδος που προτάθηκε από τους M. Pahar et.al [16]. Συγκεκριμένα, από κάθε εγγραφή εξάγεται ένας σταθερός αριθμός χαρακτηριστικών F με την εφαρμογή του μήκους άλματος να εξαρτάται από το μήκος της ηχητικής χρονοσειράς που εξάγεται κατά τη φόρτωση της εγγραφής και τα δείγματα ανά τμήμα να εξαρτώνται από τη διάρκεια του ήχου. Με την εφαρμογή της προαναφερθείσας μεθόδου, ο αριθμός των διανυσμάτων mfcc ανά τμήμα θα είναι ο ίδιος και δεν απαιτείται συμπλήρωση με μηδενικά. Ο αριθμός των τμημάτων ορίστηκε σε 100, η παράμετρος n_fft (μήκος του παραθυρικού σήματος μετά τη συμπλήρωση με μηδενικά) ορίστηκε σε 1024 και ο ρυθμός δειγματοληψίας ορίστηκε σε 2250. Συνολικά εξήχθησαν 42 χαρακτηριστικά, 13 MFCC, 13 MFCC Deltas, 13 MFCC Delta - deltas, 1 ZCR, 1 Kurtosis και 1 RMS για κάθε ένα από τα 100 τμήματα, τα οποία αθροίζονται σε ένα σχήμα χαρακτηριστικών (42, 100) για κάθε καταγραφή.

Φασματογραφήματα Mel

Για τα μοντέλα βαθιάς μάθησης, εξήχθησαν τα Mel-φασματογραφήματα χρησιμοποιώντας τη συνάρτηση librosa.feature.melspectrogram. Ο ρυθμός δειγματοληψίας ορίστηκε σε 22050, το n_fft σε 1024, ο αριθμός των τμημάτων σε 100 και τέλος, ο τύπος για το μήκος άλματος είναι ο ίδιος με τον παραπάνω.

MLP και RF μοντέλα

Πρώτον, τα δεδομένα τυποποιούνται με τη χρήση του Standard Scaler από τη βιβλιοθήκη sickit learn. τυποποιεί τα χαρακτηριστικά αφαιρώντας τη μέση τιμή και κλιμακώνοντας στη μοναδιαία διακύμανση.

Για αυτά τα μοντέλα χρησιμοποιήθηκε το πλαίσιο βελτιστοποίησης υπερπαραμέτρων Optuna προκειμένου να βρεθούν οι καλύτερες υπερπαράμετροι. Η πρόωρη διακοπή έχει οριστεί σε True και η συνάρτηση επιστρέφει το σκορ auroc του συνόλου επικύρωσης. Η μελέτη έχει οριστεί να εκτελείται για 70 δοκιμές με κατεύθυνση τη μεγιστοποίηση της τιμής απόδοσης της αντικειμενικής συνάρτησης. Επιπλέον, αξιοποιούνται ο δειγματολήπτης TPE και ο κλαδευτής Hyperband [33]. Τέλος, το μοντέλο με την υψηλότερη βαθμολογία auroc στο σύνολο επικύρωσης επιλέγεται ως το τελικό μοντέλο που θα δοκιμαστεί στο σύνολο δοκιμής.

Η διαδικασία επαναλαμβάνεται για την εκτέλεση άλλων 70 δοκιμών, αλλά αυτή τη φορά εφαρμόζεται η τεχνική Synthetic Minority Oversampling ή SMOTE. Αυτή η τεχνική υπέρ-δειγματοληψίας χρησιμοποιείται για να αντιμετωπίσει την έλλειψη δειγμάτων Covid-19 (κλάση μειονότητας) και έτσι να βοηθήσει το μοντέλο στην αποτελεσματική εκμάθηση του ορίου απόφασης μεταξύ των δύο κλάσεων. Συγκεκριμένα, η SMOTE λειτουργεί με την επιλογή ενός τυχαίου παραδείγματος από την κλάση μειονότητας και στη συνέχεια βρίσκονται k από τους πλησιέστερους γείτονες για το εν λόγω παράδειγμα. Επιλέγεται ένας τυχαία επιλεγμένος γείτονας και δημιουργείται ένα συνθετικό παράδειγμα σε ένα τυχαία επιλεγμένο σημείο μεταξύ των δύο παραδειγμάτων στο χώρο χαρακτηριστικών. Στο τέλος αυτής της διαδικασίας τα δεδομένα εκπαίδευσης θα είναι ισορροπημένα. Η μεταβλητή k ορίζεται σε 5.

Συνελικτικά Νευρωνικά Δίκτυα

Λόγω της πληθώρας προ-εκπαιδευμένων ΣΝΝ που έχουν επιτύχει εξαιρετικά αποτελέσματα στο σύνολο δεδομένων ImageNet [34], αποφασίστηκε ότι η γνώση που απέκτησαν θα μπορούσε να αξιοποιηθεί για να βελτιωθεί η γενίκευση για την τρέχουσα εργασία. Συνολικά θα δημιουργηθούν 6 διαφορετικά μοντέλα χρησιμοποιώντας 3 προ-εκπαιδευμένα ΣΝΝ ως μοντέλα βάσης. Τρία από τα έξι μοντέλα θα έχουν ένα στρώμα Gaussian Noise ως κρυφό στρώμα στην αρχιτεκτονική, ενώ για τα υπόλοιπα το GS θα χρησιμοποιηθεί ως στρώμα εισόδου. Με αυτόν τον τρόπο μπορούμε να συγκρίνουμε πώς η θέση του στρώματος GS επηρεάζει την απόδοση του μοντέλου.

Τα μοντέλα που χρησιμοποιήθηκαν ως μοντέλα βάσης για τον ταξινομητή είναι τα VGG-16, ResNet-50 και Inception-ResNet-V2 από τη βιβλιοθήκη εφαρμογών keras. Το μοντέλο φορτώνεται με τα προ-εκπαιδευμένα βάρη από την εργασία ταξινόμησης ImageNet, αλλά χωρίς το ανώτερο στρώμα. Όλα τα στρώματα είναι παγωμένα, ώστε να μην είναι εκπαιδεύσιμα και για να μην καταστραφούν οι πληροφορίες που περιέχουν κατά τη διάρκεια μελλοντικών γύρων εκπαίδευσης. Το μοντέλο θα μάθει να παρέχει προβλέψεις για την τρέχουσα εργασία προσθέτοντας μερικά εκπαιδεύσιμα στρώματα πάνω από τα παγωμένα στρώματα. Σε αυτή την περίπτωση τα στρώματα που προστίθενται είναι ένα στρώμα Gaussian Noise με τιμή τυπικής απόκλισης της κατανομής του θορύβου που έχει οριστεί στο 0,1. Στη συνέχεια προστέθηκε ένα στρώμα Global Average Pooling 2D, με 2 Dense στρώματα να ολοκληρώνουν το μοντέλο. Το πρώτο Dense layer αποτελείται από 1024 μονάδες και μια συνάρτηση ενεργοποίησης "ReLu", ενώ το δεύτερο αποτελείται από 2 μονάδες (μία για κάθε κλάση) και τη συνάρτηση ενεργοποίησης "softmax". Το μοντέλο συντάσσεται με τη χρήση του βελτιστοποιητή "Adam" με ρυθμό μάθησης 0,001 και binary cross entropy loss. Τα δεδομένα αναβαθμίζονται χρησιμοποιώντας το "ImageDataGenerator" και θέτοντας την παράμετρο rescale σε 1.0/255.0. Τέλος, το μοντέλο προσαρμόζεται για 100 και 200 εποχές, με τα βάρη των κλάσεων να έχουν οριστεί ως προς την αναλογία των υγιών προς τα θετικά δείγματα, προκειμένου να αντιμετωπιστεί το πρόβλημα της ανισορροπίας των κλάσεων.

Η ίδια διαδικασία επαναλαμβάνεται με τη διαφορά ότι το στρώμα GS ορίζεται τώρα ως στρώμα εισόδου αντί για το εκάστοτε μοντέλο βάσης.

Zero – Shot Learning

Το πλαίσιο ανοικτού κώδικα OpenAI CLIP (Contrastive Language-Image Pretraining) και η βιβλιοθήκη PyTorch χρησιμοποιούνται για αυτό το τμήμα. Επιλέγεται ένα προ-εκπαιδευμένο μοντέλο με χρήση του OpenAI CLIP με την αρχιτεκτονική ViT-B-32 και καθορίζονται τα προ-εκπαιδευμένα βάρη ("laion2b_s34b_b79k"). Επιπλέον, ανακτάται ο tokenizer που σχετίζεται με το επιλεγμένο μοντέλο για την επεξεργασία των εισόδων κειμένου. Το προς δοκιμή ζεύγος κειμένου-ετικέτας συμβολικοποιείται και στη συνέχεια εξάγονται τα χαρακτηριστικά εικόνας και κειμένου από το μοντέλο. Στη συνέχεια, υπολογίζονται οι softmax πιθανότητες το κείμενο να συσχετίζεται με την εικόνα και η προβλεπόμενη κλάση καθορίζεται από τον δείκτη της μέγιστης πιθανότητας. Τέλος, η απόδοση του μοντέλου σε σχέση με τις ετικέτες της βασικής αλήθειας αξιολογείται με τον υπολογισμό των απαραίτητων μετρικών. Τα δύο ζεύγη με την καλύτερη απόδοση θα χρησιμοποιηθούν στο σύνολο δοκιμής. Βελτιστοποίηση κατωφλίου ταξινόμησης

Τεχνικές Αντιμετώπισης Παρέκκλισης Έννοιας

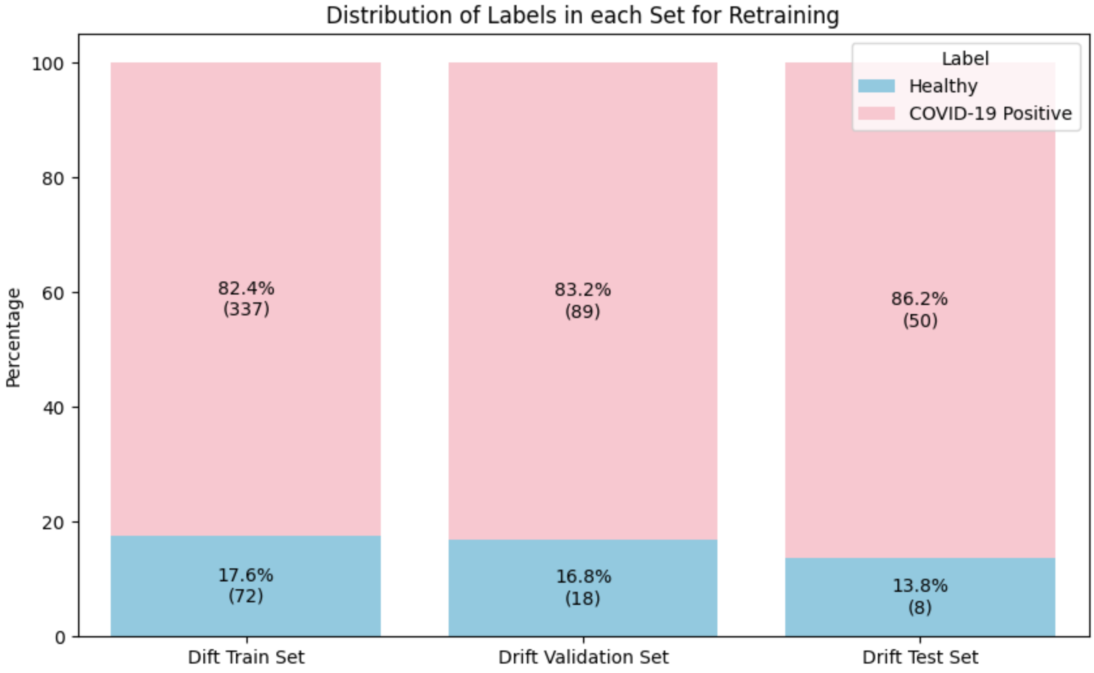
Το μοντέλο με την υψηλότερη βαθμολογία roc-auc στο σύνολο δοκιμής θα επιλεγεί για να δοκιμαστεί στο σύνολο παρέκκλισης, προκειμένου να διαπιστωθεί αν υπάρχει παρέκκλιση έννοιας ή όχι. Στη συνέχεια, το σύνολο παρέκκλισης χωρίζεται στα σύνολα εκπαίδευσης drift, επικύρωσης drift και δοκιμής drift, με το επιλεγμένο μοντέλο να επανεκπαιδεύεται στο σύνολο εκπαίδευσης drift. Η κατανομή των δεδομένων που προκύπτει παρουσιάζεται στο σχήμα 1.3.

Κατά τη διάρκεια της επανεκπαίδευσης θα εξεταστεί η επανεκπαίδευση του τελευταίου και των δύο τελευταίων πυκνών στρωμάτων, με την προσθήκη ενός παράγοντα κανονικοποίησης στα βάρη του μοντέλου. Ο παράγοντας κανονικοποίησης θα είναι η διαφορά των νέων βαρών μείον τα παλιά βάρη του επιπέδου (σε απόλυτο ή τετραγωνικό μέγεθος) και θα πολλαπλασιάζεται με μια σταθερά με τιμές 0,01, 0,1 ή 1,0.

Διακρίνουμε τις δύο επιμέρους τεχνικές:

- Όλα τα στρώματα εκτός από το τελικό πυκνό στρώμα (dense_1) παγώνουν και διερευνώνται οι ακόλουθες μέθοδοι κανονικοποίησης επιπέδου (LRM) χρησιμοποιώντας τα βάρη του στρώματος, τα οποία είναι αρχικά αποθηκευμένα ως παλιά βάρη. Το μοντέλο επανεκπαιδεύεται για 100 και 200 εποχές.

- Τα δύο τελευταία πυκνά στρώματα του επιλεγμένου μοντέλου (dense και dense_1) επανεκπαιδεύονται με την εφαρμογή των μεθόδων κανονικοποίησης που παρουσιάζονται στον πίνακα 1.1 σε κάθε στρώμα. Το μοντέλο επανεκπαιδεύεται για 100 και 200 εποχές.



Εικόνα 0.3: Διαχωρισμός δεδομένων παρέκκλισης για επανεκπαίδευση

Πίνακας 0.1: Μέθοδοι κανονικοποίησης επιπέδου

LRM	Equation
1	$new_{weights}[0] = 1.0 \times \lvert new_{weights}[0] - old_{weights}[0] \rvert$
2	$new_{weights}[0] = 0.1 \times \lvert new_{weights}[0] - old_{weights}[0] \rvert$
3	$new_{weights}[0] = 0.01 \times \lvert new_{weights}[0] - old_{weights}[0] \rvert$
4	$new_{weights}[0] = 1.0 \times \left(new_{weights}[0] - old_{weights}[0]\right)^2$
5	$new_{weights}[0] = 0.1 \times \left(new_{weights}[0] - old_{weights}[0]\right)^2$
6	$new_{weights}[0] = 0.01 \times \left(new_{weights}[0] - old_{weights}[0]\right)^2$

Βελτιστοποίηση Κατωφλίου Ταξινόμησης

Ένα μοντέλο που εκπαιδεύεται για ένα έργο δυαδικής ταξινόμησης, όπως αυτά που αναφέρθηκαν παραπάνω, επιστρέφει ένα σκορ πιθανότητας της μεταβλητής-στόχου που δείχνει πόσο πιθανό είναι να ανήκει το δείγμα σε κάθε κλάση. Το τυπικό κατώφλι που χρησιμοποιείται για να καθοριστεί αν το δείγμα ανήκει ή όχι στη θετική κλάση είναι 0,5. Μεταβάλλοντας το κατώφλι ταξινόμησης, αλλάζει η απόδοση του ταξινομητή, αφού αλλάζουν και οι τιμές των TP, TN, FP, FN που εμφανίζονται στον πίνακα σύγχυσης.

Δεδομένου ότι η τιμή 0,5 δεν είναι πάντα το ιδανικό κατώφλι, και εκτός από το γεγονός ότι το υπό εξέταση έργο είναι αυτό της ανισόρροπης ταξινόμησης, το τελευταίο βήμα στη διαδικασία ταξινόμησης θα είναι να εξεταστεί κατά πόσον η βελτιστοποίηση του κατωφλίου ταξινόμησης θα βελτιώσει την απόδοση των ταξινομητών. Αυτό επιτυγχάνεται με τον υπολογισμό των κατωφλίων της καμπύλης ROC για το σύνολο επικύρωσης και στη συνέχεια με την αξιολόγηση της απόδοσης του μοντέλου σε κάθε κατώφλι με βάση τη μέθοδο βαθμολόγησης ισορροπημένης ακρίβειας. Τέλος, το κατώφλι που μεγιστοποιεί την ισορροπημένη βαθμολογία ακρίβειας επιλέγεται ως το νέο κατώφλι απόφασης και το μοντέλο δοκιμάζεται στο σύνολο δοκιμής.

Αποτελέσματα – Σχολιασμός

Παρουσιάζονται τα μοντέλα με τα καλύτερα αποτελέσματα ανά κατηγορία.

Αποτελέσματα MLP και RF

Όταν χρησιμοποιούνται τόσο η βελτιστοποίηση SMOTE όσο και η βελτιστοποίηση κατωφλίου, παρατηρείται σημαντική βελτίωση τόσο στις τιμές ευαισθησίας όσο και στις τιμές ισορροπημένης ακρίβειας. Για να είμαστε πιο ακριβείς, το μοντέλο RF επιτυγχάνει βαθμολογία ευαισθησίας 59,42% και ισορροπημένη ακρίβεια 62,92%, με τις δύο τιμές να αποτελούν τις υψηλότερες επιτευχθείσες βαθμολογίες σε αυτές τις δύο μετρικές. Παρόλο που η βαθμολογία AUROC είναι οριακά χαμηλότερη από εκείνη που επιτυγχάνεται χωρίς τη χρήση του SMOTE, αυτό το μοντέλο RF είναι πολύ καλύτερο στην ορθή πρόβλεψη της θετικής κλάσης (Covid-19). Η ευαισθησία βελτιώθηκε περισσότερο από 10 φορές και η βαθμολογία ισορροπημένης ακρίβειας είναι επίσης 21,33% υψηλότερη, σε σύγκριση με το μοντέλο RF που δεν χρησιμοποιεί ούτε

SMOTE ούτε τη βελτιστοποίηση κατωφλίου, και 10,70% υψηλότερη από το μοντέλο που χρησιμοποιεί μόνο το SMOTE.

Πίνακας 0.2: Αποτελέσματα μοντέλων RF και MLP με εφαρμογή SMOTE και βέλτιστου κατωφλιού ταξινόμησης

Model	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC	Threshold
MLP	74.71%	40.58%	83.39%	61.99%	65.13%	0.674
RF	**65.00%**	**59.42%**	**66.42%**	**62.92%**	**69.42%**	**0.387**

Αποτελέσματα ΣΝΝ

Ο πίνακας 1.3 απεικονίζει τις μετρικές απόδοσης για τα κρυφά μοντέλα GS - CNN με τα νέα βέλτιστα κατώφλια που εφαρμόζονται στον υπολογισμό των μετρικών. Οι ισορροπημένες τιμές ακρίβειας κυμαίνονται μεταξύ 65,48% και 72,07%. Αυτό υποδηλώνει μια καλή διάκριση μεταξύ της θετικής και της αρνητικής κατηγορίας, με μια μικρή πτώση να είναι εμφανής στις ελάχιστες και μέγιστες τιμές. Επιπλέον, η τιμή ευαισθησίας κυμαίνεται από 62,32% έως 69,57% ανάλογα με το βασικό μοντέλο και τον αριθμό των εποχών που εκπαιδεύονται. Όσον αφορά το μοντέλο με την υψηλότερη βαθμολογία AUROC, παρατηρείται βελτίωση της τιμής ευαισθησίας κατά 9,31%. Συνολικά, η εφαρμογή ενός βέλτιστου κατωφλίου που προκύπτει από το σύνολο επικύρωσης φαίνεται να έχει διαφορετική επίδραση στις τιμές των μετρικών, με μια γενικότερη τάση στην βελτίωση της ευαισθησίας.

Πίνακας 0.3: Αποτελέσματα μοντέλων ΣΝΝ με κρυφό στρώμα GS και βελτιστοποιημένο κατώφλι ταξινόμησης

Base – Model	Epoch	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC	Threshold
VGG – 16	100	74.12%	62.32%	77.12%	69.72%	76.65%	0.827
	200	**74.41%**	**68.12%**	**76.01%**	**72.07%**	**80.21%**	**0.860**
ResNet	100	67.35%	62.32%	68.63%	65.48%	71.34%	0.568
	200	70.00%	63.77%	71.59%	67.68%	73.93%	0.742
Inception	100	70.29%	63.77%	71.96%	67.86%	77.14%	0.014
	200	**71.18%**	**69.57%**	**71.59%**	**70.58%**	**78.74%**	**0.003**

Αποτελέσματα CLIP

Τα ζεύγη 2 και 7 επιλέχθηκαν λόγω της απόδοσής τους στη μετρική AUROC και εξετάστηκαν στο σύνολο δοκιμών. Από τα αποτελέσματα που παρουσιάζονται στον πίνακα 1.4 μπορεί να διαπιστωθεί ότι υπάρχει πτώση της απόδοσης όλων των μετρικών σε σύγκριση με τις τιμές στο σύνολο επικύρωσης για το ζεύγος 2. Το ζεύγος 7 επιτυγχάνει ελαφρώς καλύτερη βαθμολογία AUROC (53,36%) και τιμή ευαισθησίας 15,94%. Ενώ το ζεύγος 7 επιτυγχάνει καλύτερες μετρήσεις στο σύνολο δοκιμών από το ζεύγος 2, εξακολουθεί να έχει πολύ κακές επιδόσεις στη διάγνωση του Covid - 19 από τα φασματογραφήματα Mel.

Πίνακας 0.4: Αποτελέσματα μοντέλου CLIP

Pair index	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC
2	63.82%	**18.84%**	75.28%	47.06%	47.06%
7	**75.59%**	**15.94%**	**90.77%**	**53.36%**	**53.36%**

Αποτελέσματα Αντιμετώπισης Παρέκκλισης Έννοιας

Μετά την εφαρμογή καθεμιάς από τις 6 μεθόδους κανονικοποίησης επιπέδου, φαίνεται ότι η LRM 6 επιτυγχάνει τιμή AUROC 78,50% μετά από 100 εποχές εκπαίδευσης και 78,25% μετά από 200 εποχές εκπαίδευσης. Αυτό καθιερώνει τη μέθοδο κανονικοποίησης του τελευταίου επιπέδου με 100 εποχές εκπαίδευσης ως τη μέθοδο με τις καλύτερες επιδόσεις. Συγκεκριμένα, σε σύγκριση με την μη εφαρμογή μεθόδου κανονικοίησης επιπέδου (LRM 0) με 100 εποχές εκπαίδευσης, η βαθμολογία AUROC βελτιώνεται κατά 5,02% και κατά 3,98% σε σύγκριση με την LRM 0 με 200 εποχές εκπαίδευσης.

Ο Πίνακας 1.5 απεικονίζει υψηλότερη ευαισθησία αλλά ελαφρώς χαμηλότερη ειδικότητα λόγω εφαρμογής βέλτιστου κατωφλίου. Οι σημαντικές βελτιώσεις στην ευαισθησία, υποδηλώνουν καλύτερη ανίχνευση θετικών περιπτώσεων με την εφαρμογή των νέων κατωφλίων. Το LRM 6 παρουσιάζει μια μικρή μείωση της ευαισθησίας και της ισορροπημένης ακρίβειας στις 200 εποχές, ενώ στις 100 εποχές οι μετρικές είναι πανομοιότυπες. Για άλλη μια φορά, το αποτέλεσμα της βελτιστοποίησης του κατωφλίου ταξινόμησης βελτιώνει την ευαισθησία, ενώ μειώνει την ειδικότητα. Η μετρική ισορροπημένης ακρίβειας ποικίλλει ανάλογα με το μοντέλο.

Πίνακας 0.5: Αποτελέσματα επανεκπαίδευσης των δύο τελευταίων πυκνών στρωμάτων του VGG-16 - 200 εποχές - κρυμμένου GS με εφαρμογή βέλτιστου κατωφλιού ταξινόμησης.

LRM	Epoch	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC	Threshold
0	100	81.03%	88.00%	37.50%	62.75%	74.75%	0.171
	200	82.76%	90.00%	37.50%	63.75%	75.50%	0.957
1	100	84.48%	92.00%	37.50%	64.75%	75.50%	0.55
	200	86.21%	94.00%	37.50%	65.75%	74.50%	0.004
2	100	75.86%	82.00%	37.50%	59.75%	73.25%	0.476
	200	82.76%	90.00%	37.50%	63.75%	74.25%	0.138
3	100	87.93%	96.00%	37.50%	66.75%	77.75%	0.023
	200	87.93%	96.00%	37.50%	66.75%	76.00%	0.002
4	100	82.76%	90.00%	37.50%	63.75%	74.75%	0.052
	200	82.76%	88.00%	50.00%	69.00%	73.37%	0.255
5	100	74.14%	78.00%	50.00%	64.00%	74.00%	0.788
	200	82.76%	90.00%	37.50%	63.75%	74.25%	0.041
6	**100**	**79.31%**	**84.00%**	**50.00%**	**67.00%**	**78.50%**	**0.751**
	200	**75.86%**	**80.00%**	**50.00%**	**65.00%**	**78.25%**	**0.706**

Συμπεράσματα – Μελλοντική Έρευνα

Στόχος της παρούσας διπλωματικής εργασίας είναι: (1) η ανάπτυξη μεθόδων Μηχανικής Μάθησης και Βαθιάς Μάθησης για τη διάγνωση του COVID-19 από ηχητικά δεδομένα και (2) η εφαρμογή μεθόδων προσαρμογής ολίσθησης που θα διατηρήσουν την ακρίβεια του μοντέλου που αναπτύχθηκε, σε μη σταθερά περιβάλλοντα, σε όλη τη διάρκεια του χρόνου. Για την προσπάθεια αυτή, δοκιμάστηκαν πολλαπλά μοντέλα χρησιμοποιώντας τόσο παραδοσιακές υλοποιήσεις, δηλαδή Random Forests και Multilayer Perceptron, όσο και αρχιτεκτονικές CNN.

Προκειμένου να αντιμετωπιστεί το πρόβλημα που απορρέει από τα περιορισμένα διαθέσιμα δεδομένα, χρησιμοποιήθηκε η τεχνική Transfer Learning μέσω της χρήσης των μοντέλων VGG-16, ResNet-50 και Inception-ResNet-V2, τα οποία αφορούσαν το σύνολο δεδομένων ImageNet. Επίσης, τα μοντέλα RF και MLP εκπαιδεύτηκαν με και χωρίς την εφαρμογή του SMOTE. Επιπλέον, στην παρούσα μελέτη χρησιμοποιήθηκε η μάθηση Zero-Shot Learning, με τη χρήση του μοντέλου OpenAI CLIP, για να εξεταστεί η απόδοση αυτής της μεθόδου μάθησης σε ένα έργο ιατρικής ταξινόμησης και να συγκριθούν τα αποτελέσματά της με μοντέλα που έχουν εκπαιδευτεί ή ρυθμιστεί ειδικά για αυτό το έργο.

Τα δεδομένα που χρησιμοποιήθηκαν ήταν δείγματα βήχα από το σύνολο δεδομένων Coswara που περιείχε υγιή και μολυσμένα με COVID-19 άτομα. Οι μετασχηματισμοί δεδομένων που πραγματοποιήθηκαν ήταν τόσο η εξαγωγή χαρακτηριστικών από τις ηχογραφήσεις όσο και οι μετασχηματισμοί ήχου σε εικόνα. Το καλύτερο μοντέλο που επιτεύχθηκε από τη χρήση των εξαχθέντων χαρακτηριστικών ως είσοδο ήταν το μοντέλο τυχαίου δάσους με κατώφλι 0,367, το οποίο οδήγησε σε ακρίβεια 66,47%, ευαισθησία 47,83%, ειδικότητα 71,22%, ισορροπημένη ακρίβεια 59,52% και βαθμολογία AUROC 69,91%. Η εφαρμογή του SMOTE στα δεδομένα εκπαίδευσης είχε ως αποτέλεσμα το μοντέλο τυχαίου δάσους, με τιμή κατωφλίου 0,387, να έχει την καλύτερη απόδοση. Το μοντέλο πέτυχε ακρίβεια 65,00%, ευαισθησία 59,42%, ειδικότητα 66,42%, ισορροπημένη ακρίβεια 62,92% και βαθμολογία AUROC 69,42%. Η εφαρμογή του SMOTE και η βελτιστοποίηση κατωφλίου από το σύνολο επικύρωσης βελτίωσαν τις μετρικές των μοντέλων, αν και με οριακή μείωση του AUROC.

Όσον αφορά τα μοντέλα CNN που δοκιμάστηκαν με τη χρήση Mel - Spectrograms η καλύτερη απόδοση επιτεύχθηκε από την αρχιτεκτονική που χρησιμοποιεί το VGG - 16 ως βασικό - μοντέλο και ένα κρυφό στρώμα Gaussian Noise. Τα αποτελέσματα είναι τιμή ακρίβειας 78,53%, ευαισθησία 62,32%, ειδικότητα 82,66%, ισορροπημένη ακρίβεια 72,49% και τιμή AUROC 80,21%. Η χρήση ενός στρώματος Gaussian Noise ως κρυφό στρώμα οδηγεί σε υψηλότερες τιμές AUROC (13% βελτίωση στα ΣΝΝ με μοντέλο βάσης το VGG-16 και Inception-ResNet-V2) και ευαισθησίας σε σύγκριση με τη χρήση του GS ως στρώμα εισόδου. Η χρήση βελτιστοποίησης κατωφλίου ταξινόμησης βελτίωσε τη βαθμολογία ευαισθησίας του μοντέλου, ενώ μείωσε οριακά τη βαθμολογία ισορροπημένης ακρίβειας. Οι βαθμολογίες που προέκυψαν με τη χρήση κατωφλίου 0,860 ήταν 74,41% ακρίβεια, ευαισθησία 68,12%, ειδικότητα 76,01%, ισορροπημένη ακρίβεια 72,07% και τιμή AUROC 80,21%. Συνολικά, η βελτιστοποίηση του κατωφλίου απόφασης δεν πέτυχε την ίδια συνολική βελτίωση της μετρικής όπως στα μοντέλα MLP και RF. Στην περίπτωση των μοντέλων CNN φαίνεται να υπάρχει ένας συμβιβασμός μεταξύ ευαισθησίας και ειδικότητας, αλλά λόγω του πεδίου εφαρμογής των μοντέλων μας ευνοείται μια βελτιωμένη ευαισθησία έναντι μιας αυξημένης ειδικότητας.

Τέλος, η ταξινόμηση μηδενικών - πυροβολισμών με χρήση του μοντέλου CLIP έχει τη χειρότερη επίδοση συνολικά στο σύνολο δοκιμών, με σημαντική διαφορά μεταξύ του ζευγαριού κειμένου - ετικέτας με την καλύτερη επίδοση του CLIP και του μοντέλου με τη χειρότερη επίδοση του πλήρως εκπαιδευμένου μοντέλου.

Όσον αφορά την προσαρμογή στην παρέκκλιση των εννοιών, η επανεκπαίδευση των δύο τελευταίων πυκνών στρωμάτων του μοντέλου με τη μέθοδο κανονικοποίησης επιπέδων 6 έδειξε βελτίωση κατά 5% στην τιμή AUROC (σε σύγκριση με τη μη χρήση μεθόδου κανονικοποίησης επιπέδων) και χρειάστηκαν 100 εποχές επανεκπαίδευσης στα δεδομένα εκπαίδευσης παρέκκλισης για να επιτευχθεί αυτό το αποτέλεσμα, αντί για τις 200 εποχές που χρειάστηκαν για την αρχική εκπαίδευση του μοντέλου. Τα αποτελέσματα που επιτεύχθηκαν ήταν ακρίβεια 79,31%, ευαισθησία 84,00%, ειδικότητα 50,00%, ισορροπημένη ακρίβεια 67,00% και τιμή AUROC 78,50%. Η χρήση ενός βελτιστοποιημένου μοντέλου κατωφλίου απόφασης δεν άλλαξε τις τιμές του συγκεκριμένου μοντέλου, αλλά συνολικά, παρουσιάστηκε μια αντιστάθμιση μεταξύ ευαισθησίας και ειδικότητας, όπου η ευαισθησία αυξήθηκε και η ειδικότητα μειώθηκε. Η συμπεριφορά της ισορροπημένης ακρίβειας διέφερε ανάλογα με το LRM που χρησιμοποιήθηκε.

Η μελλοντική έρευνα μπορεί να συμπεριλάβει το συνδυασμό των καταγραφών βήχα, φωνής και ομιλίας που παρέχονται στο σύνολο δεδομένων της Coswara. Η χρήση διαφορετικών πηγών αναπνευστικών ήχων θα μπορούσε να βελτιώσει την απόδοση του ταξινομητή, δεδομένου ότι το CNN θα έχει μεγαλύτερη ποικιλία πιθανών χαρακτηριστικών για να εξάγει. Μια αποτελεσματική εφαρμογή αυτής της μεθόδου θα μπορούσε να παράγει ένα μοντέλο με καλύτερες ικανότητες διάκρισης. Επιπλέον, η προοπτική της πολυτροπικής ταξινόμησης που συνδυάζει τα φασματογραφήματα Mel - με τα δεδομένα κειμένου που παρέχονται από τους χρήστες (φύλο, ηλικία, συμπτώματα κ.λπ.) θα μπορούσε ενδεχομένως να αυξήσει την απόδοση του μοντέλου, αξιοποιώντας παράλληλα τις ήδη παρεχόμενες πληροφορίες των ασθενών που έμειναν αχρησιμοποίητες στην παρούσα μελέτη. Επιπλέον, η χρήση μεθόδων συνόλου, όπως η συνάθροιση ή η στοίβαξη μοντέλων, θα μπορούσε να αθροίσει τα πλεονεκτήματα πολλαπλών ταξινομητών, ενισχύοντας τη συνολική απόδοση και τη διαγνωστική ακρίβεια. Επιπλέον, η διεξαγωγή πολλαπλών εκπαιδεύσεων σε διαφορετικά σύνολα δεδομένων για το βήχα COVID-19, συμπεριλαμβανομένων πόρων όπως το COUGHVID και το Sarcos, θα μπορούσε να παρέχει πλουσιότερα δεδομένα εκπαίδευσης, ενισχύοντας την ικανότητα του μοντέλου να γενικεύει σε διαφορετικούς πληθυσμούς και καταστάσεις.

Τέλος, η προσαρμογή της παρέκκλισης των εννοιών σε συνδυασμό με μεθόδους ανίχνευσης θα μπορούσε να διασφαλίσει την ανθεκτικότητα του μοντέλου με την πάροδο του χρόνου, επιτρέποντας στο σύστημα να ανιχνεύει και να προσαρμόζεται στις εξελισσόμενες κατανομές δεδομένων με ελάχιστη εξωτερική αλληλεπίδραση. Η αξιοποίηση και ο συνδυασμός αυτών των τεχνικών θα μπορούσε να ανοίξει το δρόμο για αποτελεσματικότερα διαγνωστικά συστήματα COVID-19 και ανακουφίζοντας τους οργανισμούς υγειονομικής περίθαλψης από την πίεση.

1

Covid-19

1.1 Introduction

Coronaviruses are a large family of single-stranded, positive-sense RNA viruses with four structural proteins, as seen in figure 2.1, including envelope (E protein), membrane (M protein), nucleocapsid (N protein), and spike (S protein) that infect humans and a wide range of animals. The coronavirus belongs to the Coronaviridae family and the Nidovirales order. The name stems from its crown-like surface glycoprotein. Among the subtypes of coronaviruses that can infect humans, the risk factor varies, as they cause respiratory tract infections that range from mild to fatal. Mild illnesses in humans include some cases of the common cold, while fatal cases are caused by the infections of severe acute respiratory syndrome (SARS), which originated in China in 2002, and Middle East respiratory syndrome (MERS) in 2012 with a fatality rate of around 40% [1]. COVID-19, which is caused by the novel severe acute respiratory syndrome coronavirus 2 (SARS-CoV2), first originated in Wuhan the capital of Hubei Province of the People's Republic of China, on December 27, 2019. It appears that at the Wuhan seafood market, where live poultry and wild animals were sold, SARS-CoV2 was able to transition from animals to humans. The virus continued to spread and by the 11th of March 2020, the World Health Organization (WHO) declared a pandemic situation [35]. So far, there have been more than 774 million infected cases and around 7 million deaths from SARS-CoV-2 infection have been identified [36]. The coronavirus is transmitted commonly by respiratory droplets and can be asymptomatic between 2 and 14 days [37]. In addition, SARS-CoV-2 is able to modify the genomic sequence of human cells during the time of replication, and thus several mutations of the virus have emerged [1].

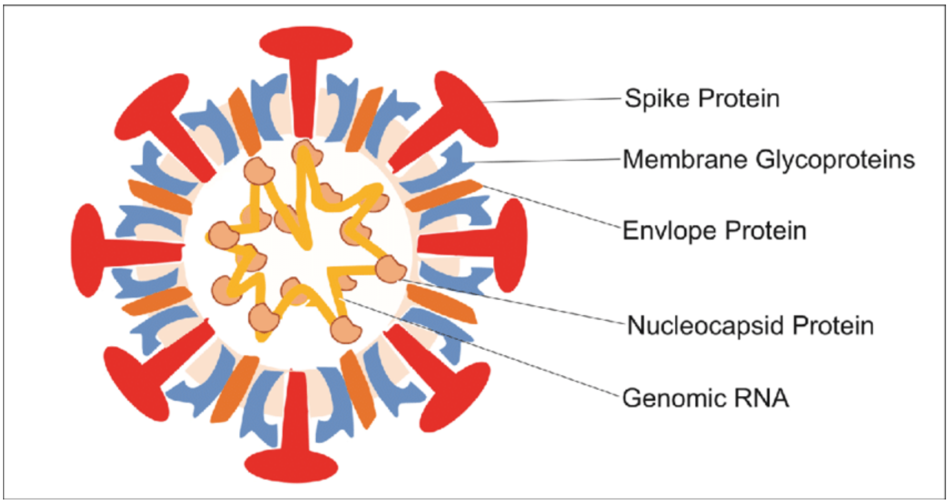


Figure 1.1: Structure of SARS-Cov-2

1.2 Transmission and Personal Infection Prevention Methods

COVID-19 can spread among individuals in several ways. The primary method of transmission is by air particles, which can be spread by activities such as speaking, coughing and sneezing. These particles can stay in the air for up to three hours and range in size from larger respiratory droplets to tiny aerosols. Contaminated aerosols or droplets can enter an individual's respiratory system through the nose, mouth or eyes and cause an infection. The virus can travel greater distances in busy or poorly ventilated interior environments, but it is mainly transmitted between people in close proximity to each other. This is because individuals frequently spend longer amounts of time in these locations, and the virus's particles either linger in the air longer or spread farther during that time. Contamination can also result by circuitous interaction. This may occur from contaminated objects coming into direct or indirect contact with the mouth, nose, or eyes through the hands. Whether a person with the virus has symptoms or not, it can still spread from them. People with mild symptoms can spread the infection to others for longer periods of time, but those with severe symptoms seem to be more contagious just before symptoms appear [2]. Additionally, data have indicated that SARS-CoV-2 transmission can also occur as a result of contact with contaminated inanimate objects, also known as fomite transmission [38]. Stainless steel and plastic surfaces seem to allow the virus to be detected for up to 48 and 72 hours respectively [39].

Despite the various ways of contamination and the fact that people may be infected with the virus and spreading it without showing symptoms themselves, there exist several measures that can halt the transmission of the disease. First and foremost, wearing a mask has been one of the firstly proposed methods during the COVID–19 outbreak [3]. A mask should be worn when an individual is either displaying symptoms of the disease or is spending time in locations where contact with infected people is possible, such as hospitals, public transport, etc. [38]. Furthermore, improving ventilation and air filtration can help prevent virus particles from accumulating in indoor spaces. Achieving better air ventilation and filtration can help reduce the possibility of infection and transmission of the virus that causes COVID-19. Spending time outside, when possible, instead of inside can also help, since viral particles spread between people more readily indoors than outdoors [3]. Some actions to avoid high concentration of air particles contaminated with the SARS-CoV-2 virus, as stated by the Centers for Disease Control and Prevention (CDC), are frequent change and the use of filters that are properly fitted and provide higher filtration in the heating, ventilation, and air conditioning (HVAC) system, and opening windows to bring in as much outdoor air as possible.

In the case of exposure to the virus the individual should abide by the following measures, as suggested by the CDC [40]and the World Health Organization (WHO) [41];

- Stay home and separate from others as much as possible.
- Self–isolate from symptom onset and comply with the self–isolation timeline provide by your local and national authorities.
- If you have a fever, cough, and difficulty breathing, seek medical attention immediately. Call by telephone first and follow the directions of your local health authority.
- If you need to leave your house or have someone near you, wear a properly fitted mask to avoid infecting others.
- Use a separate bathroom, if possible.
- Take steps to improve ventilation at home, if possible.
- Don’t share personal household items, like cups, towels, and utensils.

1.3 Covid–19 Symptoms

Globally, 80% of the reported COVID-19 cases presented with mild respiratory symptoms, 15% of cases required hospitalization and 5% cases were critical in nature The vast majority of patients, according to a study conducted by Talukder et.al [4], present with mild respiratory symptoms. The most typical symptoms are fever, dry cough, fatigue and loss of taste and/or smell; upper respiratory tract symptoms can include pharyngalgia, headaches, and myalgia [5]. Severe COVID-19 symptoms include shortness of breath, loss of speech or mobility, confusion, and chest pain. Symptoms may appear 2-14 days after contamination, with the average time interval being 5-6 days. COVID-19 can infect a lot of different cells and systems of the body, with the mostly affected parts being the upper respiratory tract (sinuses, nose, and throat) and the lower respiratory tract (windpipe and lungs). What is more, it has been observed that the severity of SARS-CoV-2 virus' symptoms varies with regards to biological factors such as gender, age, race, and even social factors as for example income and social class [42]. While exhaustion, dyspnea, joint pain, and chest discomfort are the most prevalent residual symptoms, reports of organ failure in the heart, lungs, and brain have also been made. Thromboembolic disease and myocardial damage have been documented in people with severe sickness. Regarding the long-term effects of COVID-19 on the lungs, research suggests that patients may experience prolonged symptoms, poor lung function, and radiological findings for as long as three months after being released from the hospital. The aforementioned symptoms, i.e. symptoms existing for more than 3 weeks, are labeled in literature as long COVID or post–COVID–19 syndrome. Additional chronic symptoms could be headaches, myalgia, palpitations, chest and joint pains, cognitive and mental impairments, taste and smell abnormalities, coughing, headaches, and problems with the heart and gastrointestinal tract [43].

1.4 Available Treatments

Throughout the course of COVID–19 disease's life several treatments have been tested and utilized by clinicians. The choice of the treatment is tied to the severity of the symptoms, the patient's medical history, as well as the variant of the virus with which the individual has been infected with. A case in fact is since the prevalence of the Omicron variant, treatments that were previously considered best practice, such as bamlanivimab plus etesevimab, are no longer recommended [44]. Due to the ever-changing nature of the disease, vaccinations have been proven to be the most effective way to halt the advance of the virus and decrease the severity of the symptoms of infected individuals [45].

Four vaccines produced by different companies, Pfizer/BioNTech, Moderna, Johnson & Johnson/Janssen and AstraZeneca have been approved by the European Medicines Agency (EMA) and belong to either one of the three available types of vaccine, mRNA, adenovirus and nonreplicating viral vectored. The first three vaccines were developed in the USA, proceed to a phase 3 clinical trial and are administered intramuscularly (IM), whereas the last one was developed in the United Kingdom (UK) [1]. Pfizer/BioNTech and Moderna vaccines utilize the new mRNA vaccine technology, which differentiates their products from the competition. The fundamental mechanism underlying the mRNA vaccine technology is based on a vehicle that enables the delivery of a nucleic acid molecule encoding the antigen of interest into the target cell in the human host, thus allowing the host cell to fabricate the target protein and express the antigen to elicit the immune response. In this way, upon invasion by a pathogen carrying the antigen, the immune system of the host can quickly trigger humoral and cellular immune responses, thereby preventing the disease [45]. Both of these vaccines were administered in two IM dosages, with an injection interval of 21 and 28 days respectively [1].

On the other hand, the vaccine developed by Janssen used preexisting technology with an adenovirus vector to trigger an immune response and offer protection for subsequent infection [46]. This vaccine is administered in a single intramuscular dose. The vaccine produced by AstraZeneca in collaboration with the University of Oxford is a nonreplicating viral vectored vaccine candidate in clinical development. The administration method is also in IM form with two dosages in the same timeframe as the Moderna vaccine [1].

The following figure illustrates the incidence of suspected vaccine complications recorded in the European database of suspected adverse drug reactions reports (EudraVigilance) as of August 6, 2021. [47].

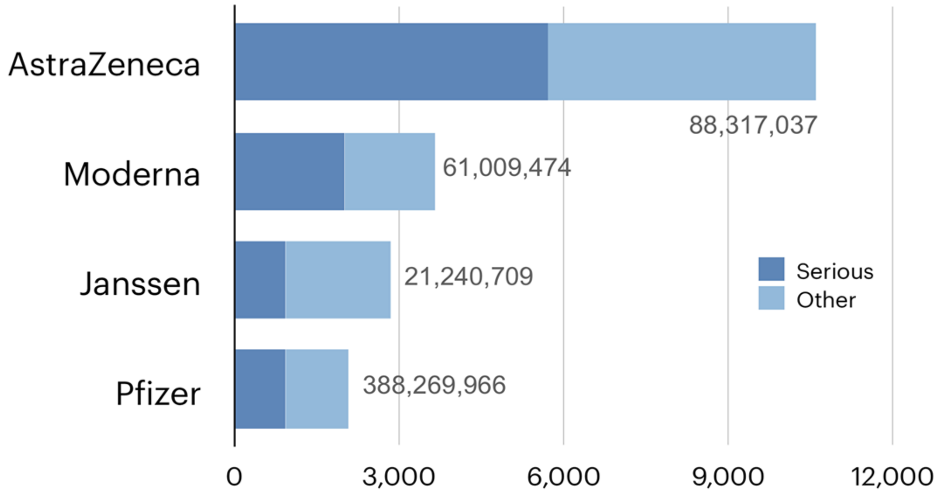


Figure 1.2: All reported adverse effects per 1M vaccine doses [18]

1.5 Motivation of the current study

Healthcare systems worldwide faced significant challenges due to the COVID-19 pandemic. The initial wave was especially difficult due to the fact that a high number of patients needed critical care, there were insufficient resources for patient management and non-COVID-19 care procedures were seriously jeopardized [48]. Alongside this, there was a lot of uncertainty regarding the therapy and clinical path, a lot of trepidation in the community, and a lot of rumors and false information about the symptoms of the disease and the result validity of the rapid testing kits. Due to the high demand in fast and easily accessible diagnostic methods, companies developed rapid testing kits that could be purchased by individuals for home–based testing, while also providing an alternative to RT–PCR tests in medical laboratories. Although the rapid tests are not as accurate as RT–PCR tests, they produce results in under an hour, whereas the latter may take up to two days. Another benefit of rapid test kits is the ability to detect asymptomatic individuals, since they can be conducted relatively easily from one's home and provide accurate results, if the person follows the instructions correctly [38]. However, in order for an individual to keep track of their condition, both in case of exposure and infection to the virus, a significant number of tests needs to be carried out, which in the long run is expensive, time consuming and is prone to inaccurate results because of improper use of the equipment. As a way to combat this issue, several methods of COVID–19 detection have been proposed in recent studies utilizing a variety of machine learning (ML) or deep learning (DL) techniques and data types [49], [8], [11]. In the current thesis the data used are audio recordings of COVID–19 patients and healthy individuals from the Coswara crowdsourced dataset.

1.5.1 Audio Signal Classification

Respiratory diseases along with breathing problems are becoming increasingly common over the years and people are required to visit hospitals and be physically examined by a doctor. In addition, the healthcare specialist may send the patient for a chest x-ray which entails an increase in costs, technical equipment and human resources usage and in waiting time for a diagnosis. This in combination with the fact that according to WHO, 45% of its member states report having less than 1 doctor per 1000 people, which is the WHO recommended ratio [50], accumulate additional strain in the healthcare system. For these reasons, audio–based diagnosis from ML and DL techniques can significantly reduce costs for patients and save valuable time both for the patient and the healthcare professional.

On a study by Mazić et al. [7] a two-layer pattern recognition system architecture is proposed, for the identification of asthmatic wheezing in children's respiratory sounds. The first layer consisted of two parallel SVM classifiers in order to highlight the differences between signals with comparable acoustic features, such wheezes and inspiratory stridors. The proposed structure is further enhanced by the use of a digital detection threshold in the second layer, which aims to improve wheeze detection. The data used were obtained and recorded in the General Hospital of Dubrovnik, Croatia. The recordings were then pre–processed and Mel-Frequency Cepstral Coefficient (MFCCs), along with other audio features were extracted.

Aleixandre et al., on a systematic review paper [8] on COVID–19 detection from audio signals, illustrates that neural network based algorithms were predominantly used by researchers with Convolutional Neural Networks (CNNs) being the first choice among them. Supervised machine learning algorithms were also widely used, with Support Vector Machines (SVMs) and Random Forests (RF) taking the first and second place respectively.

In another study conducted by McNulty et al. [6] regarding the correct inhaler use, an automatic classification system was developed utilizing a quadratic discriminant analysis (QDA) classifier. Recordings from 70 patients using a Diskus inhaler were collected and split into 3 classes (blister, inhalation, interference). A total accuracy of 85.35% was obtained on the testing dataset.

Audio signals have also been used in the diagnosis of Alzheimer's disease (AD), where Shimoda et al. [9] collected 1,616 audio files in total; 1,465 audio data files from 99 Healthy controls (HC) and 151 audio data files recorded from 24 AD patients derived from a dementia prevention program conducted by Hachioji City, Tokyo, between March and May 2020. After the extraction of vocal features from the data, 3 ML models based on extreme gradient boosting (XGBoost), RF, and logistic regression (LR) were developed and the resulting areas under the curve (AUCs) for XGboost, RF, and LR were 0.863 (95% confidence interval [CI]: 0.794–0.931), 0.882 (95% CI: 0.840–0.924), and 0.893 (95%CI: 0.832–0.954), respectively.

1.5.2 Cough Classification

A special category of audio signal data used for classification is that of cough recordings. The impact of coughing on the respiratory system varies and is a common symptom of over 100 diseases and other conditions of medical significance [10], such as COVID-19. The glottis may function differently, and the airway may become limited or clogged due to lung disease, which may affect the vocal audio quality of speech, breath, and cough [13], [51], [52]. This makes it more likely to recognize the coughing sound linked to a certain respiratory illness, such COVID-19.

One study [14] aimed to use Artificial Intelligence (AI) to discriminate COVID-19 subjects, including asymptomatic individuals, solely from a forced-cough cell phone recording. The researchers collected a dataset of COVID-19 cough recordings through their website, resulting in the largest balanced dataset up to the date of the study with 5,320 subjects. Their AI framework leveraged acoustic biomarker feature extractors to pre-screen for COVID-19 from cough recordings, which were transformed with MFCCs, and provided personalized patient saliency maps for real-time monitoring. The framework utilized a CNN architecture with transfer learning to improve COVID-19 discrimination accuracy. Validation showed the model achieved high sensitivity

(98.5%), specificity (94.2%) and area under the ROC curve (AUC) (97%) for COVID-19 diagnosis, with 100% sensitivity for asymptomatic subjects and 83.2% specificity. The findings suggest that AI techniques could provide a free, non-invasive, and large-scale COVID-19 screening tool, suitable for daily use in various settings such as schools, workplaces, and public transportation, potentially aiding in containing the spread of the virus.

What is more, Imran et al. in a 2020 study [15], introduced AI4COVID-19, an AI-powered screening solution deployable via a smartphone app. The app records and sends three 3-second cough sounds to a cloud-based AI engine, returning results within 2 minutes. The researchers showcased that the pathomorphological alternations caused by COVID-19 in the respiratory system are distinct from other common respiratory diseases and thus cough recordings can be utilized effectively in COVID-19 detection. The application initially employs a cough detection CNN – classifier to distinguish a cough from other environmental sounds by transforming the recordings into Mel – spectrograms. The overall accuracy which the cough detector achieved is 95.60%. Then the sound is forwarded to three parallel, different classifier systems, i.e., Deep Transfer Learning-based Multi Class classifier (DTL-MC), Classical Machine Learning-based Multi Class classifier (CML-MC) and Deep Transfer Learning-based Binary Class classifier (DTL-BC). The DTL-MC and DTL-BC use Mel –spectrograms as input and classify the audio as one of four possible classes in the first case (COVID-19, pertussis, bronchitis, or normal person), and one of two in the second (COVID – 19 cough or not). Lastly, the CML-MC uses a concatenated feature matrix of MFCCs and principal component analysis (PCA) extracted features as input to a SVM classifier. The overall accuracy of the three parallel classifiers is 92.64% for the DTL-MC, 88.76% for the CML-MC model and 92.85% for the DTL-BC model. A mediator receives the output from each of these three classifiers and only when all three classifiers produce identical classification results can the app declare a diagnosis. In the case where the results are not the same, the application returns "test inconclusive". While not a clinical-grade tool, AI4COVID-19 offers versatile screening capabilities accessible to anyone, anywhere, aiding in directing clinical testing and treatment to those in need, potentially saving lives.

Pahar et al. in 2021 [16] introduced a machine learning-based COVID-19 cough classifier capable of distinguishing COVID-19 positive coughs from both negative and healthy coughs recorded on smartphones. The Coswara dataset comprises 92 COVID-19 positive and 1079 healthy subjects, while the Sarcos dataset, a smaller dataset from South Africa, includes 18 COVID-19 positive and 26 COVID-19 negative subjects with SARS-CoV laboratory tests. Addressing dataset skew with the synthetic minority oversampling technique (SMOTE), seven machine learning classifiers were trained and evaluated using leave-p-out cross-validation. Results reveal the ResNet-50 classifier achieves the highest performance in discriminating between COVID-19 positive and healthy coughs, with an area under the ROC curve (AUC) of 98%. Additionally, an LSTM classifier (which has been proven effective in various medical tasks [53], [54]) effectively discriminates between COVID-19 positive and negative coughs, achieving an AUC of 94% after feature selection. Given its cost-effectiveness and ease of deployment, this non-contact cough audio classification holds promise as a practical tool for COVID-19 screening.

A year later another study was punished by the same researchers [13], which investigates the efficacy of transfer learning and bottleneck feature extraction in detecting COVID-19 from audio recordings of cough, breath, and speech. This non-contact screening method, deployable on consumer hardware like smartphones, does not require specialized medical expertise or laboratory facilities. Pre-training three DNNs (CNN, LSTM, Resnet50) on datasets lacking COVID-19 labels, the study fine-tunes these networks with smaller COVID-19 labeled cough datasets or utilizes them as bottleneck feature extractors. Results reveal ResNet-t50 classifier, trained via transfer learning, achieves optimal or near-optimal performance across all sound classes (coughs, breaths, speech), with ROC AUC scores of 98%, 94%, and 92% respectively. Coughs exhibit the strongest COVID-19 signature, followed by breath and speech. Transfer learning and bottleneck feature extraction with larger datasets enhance performance and reduce standard deviation of classifier AUCs during nested cross-validation, indicating improved generalization. The study concludes that deep transfer learning and bottleneck feature extraction enhance COVID-19 audio classification, facilitating automatic COVID-19 detection with improved and consistent overall performance.

1.5.3 Non–stationary Data

A prevalent challenge in ML and DL applications such as, fault detection, diagnostics, remaining useful life prediction in industrial components, etc. lies in the nonstationary nature of the environments where data streams are gathered. Concept drifts, also known as common causes of nonstationary behaviors, include effects like seasonality, sensor or component degradation, thermal variations, and shifts in operation modes or user interests. The outbreak of the COVID-19 pandemic, which has caused a rapid and ongoing shift in conditions across industries ranging from financial services to healthcare, is a prime example of data drift. When confronted with such nonstationary environments and situations, the adaptation of ML models emerges as a pivotal concern.

In a study conducted by Duckworth et al. [17], the application of explainable machine learning to monitor data drift during the COVID-19 pandemic is demonstrated through the use of a ml classifier and SHapley Additive exPlanations (SHAP). Pseudonymised patient attendance record of 82,402 adults from the Southampton General Hospital's Emergency Department occurring from the 1st April 2019 to the 30th of April 2020 were utilized in the study, with the data up to March 2020 being used as pre pandemic training and test data and the rest as COVID-19 test set. An XGBoost model is trained and evaluated in weekly bins throughout the complete test period and is able to achieve an average AUROC score of 85.6% on the pre pandemic test set, and 82.6% on the COVID-19 test set. The use of SHAP in explainable machine learning offers two key benefits in healthcare settings: (1) tracking variation in feature SHAP values as a measure of data drift, indicating the need for model retraining, and (2) identifying emergent health risks by observing changes in feature importance.

Disabato and Roveri on their 2019 research paper [18], introduce an adaptive mechanism enabling CNNs, which have been traditionally unsuitable for such systems due to computational demands and training data requirements, to function amidst concept drift. This mechanism employs an active approach, where adaptation is triggered by detecting concept drift, and utilizes transfer learning to transfer knowledge from the CNN before the drift to the one after, while retraining only the layers that became obsolete. The effectiveness of this approach is evaluated on two CNN types using two real-world image benchmarks, with a consistent increase in accuracy.

In [19] an unsupervised method called D3 (Discriminative Drift Detector) is presented, which utilizes a discriminative classifier with a sliding window to detect concept drift by monitoring changes in the feature space. D3 is a straightforward method compatible with existing classifiers lacking intrinsic drift adaptation mechanisms. A logistic regression model is utilized to distinguish between the old and the new data sets which are “contained” in a fixed size sliding window. A drift is detected with respect to classifier’s performance regarding the AUROC score. This process is done repeatedly as long as there is new data. Experimentation with eight datasets demonstrates that D3 outperforms methods such as ADWIN [20], DDM [21] and EDDM [22] resulting in models with improved performance on both real-world and synthetic datasets.

Another novel approach to concept drift detection is presented in this study [23], combining the development of online sequential extreme learning machines (OS-ELMs) [24] with quantifying model modifications due to newly collected data. This method is validated through synthetic case studies and applied to real-world datasets and an energy production prediction problem from a wind plant. Results demonstrate the effectiveness of the proposed method compared to alternative concept drift detection techniques. Moreover, updating the prediction model upon detecting concept drift improves the overall accuracy of the energy prediction model while minimizing the frequency of model updates.

2

Theoretical Background Information

2.1 Audio Signals

2.1.1 Mel-Frequency Cepstral Coefficients – MFCCs

The Mel scale relates the perceived frequency or pitch of a pure tone to its actual measured frequency. Humans are much better at distinguishing small changes in pitch at low frequencies than at high frequencies. Incorporating this scale makes our features more closely match what people hear [30]. The formula for converting from frequency to Mel scale is:

$$M(f) = 1125 \times \ln(1 + \frac{f}{700})$$

MFCCs are a set of features commonly used in speech and music processing. They are derived from the log mel spectrogram by applying the Discrete Cosine Transform (DCT) to the mel filterbank energies. MFCCs capture the spectral characteristics of the audio signal and are often used as input features for machine learning models [16], [55].

2.1.2 MFCCs Deltas and Delta – deltas

By calculating the first order MFCC features the Delta MFCC (velocity) features can be extracted. Since audio signals are time-variant signals, delta features represent the change in the cepstral features over time. Each of the delta feature extracted as the first derivative of MFCC feature represents the change between frames. The only benefit of

Delta features over MFCC features is that they represent the temporal information. One common technique allowing to differentiate crossing trajectories are delta features.

Delta-delta features (acceleration) are derived by computing the second order of the MFCCs, or by calculating the first order derivative of the delta. They illustrate the change between frames in the corresponding delta features.

2.1.3 Kurtosis

Kurtosis is a statistical measure used to describe the distribution of data points in a dataset, e.g. how steep or flat the peak of the curve is. If kurtosis is high, there are more peaks in the signal and their amplitudes are greater. For audio signals, it indicates the prevalence of higher amplitudes. In real-life conditions, a lot of vibrations are characterized by signals with a kurtosis value higher than three (Gaussian random).

2.1.4 Root Mean Square Value - RMS

Root mean square is a metering tool that measures the average loudness of an audio track within a window frame. The RMS value will provide a more accurate look at the perceived loudness of the audio track for the average listener.

2.1.5 Zero-Crossing Rate – ZCR

The zero-crossing rate is the rate at which an audio signal changes from positive to zero to negative or from negative to zero to positive and indicates the variability of the signal. Its value has been used widely in both speech recognition and music information retrieval, being a key feature to classify percussive sounds. ZCR is a very effective way to detect vocal activity that determines whether a frame of speech is spoken, unheard, or silent. The zero-crossing rate for unvoiced segments is much higher than for voiced segments. In ideal conditions the ZCR for a segment of silence in a clear speech should be equal to zero [30].

2.1.6 Audio to Image Transformation - Mel-Spectrograms

Mel – Spectrograms are spectrograms where the frequencies are converted to the Mel scale. Humans do not perceive frequencies linearly but in a logarithmic scale. Although the difference between two pairs of sounds, with the first one containing sounds of 500 and 1000 Hz and the second one of 7500 and 8000 Hz, equals 500 Hz in both cases, the difference between the second pair of sounds is almost unnoticeable. The Mel Scale is the result of transforming the frequency scale and constitutes a perceptual scale of pitches, which are judged by listeners to be equal in distance from one another.

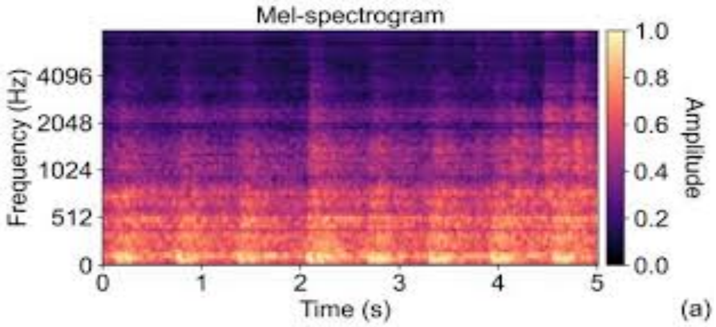


Figure 2.1: Example of a Mel-Spectrogram

2.2 Machine Learning & Deep Learning

In 1956, a group of computer scientists laid the foundation for the concept that computers could emulate human thinking and reasoning. They posited that "every aspect of learning or any other feature of intelligence [could], in principle, be so precisely described that a machine [could] be made to simulate it." [25]. This principle became known as "artificial intelligence" (AI). In essence, AI constitutes a field dedicated to automating intellectual tasks typically executed by humans. Within this domain, ML and DL emerge as specific methodologies aimed at achieving this objective by discerning patterns from data to enhance performance across a variety of tasks. ML leverages historical data as input to facilitate predictions, information classification, data clustering, and dimensionality reduction, among other functions. The range of ML techniques available empowers software applications to refine their performance iteratively.

Notably, ML finds extensive application across various industries. For instance, recommendation engines employed by e-commerce, social media platforms, and news agencies rely on ML algorithms to suggest content based on users' past behaviors. Moreover, in the realm of self-driving vehicles, ML algorithms and machine vision constitute indispensable components, enabling vehicles to navigate roads safely. In the arena of applied healthcare research, ML serves as a tool for automating and flexibly analyzing complex data structures. This approach, characterized by its computational intensity, is adept at identifying intricate patterns such as nonlinear associations, interactions, underlying dimensions, or subgroups. This contrasts with "traditional" parametric methods, which entail numerous statistical assumptions and necessitate a priori specification of dimensions, functional relationships between predictors and outcomes, and predictor interactions.

In ML, there are four commonly used learning methods (supervised, unsupervised, semi–supervised, and reinforcement learning) [56] that are depicted in the following figure 2.2, along with some of their respective applications.

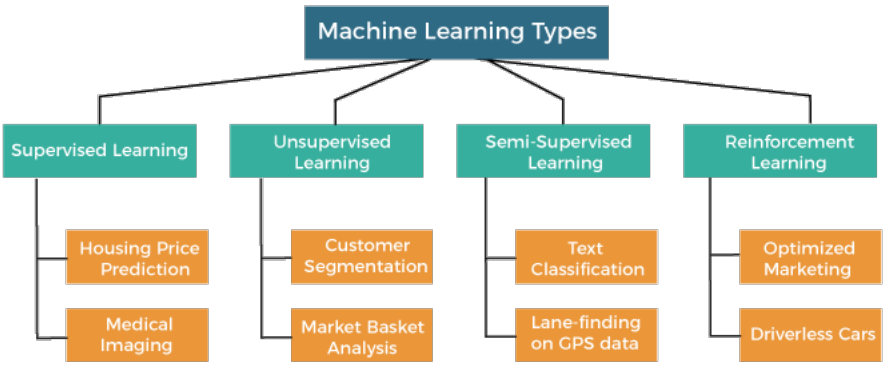


Figure 2.2: Types of Machine Learning

2.2.1 Decision Trees & Random Forests

A decision tree is a supervised learning approach primarily utilized for classification tasks, although it can also be used for regression tasks. Its structure commences with a root node, marking the initial decision point for dividing the dataset, housing a singular feature that optimally separates the data into distinct classes. Each division generates an edge connecting either to a subsequent decision node, incorporating another feature to further partition the data into homogeneous groups, or to a terminal/leaf node, responsible for predicting the class. This recursive partitioning process distinguishes the data into binary partitions.

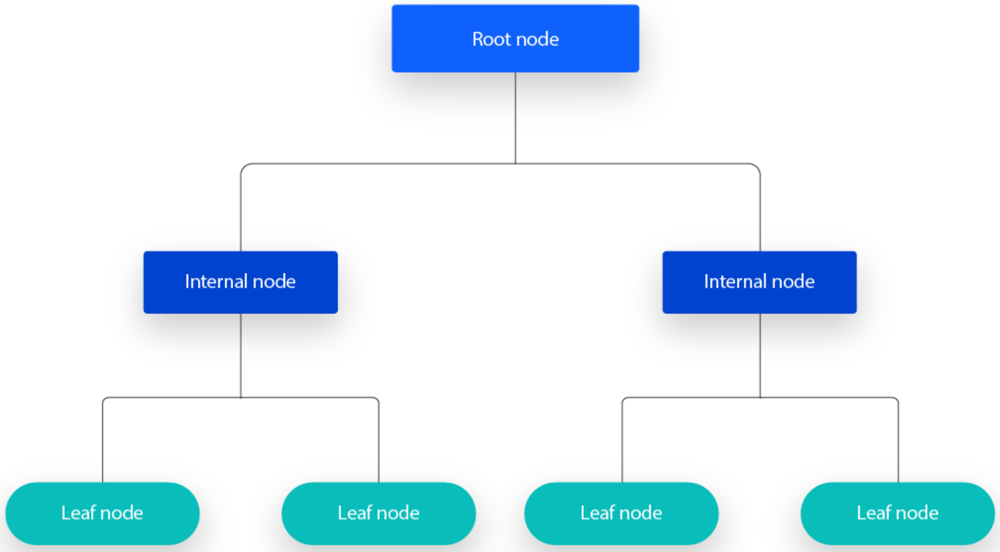


Figure 2.3: Decision Tree structure

A random forest or RF is therefore an ensemble method which extends the decision tree methodology by generating multiple decision trees. Unlike using all features to construct each decision tree, a random forest employs a subset of features to build individual trees. Subsequently, the trees collectively predict class outcomes, and the predominant class prediction among the trees determines the final model's classification.

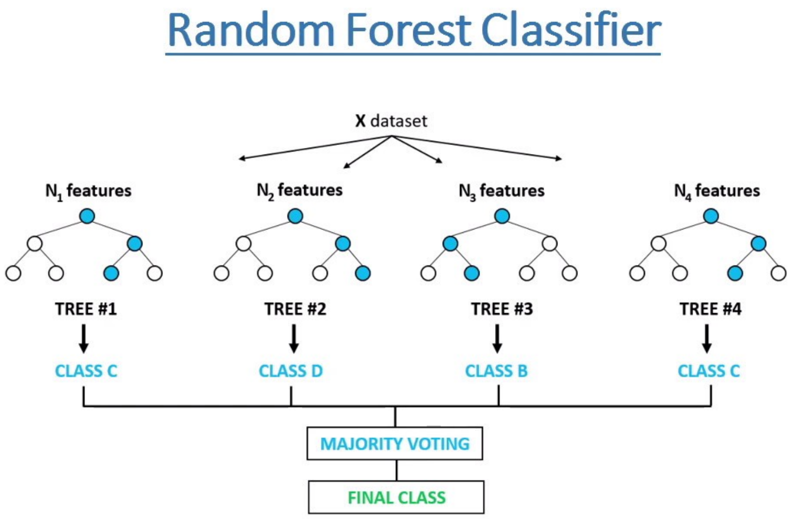


Figure 2.4: Random Forest classifier structure

2.2.2 Multi–Layer Perceptron

A machine learning technique that draws inspiration from biological neural networks is known as an artificial neural network (ANN). Every ANN is made up of nodes, which are like cell bodies, and connections, which are like axons and dendrites, which allow nodes to communicate with one another. Weighted connections between nodes are used based on their capacity to produce a desired outcome, much like a biological neural network where synapses between neurons are strengthened when their neurons have correlated outputs (the Hebbian theory states that “nerves that fire together, wire together”) [57]. Information from each node in the previous layer is passed to each node in the next layer, transformed, and then fed forward to each node in the next layer. This type of neural network is called a feedforward neural network. A multilayer perceptron or MLP, (Rosenblatt 1958), is a feedforward artificial neural network model, consisting of multiple layers of neurons fully connected to the next neurons in each layer. A number of interconnected perceptrons make up the MLP. A perceptron is a ML algorithm that looks for a line, plane, or hyperplane in a hyperdimensional space that divides the data into classes after receiving a set of features and their targets as input.

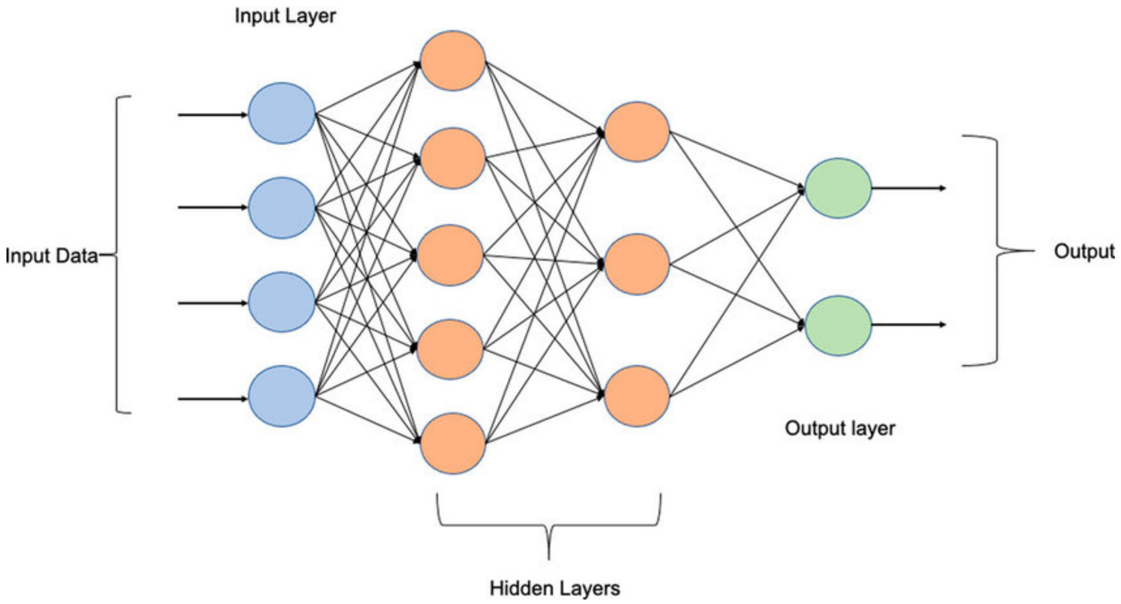


Figure 2.5: Multilayer Perceptron example

An activation function is used to transform a node’s input into a preferred output. The activation functions tested for this architecture are the hyperbolic tangent function or Tanh and the rectified linear unit function or ReLU.

Tanh is a non – linear function that takes a real number as input and, through the formula shown below, transforms it into the range of [-1, 1] with its center being zero. An issue with the tanh non – linearity is that when the neuron's activation saturates at either -1 or 1, the gradient at these regions is almost zero which causes the vanishing gradient problem [58].

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

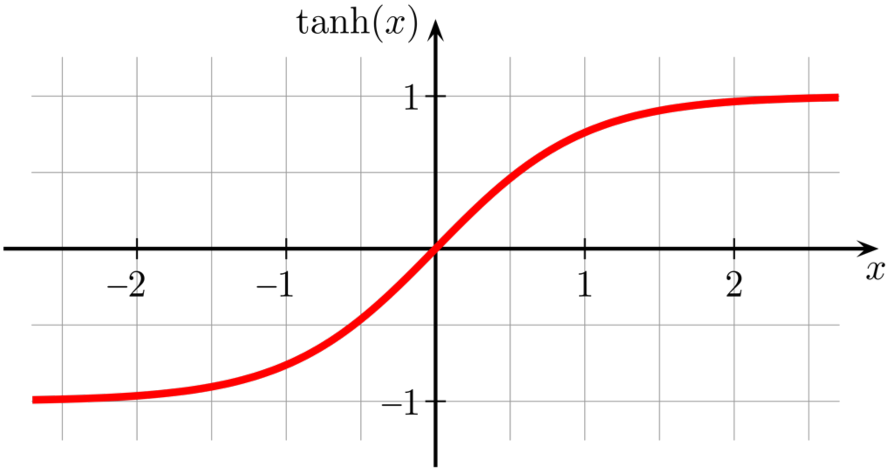


Figure 2.6: Tanh activation function

ReLU is another non – linear function that is less computationally expensive than tanh and it avoids the vanishing gradient problem. Due to these two reasons, it has become one of the most widely used activation functions within hidden layers of a neural network. The formula of this function is given below.

$$relu(x) = \max{(0, x)}$$

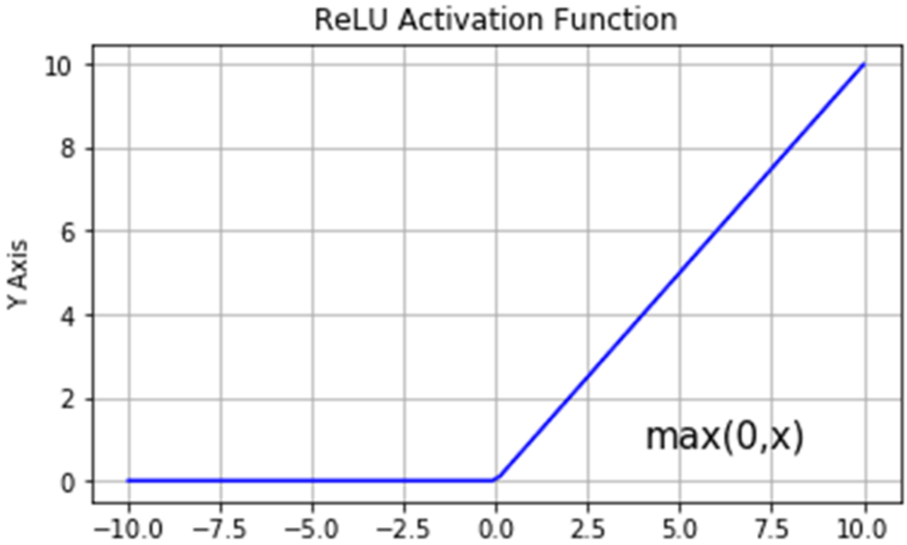


Figure 2.7: ReLu activation function

2.2.3 Convolutional Neural Networks – CNNs

For tasks involving image recognition, every input to a feedforward ANN corresponds to a pixel within the image. However, this approach has a significant disadvantage; interconnections between nodes are non-existent, and thus the spatial context of the features (with give meaning to the image) is lost. This is especially important since neighboring pixels within an image exhibit higher correlation compared to pixels at distant locations. In order to address this limitation of feedforward ANNs, Convolutional Neural Networks (CNNs) are introduced as a specialized category of the former capable of preserving the spatial correlation among pixels in an image. Unlike feedforward ANNs that process individual pixels, CNNs process and transmit areas of an image to specific nodes within subsequent layers, thereby maintaining the spatial context from which the feature was extracted. These image areas or patches, known as convolutional filters, are pivotal in discerning specific features and are extensively utilized in a variety of image processing tasks, including image blurring, sharpening, and edge detection. In the context of digital images, a grayscale image constitutes a singular matrix, while a color image comprises three stacked matrices representing red, green, and blue color channels. Convolutional filters, typically square matrices (kernels) ranging from 2x2 to 9x9, are traversed over the original image, while element-wise matrix multiplication is performed at each position. The mathematical description of the convolution of a kernel $k(z,w)$and an image $f(x,y)$ is depicted below.

$$k * f(x,y) = \sum_{s=-a}^{a} \sum_{s=-b}^{b} k(s,t)f(x-s, y-s)$$

The resulting convolution output is mapped to a new matrix, referred to as a feature map, indicating whether the convolutional filter detected relevant features or not. In CNNs, filters are trained to identify specific features within images, such as vertical lines or U-shaped objects, and annotate their positions on the feature map. Subsequently, a deep CNN employs the feature map as input for the subsequent layer, which employs new filters to generate another feature map. This iterative process continues across multiple layers, where the extracted features progressively become abstract yet valuable for predictive tasks. Ultimately, the final feature maps are compressed and fed into a feedforward ANN for image classification based on the extracted features a process commonly known as Deep Learning (DL).

Some more CNN layers beyond the convolution layer described above are:

The **pooling layer** which derives a summary statistic of the nearby outputs and uses it as the output of the NN at certain locations. This technique is especially important in reducing the spatial size of the representation, hence decreasing the required amount of computation and weights needed. The pooling operation is processed on every slice of the representation individually. There are several pooling functions such as the average of the rectangular neighborhood, L2 norm of the rectangular neighborhood, and a weighted average based on the distance from the central pixel. However, the most popular process is max pooling, which reports the maximum output from the neighborhood.

The **fully connected layer** (FC) or **dense layer** in which neurons are full connected with all neurons in the preceding and succeeding layer. A matrix multiplication followed by a bias effect is all that is need for the output of this layer to be computed. The FC layer is used as the final layer in a CNN model because it helps to map the representation between the input and the output.

Finally, another layer that is used in this thesis is the **Gaussian Noise layer** (GS) from keras api library[1]. GS applies an additive zero-centered Gaussian noise, which could be considered as a form of random data augmentation, if applied as an input layer, and is useful in mitigating overfitting. It is also a natural choice as corruption process for real

[1] https://keras.io/2.15/api/layers/regularization_layers/gaussian_noise/

valued inputs. As it is a regularization layer, it is only active at training time. The most common noise application is to the inputs of the model, but it can also be added to other parts of the NN during training. In this study CNN model architectures are implemented using the GS either as an input layer or as a hidden layer adding noise to the weights of the model, which can be considered equivalent (under some assumptions) to a more traditional form of regularization, encouraging the stability of the function to be learned [59]. This technique has been implemented successfully in the context of recurrent neural networks (RNNs) by Graves et al. [60], [61].

2.2.4 Transfer Learning

The theory behind transfer learning (TL), which is based on cognitive research, is that information gained on related tasks can be applied and enhance performance on unrelated tasks. Humans are known to be able to tackle comparable tasks by using prior knowledge. The formal definition of TL is defined by Pan and Yang with notions of domains and tasks. “A domain consists of a feature space X and marginal probability distribution P(X), where X = {x_1, ..., x_n} ∈X. Given a specific domain denoted by D = {X, P(X)}, a task is denoted by T = {Y, f (·)} where Y is a label space and f (·) is an objective predictive function. A task is learned from the pair {x_i, y_i} where xi ∈ X and y_i ∈ Y. Given a source domain D_S and learning task T_S, a target domain D_T and learning task T_T, transfer learning aims to improve the learning of the target predictive function f_T (·) in D_T by using the knowledge in D_S and T_S” [62]. In similar fashion, one can learn how to drive a motorbike T_T (transferred task) based on one’s cycling skill T_s (source task) where driving two-wheel vehicles is regarded as the same domain $D_S = D_T$. This does not mean that one will not learn how to drive a motorbike without riding a bike, but it takes less effort to practice driving the motorbike by adapting one’s cycling skills. Similarly, learning the parameters of a network from scratch will require larger annotated datasets and a longer training time to achieve an acceptable performance [63]. Some of the advantages of TL, are reduced training time, improved neural network performance (in most cases), and the absence of a large amount of data to accomplish such performance. These advantages are the main reason why TL is used in this study and why TL has shown promising results in the field of medical imaging [64].

TL with CNN involves the transfer of knowledge at the parameter level. Pretrained CNN models employ the parameters of convolutional layers for new tasks, particularly in the medical domain. In TL with CNN for medical image classification, the classification of medical images (target task) can be accomplished by leveraging the generic features learned from natural image classification (source task), where labels are available in both domains. In the scenario studied in this thesis, both domains involve image analysis, and pretrained CNN models use ImageNet data for medical image classification in a supervised manner. Broadly, two TL approaches exist for leveraging CNN models: the feature extractor and fine-tuning methods. The feature extractor approach involves freezing the convolutional layers, while the fine-tuning method updates parameters during model fitting. Each approach can be further categorized into two subcategories, resulting in four TL approaches. In the feature extractor hybrid approach, the FC layers are discarded, and a machine learning algorithm is attached to the feature extractor. Conversely, in the other types, the structure of the given networks remains unchanged. Fine-tuning from scratch represents the most time-intensive approach, as it updates the entire ensemble of parameters during the training process [63]. In this thesis the structure of the pretrained CNN model or base model remains unchanged and new layers are added on top of the base model, in order to be trained on the new data.

2.2.5 Zero – Shot Learning

As it was previously stated, large, annotated datasets prepared by clinicians or experts are required to train a deep learning model. Especially in areas such as medical imaging, where sufficient data quantities are not readily available, or they simply do not exist yet (e.g. outbreak of COVID – 19), it becomes necessary to establish an alternative method that requires less or no annotated data at all to provide predictions on the new concept. That particular challenge of learning a new concept without receiving any examples beforehand, thus avoiding the requirement for labor-intensive data collecting and professional annotation, is called Zero-Shot Learning (ZSL) [26], [27], [28]. This new learning technique does not use examples from unknown categories in training, but instead builds recognition models using information from previously encountered categories and additional data. The additional data could be vectors of word labels, characteristics, or textual descriptions. As a result, ZSL is intrinsically interdisciplinary, combining textual and visual data as two complementary parts [27].

The process of employing phrases, keywords, or labels that the model can utilize for making predictions is called phrase engineering and it is interlinked with the research questions at hand. It is essential to acknowledge that identifying highly discriminative phrases for ZSL can be laborious and may necessitate an iterative approach involving various phrase adjustments. In cases where the model exhibits subpar performance in the classification task, experimenting with certain phrases on manually annotated data and documenting the phrases tested should be tried. In order to get more accurate results, researchers can employ their experience, knowledge of the literature, data, and theoretical frameworks, as well as some creativity, in phrase engineering. In this study several phrases have been tested and compared so as to examine how different labels can affect the prediction output of the model.

The model utilized on this approach is called CLIP[2], which stands for Contrastive Language-Image Pre-training. CLIP is a NN trained on 400 million (image, text) pairs. Given an image, it is able to predict, in natural language, the most relevant text snippet without having to be directly optimized for the task. In the original paper by Radford et al., researchers showcased that CLIP can match the performance of the original ResNet50 model on ImageNet "zero-shot" without using any of the original 1.28 million labeled examples [65]. The implementation approach of CLIP is depicted at figure 2.8.

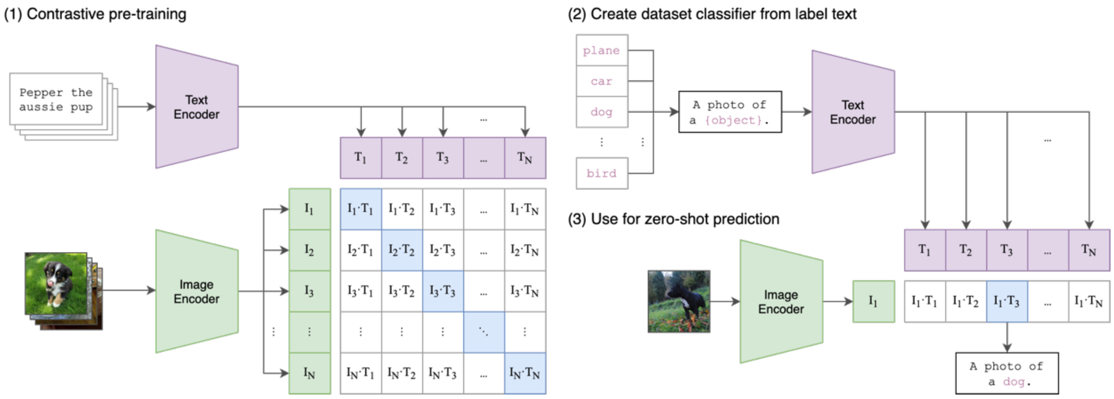

(1) Contrastive pre-training
Pepper the aussie pup
Text Encoder
Image Encoder
T_1 T_2 T_3 ... T_N
I_1 I_2 I_3 ⋮ I_N
$I_1 \cdot T_1$ $I_1 \cdot T_2$ $I_1 \cdot T_3$ ... $I_1 \cdot T_N$
$I_2 \cdot T_1$ $I_2 \cdot T_2$ $I_2 \cdot T_3$ ... $I_2 \cdot T_N$
$I_3 \cdot T_1$ $I_3 \cdot T_2$ $I_3 \cdot T_3$ ... $I_3 \cdot T_N$
$I_N \cdot T_1$ $I_N \cdot T_2$ $I_N \cdot T_3$ ... $I_N \cdot T_N$
(2) Create dataset classifier from label text
plane
car
dog
bird
A photo of a {object}.
Text Encoder
(3) Use for zero-shot prediction
Image Encoder
I_1
T_1 T_2 T_3 ... T_N
$I_1 \cdot T_1$ $I_1 \cdot T_2$ $I_1 \cdot T_3$ ... $I_1 \cdot T_N$
A photo of a dog.

Figure 2.8: CLIP

In summary an image encoder and a text encoder are trained in tandem to predict the correct couplings of a batch of (image, text) training examples. The goal is to maximize

[2] https://github.com/openai/CLIP

the cosine similarity of the image and text embeddings of the correct pairs in the batch while minimizing the cosine similarity of the embeddings of the incorrect (image, text) combinations. At test time the learned text encoder synthesizes a zero-shot linear classifier by embedding the descriptions of the target dataset's classes.

2.3 Concept Drift

The term "concept drift" refers to unforeseen shifts in the streaming data's underlying distribution over time. As a result, predictions of the models trained in the past may become less accurate as time passes or opportunities to improve the accuracy might be missed. It was originally proposed by Schlimmer and Granger [29] in 1986, who aimed to point out that noisy data may eventually become no-noisy information at a different time. Therefore, learning models need to have mechanisms for continuous diagnostics of performance, and be able to adapt to changes in data over time. Research on concept drift includes creating methods for drift detection, understanding and adaptation.

Changes in underlying data occur due to changing personal interests, changes in population, adversary activities or they can be attributed to a complex nature of the environment. In a study conducted by Uchida and Yoshida in daily infection data of COVID – 19 in Japan [66] the concept drift detection points the extracted points appear to correspond to new COVID-19 variants and other important state changes.. In the traditional supervised learning methods, the training and the testing data come from the same distribution. In real world scenarios though, the predictions need to be made online and often in real time, without any guarantees that the data will belong in the same distribution. Hence, at any point in time the testing data may be coming from a different distribution than the training data has come, and the model's predictive capabilities may degrade severely [67].

Assuming that X_T is the input vector of an ML model at time T and Y_T is the corresponding output target vector, the concept drifts are typically distinguished into three categories:

- **Virtual drifts** in which the distribution of the input data P_T (X) changes with time, whereas the posterior probability of the output P_T (Y|X), representing the mapping relationship between X_T and Y_T, is not changing.
- **Real drifts** in which the posterior probability distributions P_T (Y|X) varies over time, independently of variations in P_T (X).

- **Hybrid drifts** in which virtual and real drifts occur at the same time (hybrid drifts are the most common in industrial applications).

With respect to the type of distribution modification, concept drifts are typically classified as sudden, incremental, gradual, or recurring and are illustrated in Figure 2.9.

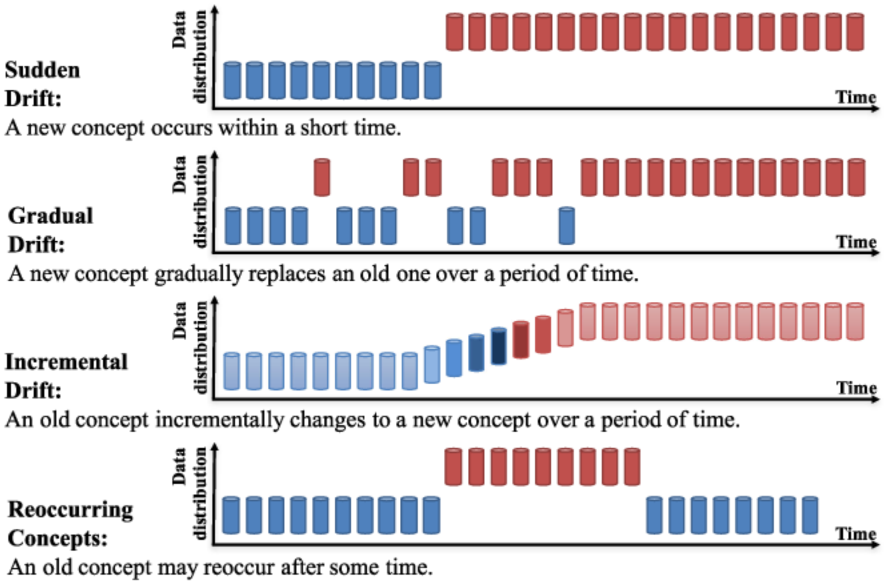

Sudden Drift:
Data distribution
Time
A new concept occurs within a short time.
Gradual Drift:
Data distribution
Time
A new concept gradually replaces an old one over a period of time.
Incremental Drift:
Data distribution
Time
An old concept incrementally changes to a new concept over a period of time.
Reoccurring Concepts:
Data distribution
Time
An old concept may reoccur after some time.

Figure 2.9: Types of concept drift

Considering examples of sensor anomalies, an abrupt signal bias is a sudden drift, a bias whose amplitude increases with time is an incremental drift, signal spikes with increasing frequency before sensor failure constitute a gradual drift, and sensor readings occurring during some periodic plant conditions constitute a recurring drift [68], [69].

In literature there exists significant amount of research in the accurate detection of when a concept drift has occurred, but the current thesis' focus is on examining strategies for updating existing learning models according to the detected drift, which is known as concept drift adaptation. There are three main groups of drift adaptation methods, namely simple retraining, ensemble retraining and model adjusting, that aim to handle different types of drift [68].

Reacting to concept drift often involves retraining a new model with the latest data to replace the outdated one, which necessitates an explicit concept drift detector to determine when retraining is necessary. Typically, a window strategy is employed in this method to retain recent data for retraining while preserving old data for distribution change testing. However, determining an appropriate window size is a challenging task since, a small window better reflects the latest data distribution, whereas a large window increases the training data of the new model. ADWIN [20], a popular window scheme algorithm proposed by Bifet and Gavaldà, addresses this dilemma by dynamically adjusting window sizes based on the rate of change between sub-windows, thus eliminating the need for predefined window sizes. After finding the optimal window cut, the window containing outdated data is discarded, allowing new model to be trained with the latest data.

In cases of recurring concept drift, the preservation and reuse of old models offer significant advantages over repeatedly retraining new models. This principle underlies the use of ensemble methods in managing concept drift, which has garnered attention in the stream data mining community. Ensemble methods entail a collection of base classifiers that may vary in type or parameters, combining their outputs using specific voting rules to predict incoming data. Adaptive ensemble methods have been developed to address concept drift, either by extending classical ensemble methods or by creating adaptive voting rules. Classical ensemble methods like Bagging, Boosting, and Random Forests have been adapted to handle streaming data with concept drift, with approaches such as online bagging [70] and Leveraging Bagging combining with drift detection algorithms like ADWIN [71] to address concept drift. Furthermore, the Adaptive Random Forest (ARF) algorithm extends the random forest tree algorithm with a concept drift detection method, such as ADWIN, to determine when to replace an outdated tree with a new one [72].

An alternative to retraining entire models is to develop adaptive models that can learn from changing data, partially updating themselves as the data distribution shifts. This approach proves more efficient when drift occurs in local regions. Many methods in this category are based on the decision tree algorithm, leveraging trees' ability to examine and adapt to individual sub-regions independently [73], [74], [75].

2.4 Optuna Hyperparameter Optimization Framework

Optuna[3] is an automatic hyperparameter optimization software framework [76], particularly designed for machine learning and deep learning. It features an imperative, define-by-run style user API. Thanks to this API, the code written with Optuna is highly

[3] https://optuna.org

modular, and the user of Optuna can dynamically construct the search spaces of the hyperparameters. Another benefit of Optuna is that it is framework agnostic and thus it can be used with any machine learning or deep learning framework.

In order for the framework to be utilized, the user has to wrap their model with an *objective* function, where the hyperparameter search–space is specified using a *trial object* and return the metric that is going to be used as an optimization criterion e.g. accuracy, auc-roc score, mean squares error (MSE), etc. Lastly, the user creates a *study object*, where the number of trials, direction of optimization (maximization or minimization of the return metric) and other parameters are specified, and the optimization can be executed. Optuna also offers the ability to save a study and continue it later, prune an unpromising trial, use different sampling methods, as well as a plethora of fast and useful visualizations.

The sampling method utilized is the Tree-structured Parzen Estimator algorithm or TPE [77]. On each trial, for each parameter, TPE fits one Gaussian Mixture Model (GMM) L(X) to the set of parameter values associated with the best objective values, and another GMM G(X) to the remaining parameter values. It chooses the parameter value X that maximizes the ratio L(X) / G(X).

2.5 Evaluation Metrics

There are four important values produced during predicting the class in which the evaluation samples belong, and these are the number of true positive (TP), true negative (TN), false positive (FP) and false negative (FN) predictions. The definition of the aforementioned values is given below:

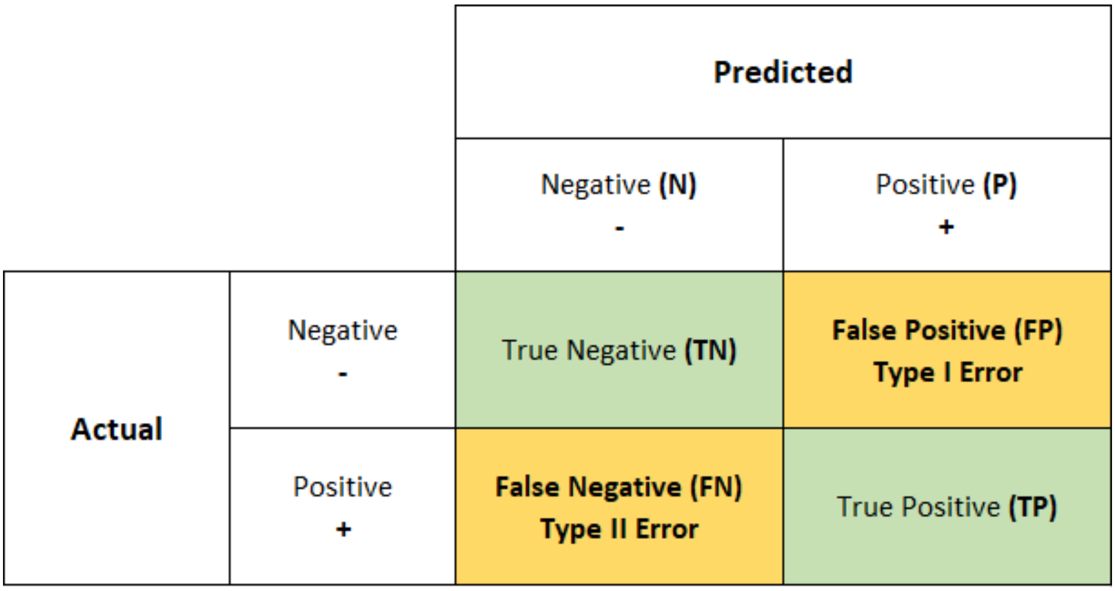
TP is a test result that correctly indicates the presence of a condition or characteristic.

TN is a test result that correctly indicates the absence of a condition or characteristic

FP is a test result which wrongly indicates that a particular condition or attribute is present.

FN is a test result which wrongly indicates that a particular condition or attribute is absent.

The above values are better depicted in a **confusion matrix** (figure 2.10). Each row of the matrix represents the instances in an actual class while each column represents the instances in a predicted class. The name stems from the fact that it makes it easy to see whether the system is confusing two classes. With the help of a confusion matrix, a variety of classification metrics can be calculated. The definition of the metrics utilized in the current thesis to evaluate the performance of each model are presented below:



		Predicted	
		Negative **(N)** -	Positive **(P)** +
Actual	Negative -	True Negative **(TN)**	**False Positive (FP)** **Type I Error**
	Positive +	**False Negative (FN)** **Type II Error**	True Positive **(TP)**

Figure 2.10: Confusion Matrix for binary classification

Accuracy is the ratio of the number of correct predictions to the number of total predictions made and can be calculated using the following formula:

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

However, accuracy, solely used, is not a good indicator of a model's performance when the dataset used is imbalanced, as it is in the present study. This is due to the fact that by classifying all the samples in the majority class ("healthy" in the current problem), very high accuracy results could be produced, but the developed model would have no discriminative ability between the two classes.

In health related tasks, a metric that is highly indicative of a model's performance is Sensitivity, or Recall or True Positive Rate. **Sensitivity** provides information about the number of positive (covid-19) samples correctly predicted as positive, out of the total number of samples belonging to the positive class.

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

Precision is another metric used, which calculates the number of correct predictions of samples belonging to the positive class out of the total number of samples predicted to belong to this class and is calculated using the following formula:

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

In contrast with Precision, Sensitivity is a metric of higher importance, since predicting a Covid positive sample as healthy can cause more undesirable consequences than predicting a healthy sample as Covid positive.

What is more, **Specificity** is another useful metric that is indicative of the number of negative samples predicted correctly by the classifier.

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

Another very useful metric that incorporates two of the above metrics and is suitable for imbalanced classification problems is the **F1-score**. The F1-score is the harmonic mean of the precision and recall. It thus symmetrically represents both precision and recall in one metric. The highest possible value of an F1-score is 1.0, indicating perfect precision and recall, and the lowest possible value is 0 if either precision or recall are zero. The formula to calculate the F1-score is the following:

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} = \frac{2 \times TP}{2 \times TP + FP + FN}$$

Area Under the Curve (AUC) metric is also calculated. The **AUC-ROC** curve (Area Under the Curve of Receiver Characteristic Operator) is a probability curve which plots the True Positive Rate (TPR) against the False Positive Rate (FPR) at various threshold

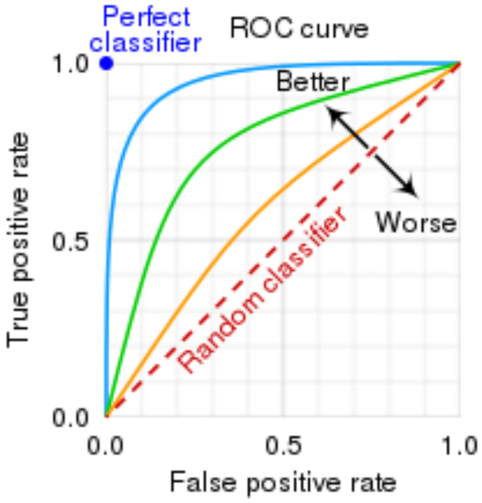


Figure 2.11: ROC curve

values, with the Area Under the Curve (AUC) value measuring the ability of a classifier to distinguish between the positive and negative class. The TPR equals the sensitivity value, while the FPR can be calculated using formula 2.14 and depicts the percentage of the negative class that was incorrectly classified.

$$FPR = 1 - Specificity$$

The value of the AUC ranges from 0.0 to 1.0, with 0.0 being that the model predicts all the positive samples as negatives and vice versa, while 1.0 means the model can perfectly distinguish the two classes. A value greater than 0.50 means that the model has some separation ability, with greater values depicting a better performance, while scores less than or equal to 0.5 depict a model with no class separation ability.

Lastly, **Balanced Accuracy** is another metric used in binary classification tasks. It is equal to the arithmetic mean of specificity and sensitivity. This metric is useful when dealing with imbalanced data, such as the task examined in the current thesis. The formula is given below:

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

3

Data Analysis - Preprocessing - Methods Implemented

3.1 Coswara Dataset

The Coswara[4] dataset [30], [31], [32] is a crowdsourced dataset containing 3 kinds of respiratory sounds; cough, breath and speech sounds, and metadata information for each user. The dataset is comprised of audio samples provided by 2746 different user ids. Each user submitted the following 9 different recordings, two types of cough sounds, heavy and shallow, two types of breath sounds, shallow and deep, two types of one to twenty digit counting, normal and fast and 3 different sustained vowel phonations. Each audio sample is accompanied by metadata information including demographic information such as, age, gender, country of origin etc., Covid-19 test type and the current health status of the user. All audio files have been manually assessed, with regard to the quality of the audio sample and the category it belongs to, by 13 annotators with each file being annotated once. The distribution of the Covid status labels (healthy, no respiratory illness exposed, respiratory illness not identified, positive mild, positive moderate, positive asymptomatic, fully recovered and under validation) is shown in figure 3.1.

[4] https://github.com/iiscleap/Coswara-Data

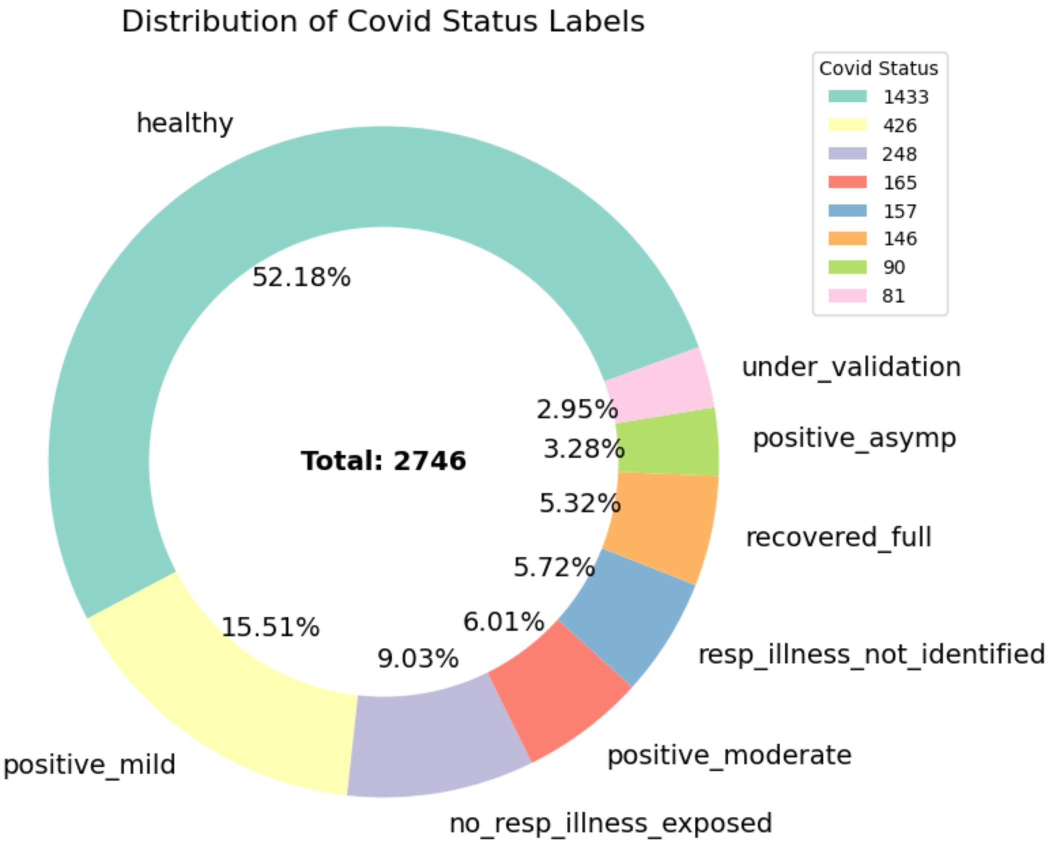


Figure 3.1: Covid-19 status distribution of the recordings in the Coswara dataset

Furthermore, from the provided metadata it is observed that 69.19% of the users identify as male, 30.74% as female and 0.07% as other. Regarding the country of origin of the participants, the vast majority of samples are from India (91.59%), 3.17% from the United States and the rest 5.24% are from other countries. In addition, from the accompanying metadata information it can be derived that over half of the data were recorded in 2020 (54.26%), a little over a quarter (28.04%) in 2021 and the last 17.70% in 2022. Last but not least, due to the completion of the rest of the metadata information not being mandatory [31] when submitting the recordings, meaningful distributions about the medical background of the contributors cannot be produced. However, it should be noted that when participants were asked whether they were a returning user or not, 72.94% stated no, while only 2.29% replied yes and 24.76% did not answer the question. The above information is schematically presented in figure 3.2.

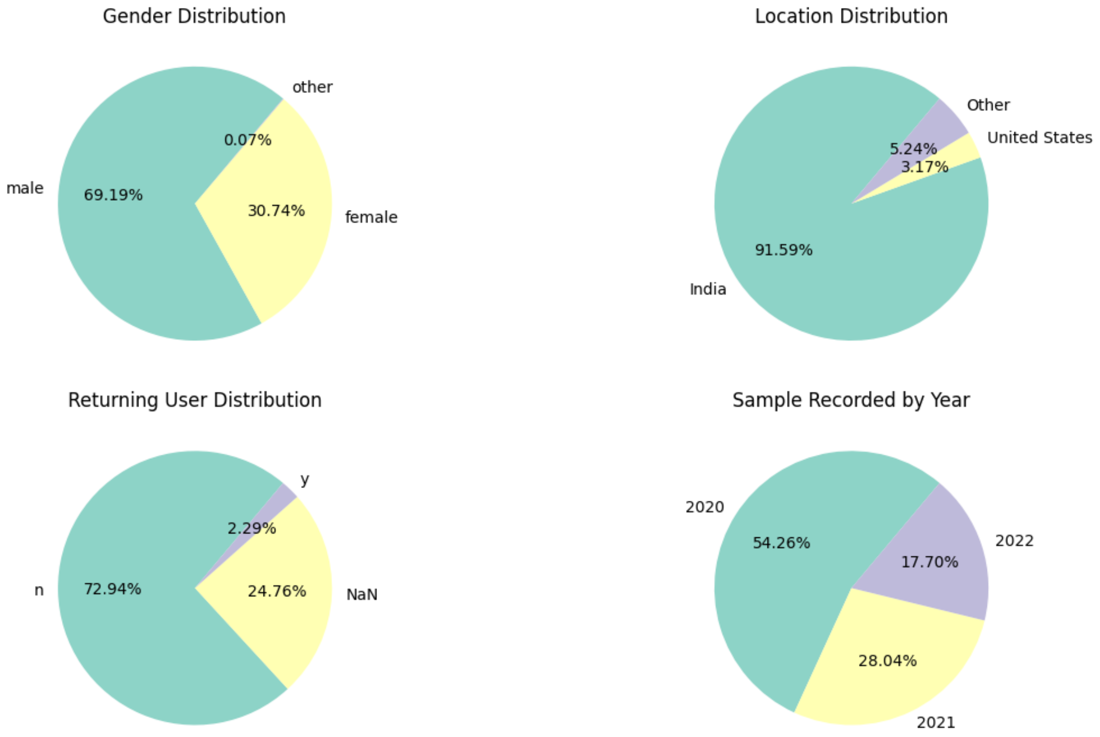


Figure 3.2: Metadata Statistics for the Coswara dataset

3.1.1 Data Cleaning

For the purpose of this study, only the two types of cough samples (heavy, shallow) were utilized in a single dataset. The samples with a status in one of the three categories i.e. positive mild, positive moderate and positive asymptomatic are classified as positive, the samples declared healthy remain as is and the rest of the samples are discarded. The constituting distributions regarding the Covid status and the metadata statistics are illustrated in figures 3.3 and 3.4 respectively.

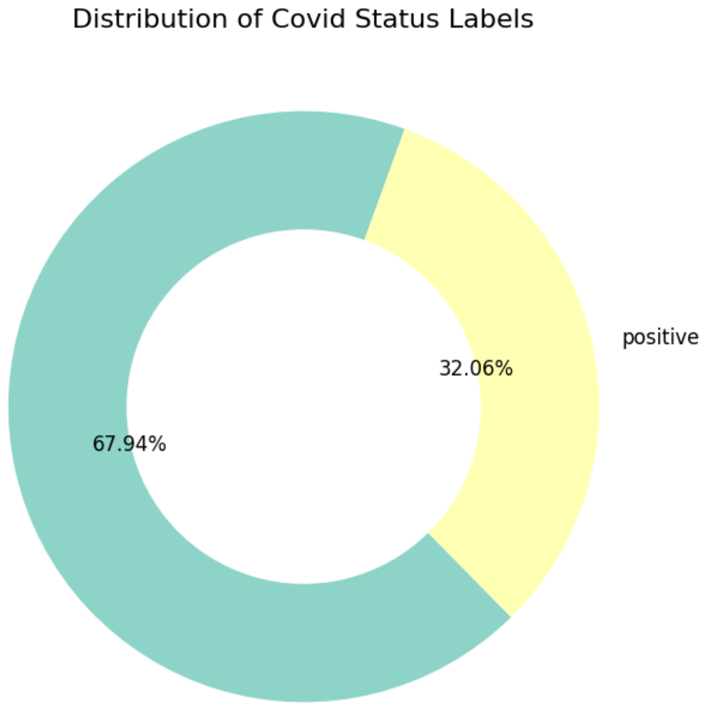


Figure 3.3: Label distribution

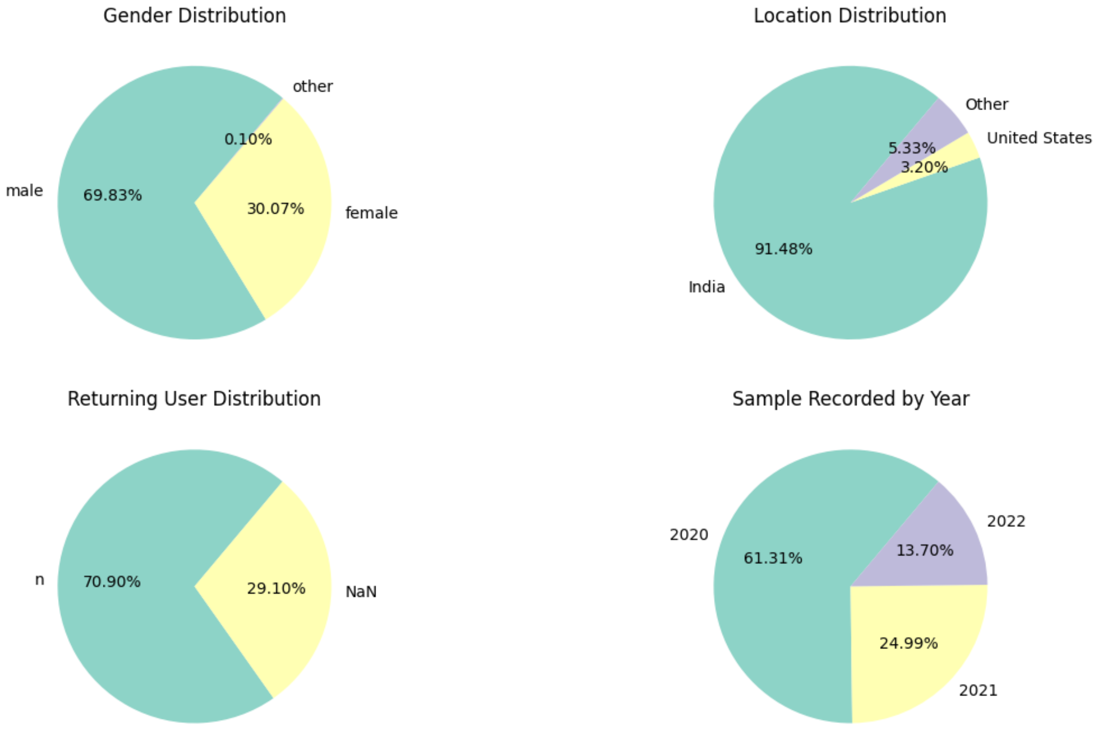


Figure 3.4: Processed metadata Statistics distribution

What is more, the figure 3.5 showcases the healthy – positive label distribution of the participants by month. In order to apply concept drift adaptation methods, the data recorded from October 2021 are going to be used as a drift set and thus will be excluded from the train, validation and test sets.

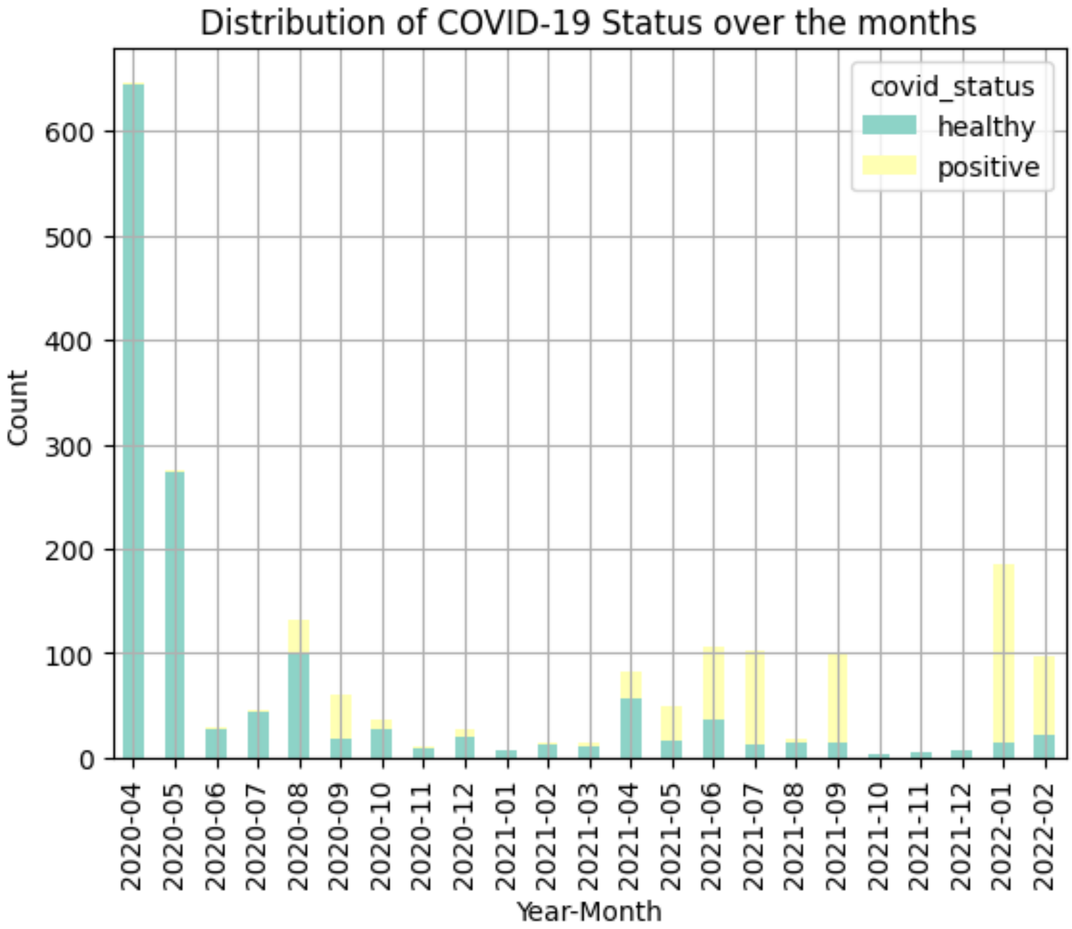


Figure 3.5: Distribution of labels over the months

In order to reduce the total duration of the audio data for training, silence trimming was implemented. This was achieved by splitting the audio of each recording into its non-silent intervals, using a threshold of 30 dB as a criterion to distinguish silent from non-silent intervals, and the remaining segments were concatenated in order to reconstruct the recording. By using this method, non-essential audio was discarded, and the total duration of the recordings was reduced.

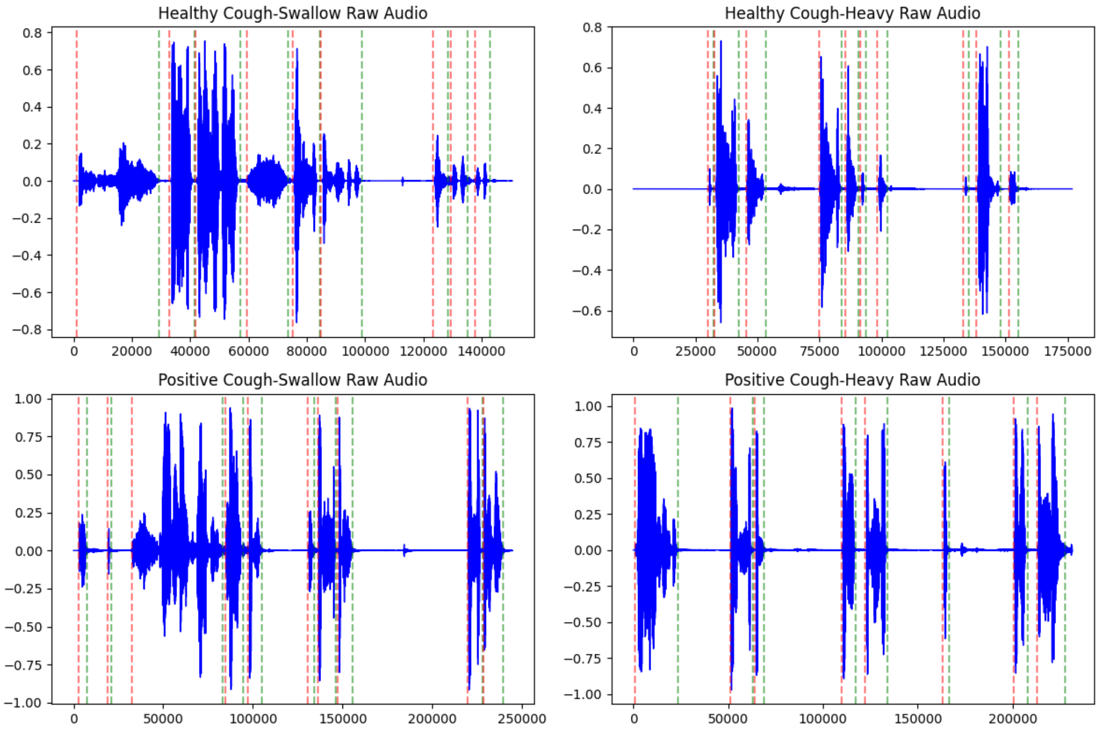


Figure 3.7: Detection of non-silent segments. The red line depicts the start of the non-silent event and the green line the end.

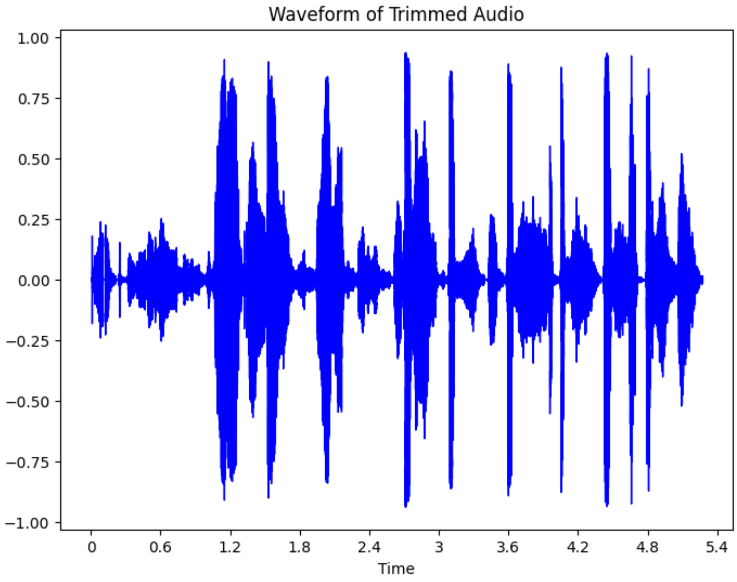


Figure 3.6: The reconstructed Positive Cough-Shallow recording

Finally, 66 recordings were discarded due to the fact that they either contained 0 seconds of audio (60 recordings), or their duration was less than 0.35 seconds with irrelevant audio (6 recordings which contained voices or indistinguishable sounds). The remaining data were then separated into 4 sets, namely the train set, validation set, test set and drift set. The resulting data distribution is showcased in the following figure 3.8.

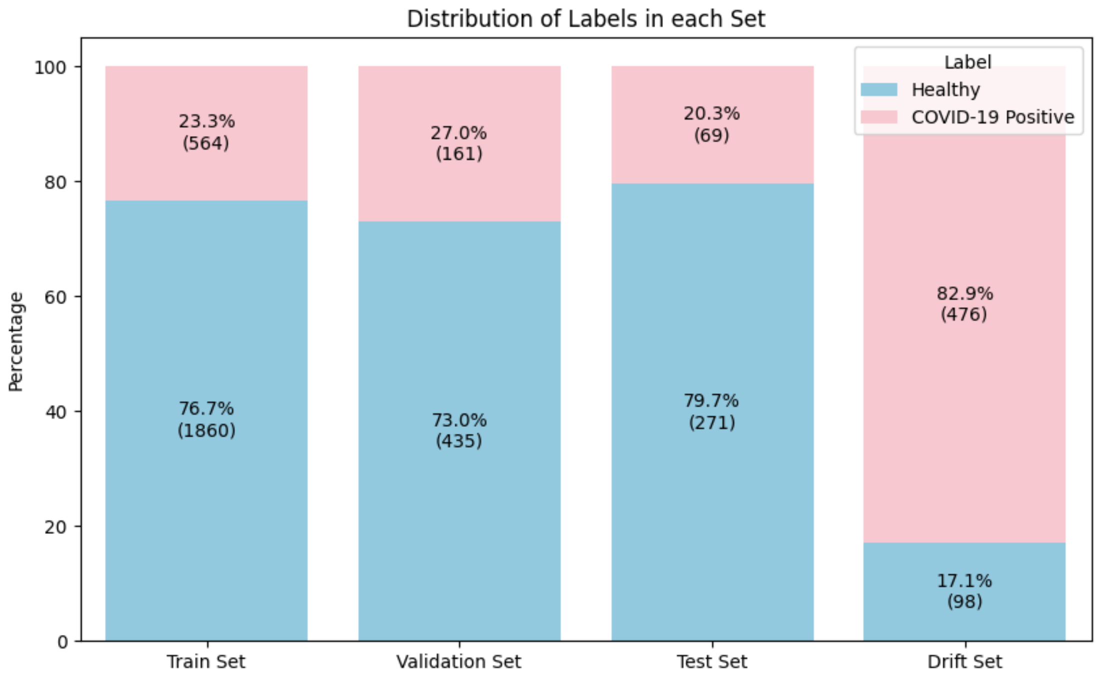


Figure 3.8: Final data split

3.1.2 Feature Extraction

For the classification task to be performed, features are extracted from the cough recordings. The features utilized in this thesis are MFCCs, MFCC deltas, MFCC delta-deltas, zero – crossing rate, kurtosis, and the root mean square value. In order to handle the problem of recordings having various duration lengths, the method proposed by M. Pahar et.al [16] was used. Particularly, a fixed number of features F is extracted from each recording by implementing the *hop length* to be dependent on the length of the audio timeseries extracted when loading the recording and the *samples per segment* to be dependent on the duration of the audio. By applying the aforementioned method, the *number of mfcc vectors per segment* is going to be the same and no padding is needed. The *librosa*[5] library was used to handle the audio data feature extraction, along with *scipy stats*[6] for the extraction of the kurtosis. For the librosa functions, the *number of*

[5] https://librosa.org/doc/latest/index.html

[6] https://docs.scipy.org/doc/scipy/reference/stats.html

segments was set to 100, the *n_fft* parameter (length of the windowed signal after padding with zeros) was set to 1024 and the *sampling rate* was set to 2250. In total 42 features were extracted, 13 MFCCs, 13 MFCC Deltas, 13 MFCC Delta – deltas, 1 ZCR, 1 Kurtosis and 1 RMS for each one of the 100 segments, which accumulates to a *feature shape* of (42, 100) for each recording.

3.1.3 Mel – Spectrograms

For the deep learning models, Mel–spectrograms were extracted using the *librosa.feature.melspectrogram* function. The *sampling rate* was set to 22050, the *n_fft* to 1024, the *number of segments* to 100 and last but not least, the formula for the *hop length* is the same as above.

3.2 Classification Methods

For the proposed classification task, 6 different models were examined. MLP and Random Forest models, 3 pre – trained CNNs namely, VGG–16, ResNet–50, Inception ResNet–V2 and CLIP, a zero – shot classifier.

3.2.1 Multilayer Perceptron – MLP

Firstly, the data are standardized using the *Standard Scaler* [7] from the *sickit learn* library. This scaler standarizes the features by removing the mean and scaling to unit variance.

For this model, Optuna hyperparameter optimization framework was used in order to find the best hyperparameters for the model. The following table 3.1 showcases the search space for the different hyperparameters in the implemented *objective* function. *Early stopping* is set to *True,* and the function returns the auroc score of the validation set. The study is set to run for 70 trials with the direction to maximize the return value

[7] https://scikitlearn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html

of the objective function. In addition, the TPE sampler and the Hyperband pruner [33] are utilized. Finally, the model with the highest auroc score on the validation set is selected as the final model to be tested on the test set.

Table 3.1: MLP Hyperparameter Search Space

Parameter	Search space
n_layers	1 – 10, step 1
hidden_layer_sizes	32, 64, 128, 256, 512, 1024, 2048
activation	relu, tanh
solver	sgd, adam
alpha	0.00001 – 0.1
learning_rate	constant, invscaling, adaptive
learning_rate_init	0.00001 – 0.1
max_iter	100 – 1000, step 100

The same set-up is used to run another 70 trials but this time the Synthetic Minority Oversampling Technique, or SMOTE is applied. This oversampling technique is used to handle the lack of Covid–19 samples (minority class) and thus assist the model in effectively learning the decision boundary between the two classes. Specifically, SMOTE works by choosing a random example from the minority class, then *k* of the nearest neighbors for that example are found. A randomly selected neighbor is chosen and a synthetic example is created at a randomly selected point between the two examples in the feature space. By the end of this process the training data will be balanced. The variable k is set to 5.

3.2.2 Random Forests

Firstly, the data are standardized using the *Standard Scaler* from the *sickit learn* library. This scaler standardizes the features by removing the mean and scaling to unit variance.

For this model, Optuna hyperparameter optimization framework was used in order to find the best hyperparameters for the model. The following table 3.2 showcases the search space for the different hyperparameters in the implemented *objective* function.

Early stopping is set to *True,* and the function returns the roc–auc score of the validation set. The study is set to run for 70 trials with the direction to maximize the return value of the objective function. In addition, the TPE sampler and the Hyperband pruner are utilized. Finally, the model with the highest auroc score on the validation set is selected as the final model to be tested on the test set.

Table 3.2: Random Forest Hyperparameter Search Space

Parameter	Search space
n_estimators	100 – 4000, step 100
max_depth	5 – 20, step 1
min_samples_slpit	2 – 10, step 1
min_samples_leaf	1 – 10, step 1
max_features	sqrt, log2
class_weight	balanced, balanced_subsample
criterion	gini, entropy, log_loss

The same set-up is used to run another 70 trials but this time the Synthetic Minority Oversampling Technique or SMOTE is applied with variable k set to 5.

3.2.3 Transfer Learning

Due to the plethora of pre–trained CNNs that have achieved great results in the ImageNet dataset [34], it was decided that their gained knowledge could be exploited in order to improve generalization for the current task. In total 6 different models are going to be created using 3 pre – trained CNNs as base-models. Three of the six models are going to have a Gaussian Noise layer as a hidden layer in the architecture, while for the rest the GS will used as the input layer. In that way we can compare how the location of the GS layer affects the model's performance.

The first model utilized as a base-model for the classifier is **VGG–16** from the *keras* applications library. The model is loaded with the pretrained weights from the ImageNet classification task, but without the top layer. All the layers are frozen so that they are not trainable, and to avoid destroying any of the information they contain during future training rounds. The model will learn to provide predictions for our current task by adding a few trainable layers on top of the frozen layers. In this case the layers added are a Gaussian Noise layer with a standard deviation value of the noise distribution set at 0.1. Then a Global Average Pooling 2D layer was added, with 2 Dense layers completing the model. The first Dense layer consists of 1024 units and a "ReLu" activation function, while the second consists of 2 units (one for each class) and the "softmax" activation function. The model architecture is depicted at Figure 3.9.

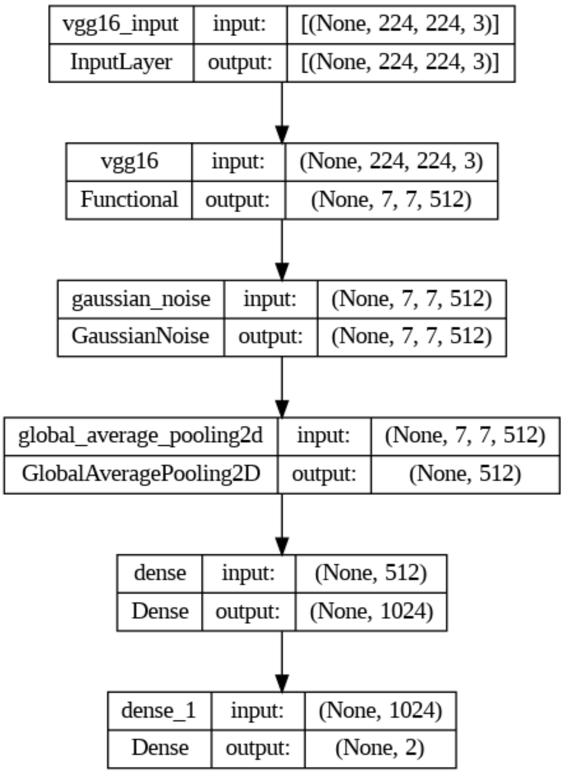

vgg16_input | input: | [(None, 224, 224, 3)]
InputLayer | output: | [(None, 224, 224, 3)]
vgg16 | input: | (None, 224, 224, 3)
Functional | output: | (None, 7, 7, 512)
gaussian_noise | input: | (None, 7, 7, 512)
GaussianNoise | output: | (None, 7, 7, 512)
global_average_pooling2d | input: | (None, 7, 7, 512)
GlobalAveragePooling2D | output: | (None, 512)
dense | input: | (None, 512)
Dense | output: | (None, 1024)
dense_1 | input: | (None, 1024)
Dense | output: | (None, 2)

Figure 3.9: CNN with VGG-16 base model and hidden GS layer

The model is compiled using the "Adam" optimizer with a learning rate of 0.001 and binary cross entropy loss. The data are rescaled using "ImageDataGenerator" and setting the rescale parameter to 1.0/255.0. Lastly, the model is fitted for 100 and 200 epochs, with class weights set to the ratio of healthy to positive samples, in order to handle the class imbalance problem.

The same process is repeated with the exception that the GS layer is now set as the input layer instead of the VGG – 16 base-model. The model architecture is shown at figure 3.10.

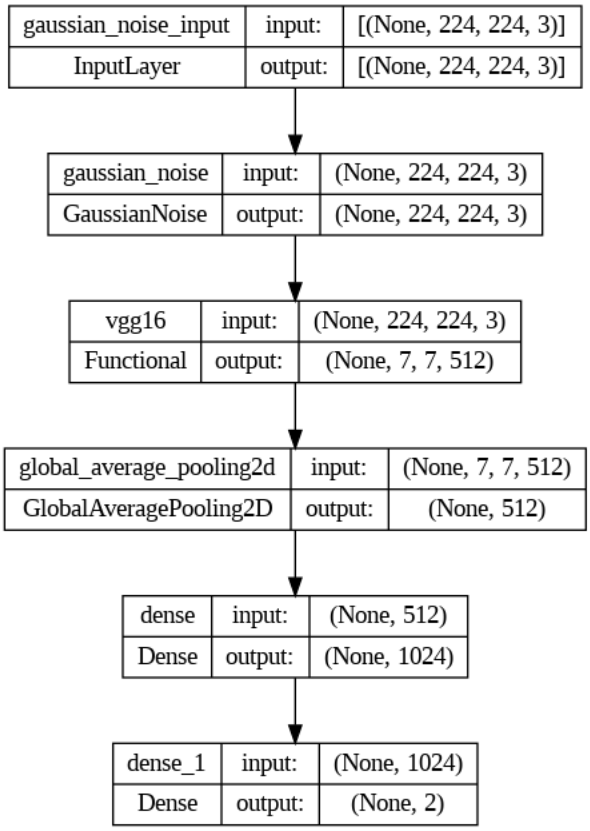


Figure 3.10: CNN with VGG-16 base model and input GS layer

The second model utilized as a base-model for the classifier is the **ResNet–50** from the *keras* applications library. The model is loaded with the pretrained weights from the ImageNet classification task, but without the top layer. All the layers are frozen so that they are not trainable, and to avoid losing any of the information they contain during future training rounds. The model will learn to provide predictions for our current task by adding a few trainable layers on top of the frozen layers. In this case the layers added are a Gaussian Noise layer with a standard deviation value of the noise distribution set at 0.1. Then a Global Average Pooling 2D layer was added, with 2 Dense layers completing the model. The first Dense layer consists of 1024 units and a “ReLu” activation function, while the second consists of 2 units (one for each class) and the “softmax” activation function. The model architecture is depicted at figure 3.11. The model is compiled using the “Adam” optimizer with a learning rate of 0.001 and binary cross entropy loss. The data are rescaled using the ImageDataGenerator and setting the rescale parameter to 1.0/255.0. Lastly, the model is fitted for 100 and 200 epochs, with class weights set to the ratio of healthy to positive samples, in order to handle the class imbalance problem.

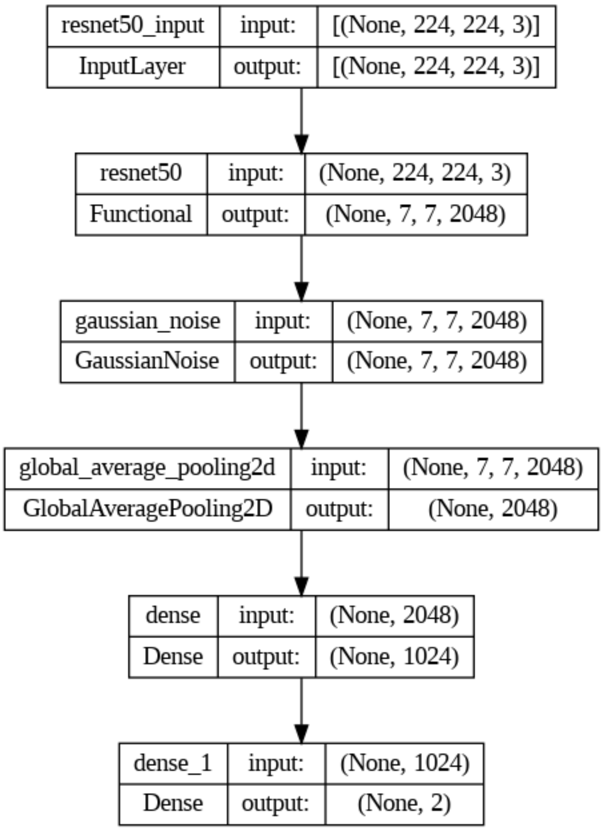

resnet50_input | input: | [(None, 224, 224, 3)]
InputLayer | output: | [(None, 224, 224, 3)]
resnet50 | input: | (None, 224, 224, 3)
Functional | output: | (None, 7, 7, 2048)
gaussian_noise | input: | (None, 7, 7, 2048)
GaussianNoise | output: | (None, 7, 7, 2048)
global_average_pooling2d | input: | (None, 7, 7, 2048)
GlobalAveragePooling2D | output: | (None, 2048)
dense | input: | (None, 2048)
Dense | output: | (None, 1024)
dense_1 | input: | (None, 1024)
Dense | output: | (None, 2)

Figure 3.11: CNN with ResNet-50 base model and hidden GS layer

The same process is repeated with the exception that the GS layer is now set as the input layer instead of the ResNet – 50 base-model. The model architecture is shown at figure 3.12.

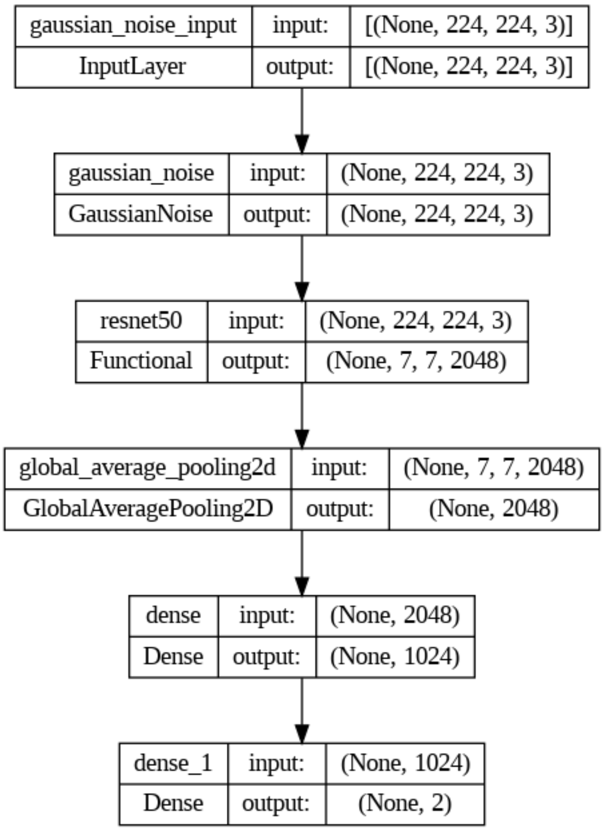


Figure 3.12: CNN with ResNet-50 base model and input GS layer

The third model utilized as a base-model for the CNN is the **Inception ResNet–V2** from the *keras* applications library. The model is loaded with the pretrained weights from the ImageNet classification task, but without the top layer. All the layers are frozen so that they are not trainable, and to avoid destroying any of the information they contain during future training rounds. The model will learn to provide predictions for our current task by adding a few trainable layers on top of the frozen layers. These new layers will learn to turn the old features into predictions on the new dataset. In this case the layers added are a Gaussian Noise layer with a standard deviation value of the noise distribution set at 0.1. Then a Global Average Pooling 2D layer was added, with 2 Dense layers completing the model. The first Dense layer consists of 1024 units and a “ReLu” activation function, while the second consists of 2 units (one for each class) and the “softmax” activation function. The model architecture is depicted at figure 3.13.

The model is compiled using the “Adam” optimizer with a learning rate of 0.001 and binary cross entropy loss. The data are rescaled using the ImageDataGenerator and setting the rescale parameter to 1.0/255.0. Lastly, the model is fitted for 100 and 200 epochs, with class weights set to the ratio of healthy to positive samples, in order to handle the class imbalance problem.

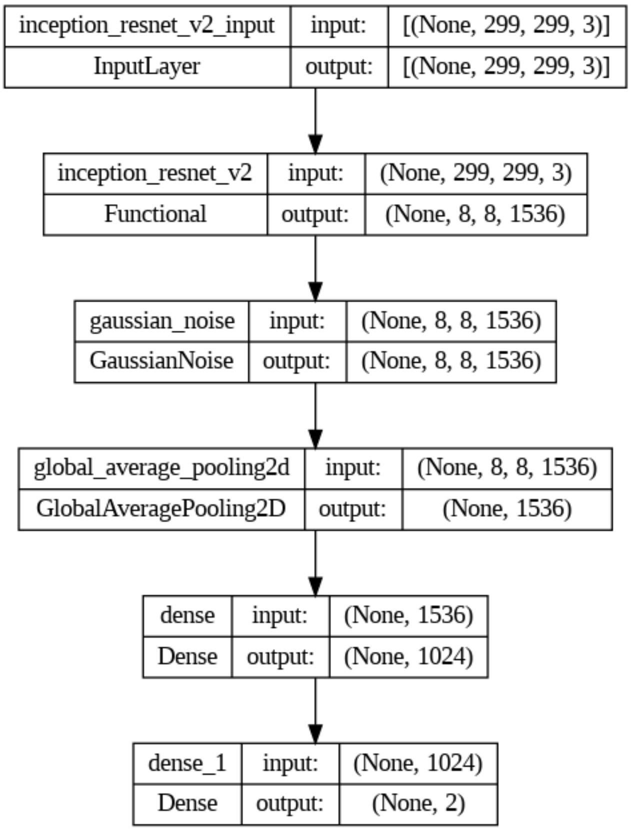

inception_resnet_v2_input | input: | [(None, 299, 299, 3)]
InputLayer | output: | [(None, 299, 299, 3)]
inception_resnet_v2 | input: | (None, 299, 299, 3)
Functional | output: | (None, 8, 8, 1536)
gaussian_noise | input: | (None, 8, 8, 1536)
GaussianNoise | output: | (None, 8, 8, 1536)
global_average_pooling2d | input: | (None, 8, 8, 1536)
GlobalAveragePooling2D | output: | (None, 1536)
dense | input: | (None, 1536)
Dense | output: | (None, 1024)
dense_1 | input: | (None, 1024)
Dense | output: | (None, 2)

Figure 3.13: CNN with Inception-ResNet-V2 base model and hidden GS layer

Lastly, the same process is repeated with the exception that the GS layer is now set as the input layer instead of the Inception – ResNet – V2 base-model. The model architecture is shown at figure 3.14.

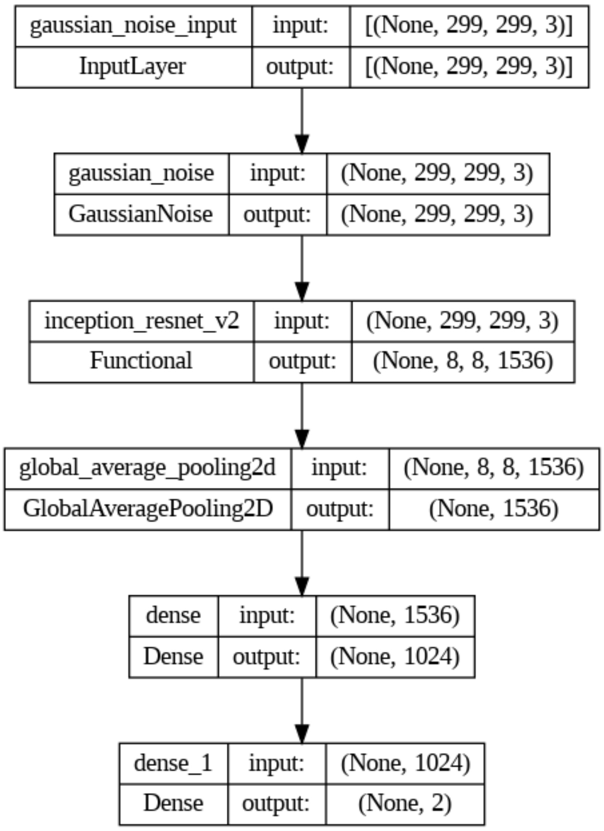


Figure 3.14: CNN with Inception-ResNet-V2 base model and input GS layer

3.2.4 Zero – Shot Learning

The open-source framework OpenAI CLIP (Contrastive Language-Image Pretraining) and the PyTorch library are utilized for this section. A pre-trained model using OpenAI CLIP with the ViT-B-32 architecture is selected and the pretrained weights ('laion2b_s34b_b79k') are specified. Furthermore, the tokenizer associated with the chosen model is retrieved for processing text inputs. The text-label pair to be tested is tokenized and then the image and text features are extracted from the model. Subsequently, softmax probabilities of the text being associated with the image are calculated and the predicted class is determined by the index of the maximum probability. Finally, the model's performance against the ground truth labels is evaluated by calculating the necessary metrics. The different text-label pairs tested on the validation set are displayed in the following Table 3.3.

Table 3.3: CLIP Text - label pairs tested on the validation set

Pair index	Healthy	Covid – 19
0	Negative recording	Positive recording
1	Negative Mel-spectrogram recording	Positive Mel-spectrogram recording
2	Mel-spectrogram of a cough recording of a covid-19 negative person	Mel-spectrogram of a cough recording of a covid-19 positive person
3	Negative Mel-spectrogram of a cough recording of covid-19	Positive Mel-spectrogram of a cough recording of covid-19
4	Negative Mel-spectrogram of a cough recording	Positive Mel-spectrogram of a cough recording
5	covid-19 negative	covid-19 positive
6	image of audio from covid-19 negative individual	image of audio from covid-19 positive individual
7	healthy	Covid-19 positive

3.3 Concept Drift Adaptation Methods

The model with the highest roc-auc score on the test set will be selected to be tested on the drift set, in order to establish whether a concept drift is present or not. Then, the drift set is split into drift-training, drift-validation and drift-test sets, with the selected model being retrained on the drift-training set. The resulting data distribution is showcased in figure 3.15. During retaining freezing the model and training only layer dense_1 will be examined, as well as adding a regularization factor in the model's weights. To be more specific, the regularization factor is going to be the difference of the new weights minus the old weights of the layer (absolute or squared) and multiplied by a constant with values 0.01, 0.1 or 1.0.

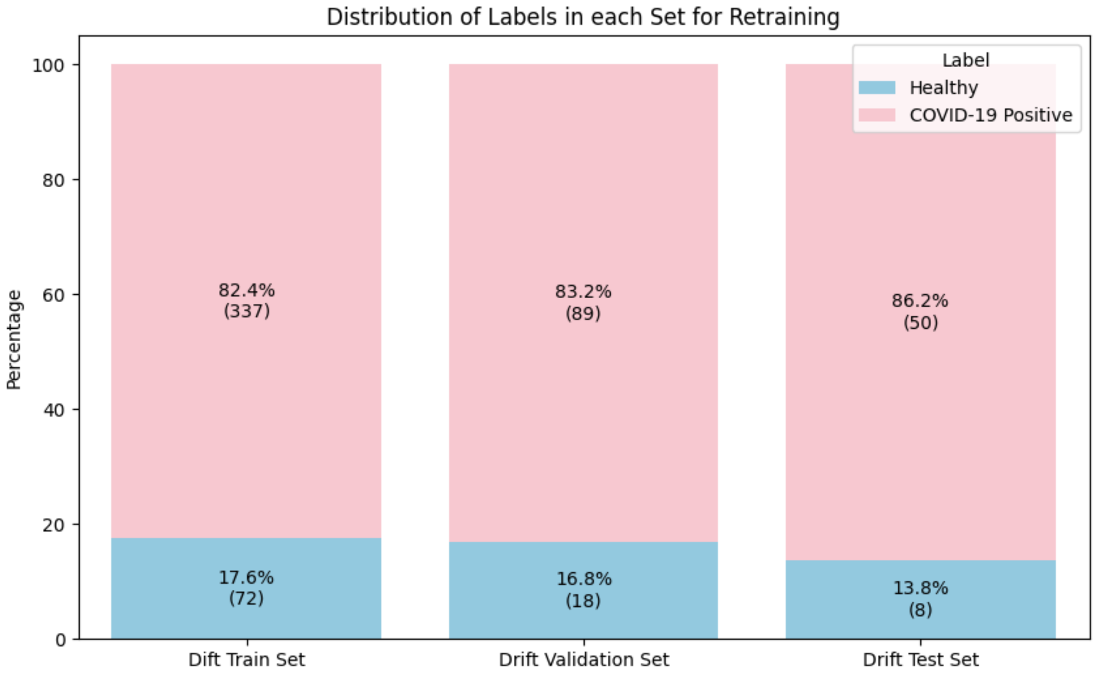


Figure 3.15: Drift data split for retraining

3.3.1 Retraining only the final dense layer

All the layers except the final dense layer (dense_1) are frozen and the following layer regularization methods (LRM) are investigated using the layer's weights, which are initially stored as old $_{\text{weights}}$. The model is retrained for 100 and 200 epochs.

Table 3.4: Regularization methods

LRM	Equation
1	$new_{weights}[0] = 1.0 \times \lvert new_{weights}[0] - old_{weights}[0] \rvert$
2	$new_{weights}[0] = 0.1 \times \lvert new_{weights}[0] - old_{weights}[0] \rvert$
3	$new_{weights}[0] = 0.01 \times \lvert new_{weights}[0] - old_{weights}[0] \rvert$
4	$new_{weights}[0] = 1.0 \times \left(new_{weights}[0] - old_{weights}[0]\right)^2$
5	$new_{weights}[0] = 0.1 \times \left(new_{weights}[0] - old_{weights}[0]\right)^2$
6	$new_{weights}[0] = 0.01 \times \left(new_{weights}[0] - old_{weights}[0]\right)^2$

3.3.2 Retraining the entire model

The selected model's last two dense layers (dense and dense_1) are retrained with the regularization methods shown in table 3.4 being applied each layer. The model is retrained for 100 and 200 epochs.

3.4 Classification Threshold Optimization

A model trained for a binary classification task, like the ones mentioned above, return a probability score of the target variable that indicates how likely it is that the sample belongs to each class. The standard threshold used to determine whether the sample belongs to the positive class, or not, is 0.5. By varying the classification threshold, the classifier's performance changes, since the values of TP, TN, FP, FN displayed in the confusion matrix also change.

Since the 0.5 value is not always the ideal threshold, and in addition to the fact that the task at hand is that of an imbalanced classification, the last step in the classification process will be to examine whether optimizing the classification threshold will improve the classifiers' performance. This is achieved by calculating the ROC curve's thresholds from the validation set and then evaluating the model's performance at each threshold based on the balanced accuracy scoring method. Finally, the threshold that maximizes the balanced accuracy score is chosen as the new decision threshold, and the model is tested on the test set.

4

Results – Discussion

4.1 Models trained and tested using Extracted Features

The classification results obtained from the training of the 2 models described in sections 3.3.1 and 3.3.2 using features extracted from the audio recordings, are showcased in tables 4.1 and 4.2. The values of the metrics presented are acquired from the best performing model after 70 trials using Optuna. The model with the highest AUROC value along with the model with the highest sensitivity have been highlighted.

Table 4.1: Performance metrics for the RF and MLP models

Model	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC
MLP	80.00%	5.80%	98.89%	52.35%	66.25%
RF	**78.82%**	**5.80%**	**97.42%**	**51.61%**	**69.91%**

Table 4.1 illustrates that the RF model achieves the highest AUROC score with a value of 69.91% and balanced accuracy score of 51.61%. The balanced accuracy score is a result of high disparity between the sensitivity and specificity metrics. Furthermore, the sensitivity value for both models is 5.80%, indicating that the models cannot predict the positive (Covid–19) class correctly. The low sensitivity scores coupled with the very high specificity scores illustrate that once more both the MLP and the RF models cannot

discriminate between the two classes and classify almost everything to the majority class (Healthy).

Table 4.2: Performance metrics for the RF and MLP models with SMOTE

Model	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC
MLP	74.41%	**40.58%**	83.03%	61.80%	65.19%
RF	**79.41%**	**18.84%**	**94.83%**	**56.84%**	**69.42%**

Table 4.2 illustrates that the RF model once more achieves the highest AUROC score with a value of 69.42% and balanced accuracy score of 56.84%. The application of SMOTE on the training data improved the sensitivity and balanced accuracy metrics. The former value of the RF model is 18.84% and of the MLP model is 40.58%. Although the sensitivity metric is considerably higher for both models, it is still not adequate enough to predict the positive (Covid–19) class correctly. The low sensitivity scores coupled with the very high specificity scores illustrate that both the MLP and the RF models cannot discriminate between the two classes and classify almost everything to the majority class (Healthy). Overall, applying SMOTE significantly improved the sensitivity (by 600.86% for the MLP and 225.38% for the RF model) and balanced accuracy score (by 18.05% for the MLP and 10.13% for the RF model) metrics of the two models, while decreasing specificity and slightly reducing AUROC.

Table 4.3: Performance metrics for the RF and MLP models with new thresholds

Model	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC	Threshold
MLP	72.94%	44.93%	90.07%	62.50%	66.25%	0.202
RF	**66.47%**	**47.83%**	**71.22%**	**59.52%**	**69.91%**	**0.367**

Table 4.3 showcases that the application of classification threshold optimization had a significant impact on the sensitivity and balanced accuracy values, achieving higher scores than before. The RF model achieves the highest AUROC score with a value of 69.91% (the same as in table 4.1, as is expected) and balanced accuracy score of

59.52%, a 15.33% increase when compared to table 4.1. The MLP model shows an 674.66% improvement in sensitivity and 19.39% in balanced accuracy score, while the RF model achieves an 47.83% score on the sensitivity metric (an improvement of 724,66%).

Table 4.4: Performance metrics for the RF and MLP models with SMOTE and new thresholds

Model	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC	Threshold
MLP	74.71%	40.58%	83.39%	61.99%	65.13%	0.674
RF	**65.00%**	**59.42%**	**66.42%**	**62.92%**	**69.42%**	**0.387**

Finally, when both SMOTE and threshold optimization are utilized, a significant improvement is observed on both sensitivity and balanced accuracy values. To be more precise, the RF model reaches a sensitivity score of 59.42% and a balanced accuracy score of 62.92%, with both values being the highest achieved scores in those two metrics. Although the AUROC score is marginally lower than the one reached without the use of SMOTE, this RF model is much better at predicting the positive (Covid–19) class correctly. Sensitivity improved more than 10 times when compared to table 4.1 and balanced accuracy score is also 21.33% higher, when compared to the RF model that does not use neither SMOTE nor threshold optimization, and 10.70% higher than the model that utilizes only SMOTE.

4.2 Models trained and tested using Mel – Spectrograms

The classification results obtained from the training of the 6 models described in section 4.3.3 using Mel – Spectrograms, are showcased in tables 5.5 and 5.6. The values of the metrics presented have been acquired by training each model for 100 and 200 epochs. The two models with the highest AUROC values along with the model with the highest sensitivity have been highlighted. Tables 4.7 and 4.8 depict the new metric values after the implementation of classification threshold optimization. The last column of the two aforementioned tables display the optimal threshold value selected, using the validation set.

Table 4.5: Performance metrics for the hidden GS - CNN models

Base – Model	Epoch	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC
VGG – 16	100	67.94%	69.57%	67.53%	68.55%	76.65%
	200	**78.53%**	**62.32%**	**82.66%**	**72.49%**	**80.21%**
ResNet	100	64.12%	73.91%	61.62%	67.77%	71.34%
	200	55.88%	**84.06%**	48.71%	66.38%	73.93%
Inception	100	78.24%	57.97%	83.39%	70.68%	77.14%
	200	**77.65%**	**63.77%**	**81.18%**	**72.47%**	**78.74%**

Table 4.5 illustrates that the CNN with the VGG–16 as base model and trained for 200 epochs achieves the highest AUROC score, with a value of 80.21%. The second and third highest AUROC values are achieved by the architecture using Inception as a base model and trained for 200 and 100 epochs respectively. The balanced accuracy values range between 66.38% to 72.49%. This indicates a good distinction between the positive and negative class. What is more, the sensitivity value fluctuates from 57.97% to 84.06% depending on the base model and the number of epochs trained. Since the task examined in the current thesis is Covid–19 diagnosis the sensitivity metric is important because it indicates whether the model can predict the positive (Covid–19) class correctly. The architecture with the highest sensitivity value was trained for 200 epochs and used the ResNet – 50 model as the base-model. Regarding the different training epochs, it should be noted that VGG -16 shows significant increase in accuracy, specificity, balanced accuracy and AUROC, while sensitivity decreases by 7 points. Overall, the performance of the model improves. Similarly, the Inception model achieves better results with additional training, although accuracy and specificity slightly decline. Finally, ResNet exhibits a trade-off with increased sensitivity but decreased specificity and overall accuracy with more training. The slight increase in AUROC suggests better performance in distinguishing between classes despite lower accuracy.

Table 4.6: Performance metrics for the input GS - CNN models

Base – Model	Epoch	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC
VGG – 16	100	**74.12%**	**60.87%**	**77.49%**	**69.18%**	**74.46%**
	200	73.82%	46.38%	80.81%	63.59%	71.07%
ResNet	100	68.82%	**65.22%**	69.74%	67.48%	72.53%
	200	**72.65%**	**62.32%**	**75.28%**	**68.80%**	**74.82%**
Inception	100	79.12%	30.43%	91.51%	60.97%	68.31%
	200	75.29%	49.28%	81.92%	65.60%	69.41%

Table 4.6 illustrates that the CNN with the ResNet–50 as base model and trained for 200 epochs achieves the highest AUROC score. The second and third highest AUROC values are achieved by the architecture using VGG-16 and ResNet–50 as base models and trained for 100 respectively. The balanced accuracy values range between 60.97% to 69.18%, depicting a higher disparity and lower scores than the models applying the Gaussian Noise layer as a hidden layer. What is more, the sensitivity value fluctuates from 30.43% to 65.22% depending on the base model and the number of epochs trained, which illustrate a significant decrease when compared to the models using the GS as a hidden layer (Table 4.5). The same can be observed for the acquired AUROC values. As the epochs increase, VGG – 16 shows an increase in specificity and a decrease in sensitivity suggesting a trade-off where the model becomes more confident in correctly classifying the negative cases, at the cost of misclassifying the positive ones. On the other hand, Inception displays an increase in sensitivity and a decrease in specificity. Lastly, ResNet seems to be more robust to additional training epochs, since most metrics show a consistent improvement.

Table 4.7: Performance metrics for the hidden GS - CNN models with new threshold

Base – Model	Epoch	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC	Threshold
VGG – 16	100	74.12%	62.32%	77.12%	69.72%	76.65%	0.827
	200	**74.41%**	**68.12%**	**76.01%**	**72.07%**	**80.21%**	**0.860**
ResNet	100	67.35%	62.32%	68.63%	65.48%	71.34%	0.568
	200	70.00%	63.77%	71.59%	67.68%	73.93%	0.742
Inception	100	70.29%	63.77%	71.96%	67.86%	77.14%	0.014
	200	**71.18%**	**69.57%**	**71.59%**	**70.58%**	**78.74%**	**0.003**

Table 4.7 illustrates the performance metrics for the hidden GS – CNN models with the new optimal thresholds applied in the metrics' calculation. The balanced accuracy values range between 65.48% to 72.07%. This indicates a good distinction between the positive and negative class, with a slight drop being evident in the minimum and maximum values. What is more, the sensitivity value fluctuates from 62.32% to 69.57% depending on the base model and the number of epochs trained. With regards to the model with the highest AUROC score, an improvement of 9.31% can be observed in the sensitivity value. Overall, the application of an optimal threshold derived from the validation set seems to have a varying effect in the metrics' values, which is not consistent from model to model or metric to metric.

Table 4.8: Performance metrics for the input GS - CNN models with new threshold

Base – Model	Epoch	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC	Threshold
VGG – 16	**100**	**72.65%**	**72.46%**	**72.69%**	**72.58%**	**74.46%**	**0.998**
	200	74.41%	46.38%	81.55%	63.96%	71.07%	0.978
ResNet	100	69.71%	62.32%	71.59%	66.95%	72.53%	0.864
	200	**70.88%**	**63.77%**	**72.69%**	**68.23%**	**74.82%**	**0.918**
Inception	100	66.47%	60.87%	67.90%	64.38%	68.31%	0.003
	200	70.29%	60.87%	72.69%	66.78%	69.41%	0.019

Table 4.8 illustrates the performance metrics for the input GS – CNN models with the new optimal thresholds applied in the metrics' calculation. The balanced accuracy values range between 64.38% to 72.58%. This indicates a good distinction between the positive and negative class, with models VGG – 16 and Inception scoring higher than before. What is more, the sensitivity value fluctuates from 46.38% to 72.46% depending on the base model and the number of epochs trained and are significantly improved in 4 out of the six models. Overall, the application of an optimal threshold derived from the validation has improved the values of sensitivity and balanced accuracy metrics.

4.3 Zero – Shot Learning using Mel – Spectrograms

The classification results obtained utilizing the CLIP model described in section 4.3.4 using Mel – Spectrograms, are showcased in tables 4.9 and 4.10. Table 4.9 presents the results acquired from testing the 8 text – label pairs, shown in table 4.3, on the validation set. From these 8 sets, the two with the highest AUROC values were selected to be tested on the test set and the results are depicted on table 4.10.

Table 4.9: CLIP performance metrics of the text-label pairs on the validation data

Pair index	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC
0	73.32%	5.59%	98.39%	51.99%	51.99%
1	63.76%	22.98%	78.85%	50.92%	50.92%
2	**67.62%**	**31.06%**	**81.15%**	**56.10%**	**56.10%**
3	45.81%	50.31%	44.14%	47.22%	47.22%
4	42.79%	**70.81%**	32.41%	51.61%	51.61%
5	72.99%	0%	100%	50.00%	50.00%
6	72.99%	0%	100%	50.00%	50.00%
7	**71.14%**	**11.18%**	**93.33%**	**52.26%**	**52.26%**

Table 4.9 depicts that the AUROC scores range from 47.22% to 56.10%, with the top two resulting from pairs 2 and 7. It is worth noting that pair 2 is the most descriptive of the 8 text – label pairs utilized on the validation set, while pair 7 is the least descriptive.

This indicates that a more thorough phrase engineering, stemming from the relevant literature, could significantly improve the performance of the model. Furthermore, the sensitivity values vary between 0% and 70.81%, meaning that the model can either not predict at all TP samples or can achieve an adequate performance on that particular metric.

Table 4.10: CLIP performance metrics of the text-label pairs on the test set

Pair index	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC
2	63.82%	**18.84%**	75.28%	47.06%	47.06%
7	**75.59%**	**15.94%**	**90.77%**	**53.36%**	**53.36%**

Pairs 2 and 7 were selected due to their performance on the AUROC metric and were examined on the test set. From the results presented on table 4.10 it can be stated that there is a drop in performance of all metrics when compared to table 4.9 for pair 2. Pair 7 achieves a slightly better AUROC score (53.36%) than in table 4.9 (52.26%) and a sensitivity value of 15.94%. While pair 7 results in better metrics in the test set than pair 2, it still performs very poorly in diagnosing Covid – 19 from Mel – spectrograms.

4.4 Concept Drift Adaptation results

The model VGG – 16 trained for 200 epochs is selected, as the best model due to its high AUROC score, to be used for concept drift adaptation. Initially, the model is tested on the entire drift set in order to observe whether a drift is present. Table 4.11 illustrates the aforementioned results. The classification results obtained after retaining the VGG-16 200 epoch with hidden GS on the drift training data and tested on the drift test set, with the methodology described in section 4.4, are showcased in tables 4.12 and 4.13 Table 4.12 presents the results acquired from retraining only the last dense layer of the model and using the 6 regularization methods shown in table 3.4. Table 4.13 depicts the results acquired from retraining the last two dense layers using the 6 regularization methods shown in table 3.4. Tables 4.14 and 4.15 depict the new metric values after the application of classification threshold optimization. The row with Layer Regularization Method (LRM) 0 on all of the aforementioned tables shows the model’s performance if no LRM is applied on the layer(s). The models with the highest AUROC values, along with the model with the highest sensitivity, have been highlighted.

Table 4.11: Performance metrics for the hidden GS VGG - 16 model trained for 200 epochs on the drift set

Base - Model	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC
VGG – 16	37.63%	31.93%	65.31%	48.62%	54.66%

Table 4.11 clearly illustrates the existence of a concept drift. The classification metrics decrease significantly when the model is tested on the drift set, with AUROC score falling to 54.66% (a 31.85% decline when compared to the 80.21% AUROC score on the test set).

Table 4.12: Concept drift adaptation results from retraining the last dense layer of VGG-16 - 200 epoch - hidden GS

LRM	Epoch	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC
0	100	79.31%	84.00%	50.00%	67.00%	75.75%
	200	**82.76%**	**88.00%**	**50.00%**	**69.00%**	**76.25%**
1	100	81.03%	86.00%	50.00%	68.00%	73.50%
	200	79.31%	84.00%	50.00%	67.00%	72.50%
2	100	77.59%	82.00%	50.00%	66.00%	76.00%
	200	79.31%	84.00%	50.00%	67.00%	74.50%
3	100	79.31%	84.00%	50.00%	67.00%	75.00%
	200	84.48%	90.00%	50.00%	70.00%	74.00%
4	**100**	**79.31%**	**84.00%**	**50.00%**	**67.00%**	**76.25%**
	200	79.31%	84.00%	50.00%	67.00%	74.00%
5	**100**	**75.86%**	**82.00%**	**37.50%**	**59.75%**	**76.25%**
	200	77.59%	84.00%	37.50%	60.75%	75.25%
6	100	81.03%	86.00%	50.00%	68.00%	74.75%
	200	84.48%	**92.00%**	37.50%	64.75%	74.50%

Without any layer regularization method applied, retraining only the last dense layer of the model an AUROC score of 75.75% is achieved after 100 epochs of training and a score of 76.25% is achieved after 200 epochs. Sensitivity scores are considerably higher when compared to the previous sections ranging between 82.00% and 92.00%. After applying each one of the 6 layer regularization methods, it is illustrated that only LRM 4 and 5 achieve an AUROC value of 76.25% after 100 epochs of training. No other model manages to achieve or surpass the AUROC score of LRM 0. In addition, the specificity scores change between 37.50% and 50.00%, marking a significant drop when compared to the previous sections.

Table 4.13: Concept drift adaptation results from retraining the last two dense layers of VGG-16 - 200 epoch - hidden GS

LRM	Epoch	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC
0	100	77.59%	84.00%	37.50%	60.75%	74.75%
	200	77.59%	82.00%	50.00%	66.00%	75.50%
1	100	81.03%	**88.00%**	37.50%	62.75%	75.50%
	200	82.76%	88.00%	50.00%	69.00%	74.50%
2	100	74.14%	80.00%	37.50%	58.75%	73.25%
	200	81.03%	86.00%	50.00%	68.00%	74.25%
3	100	81.03%	86.00%	50.00%	68.00%	77.75%
	200	77.59%	82.00%	50.00%	66.00%	76.00%
4	100	81.03%	86.00%	50.00%	68.00%	74.75%
	200	82.76%	88.00%	50.00%	69.00%	73.37%
5	100	77.59%	82.00%	50.00%	66.00%	74.00%
	200	77.59%	82.00%	50.00%	66.00%	74.25%
6	**100**	**79.31%**	**84.00%**	**50.00%**	**67.00%**	**78.50%**
	200	**79.31%**	**84.00%**	**50.00%**	**67.00%**	**78.25%**

Without any layer regularization method applied, retraining the last two dense layers of the model, an AUROC score of 74.75% is achieved after 100 epochs of training and a score of 75.50% is achieved after 200 epochs. Sensitivity scores are once more considerably higher when compared to the previous sections, ranging between 82.00% and 88.00%. After applying each of the 6 layer regularization methods, it is illustrated that LRM 6 achieves an AUROC value of 78.50% after 100 epochs of training and 78.25% after 200 training epochs. This establishes the last layer regularization method with 100 epochs of training as the best performing one. Namely, when compared to LRM 0 with 100 epochs of training, the AUROC score improves by 5.02% and 3.98% when compared to LRM 0 with 200 epochs of training. Balanced accuracy scores vary between 58.75% and 69.00%, with most values being 66% or higher.

Table 4.14: Concept drift adaptation results from retraining the last dense layer of VGG-16 - 200 epoch - hidden GS with new thresholds

LRM	Epoch	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC	Threshold
0	100	84.48%	92.00%	37.50%	64.75%	75.75%	0.035
	200	82.76%	90.00%	37.50%	63.75%	76.25%	0.015
1	100	81.03%	86.00%	50.00%	68.00%	73.50%	0.528
	200	87.93%	**96.00%**	37.50%	66.75%	72.50%	0.004
2	100	81.03%	88.00%	37.50%	62.75%	76.00%	0.030
	200	82.76%	90.00%	37.50%	63.75%	74.50%	0.045
3	100	86.21%	94.00%	37.50%	67.00%	75.00%	0.004
	200	84.48%	92.00%	37.50%	64.75%	74.00%	0.013
4	**100**	**84.48%**	**92.00%**	**37.50%**	**64.75%**	**76.25%**	**0.027**
	200	77.59%	82.00%	50.00%	66.00%	74.00%	0.673
5	**100**	**82.76%**	**90.00%**	**37.50%**	**63.75%**	**76.25%**	**0.133**
	200	82.76%	90.00%	37.50%	63.75%	75.25%	0.138
6	100	81.03%	86.00%	50.00%	68.00%	74.75%	0.658
	200	84.48%	92.00%	37.50%	64.75%	74.50%	0.222

Table 4.14 generally shows higher sensitivity at the cost of specificity, due to different threshold settings compared to table 4.12. Both tables indicate the trade-off between sensitivity and specificity (as shown in section 4.2), with threshold adjustments playing

a significant role in the model's performance metrics. In general, sensitivity tends to improve with the optimal thresholds derived from the drift validation set, while specificity often decreases. The balanced accuracy values range between 62.75% and 68.00%

Table 4.15: Concept drift adaptation results from retraining the last two dense layers of VGG-16 - 200 epoch - hidden GS with new thresholds

LRM	Epoch	Accuracy	Sensitivity	Specificity	Balanced Accuracy	AUROC	Threshold
0	100	81.03%	88.00%	37.50%	62.75%	74.75%	0.171
	200	82.76%	90.00%	37.50%	63.75%	75.50%	0.957
1	100	84.48%	92.00%	37.50%	64.75%	75.50%	0.55
	200	86.21%	94.00%	37.50%	65.75%	74.50%	0.004
2	100	75.86%	82.00%	37.50%	59.75%	73.25%	0.476
	200	82.76%	90.00%	37.50%	63.75%	74.25%	0.138
3	100	87.93%	96.00%	37.50%	66.75%	77.75%	0.023
	200	87.93%	96.00%	37.50%	66.75%	76.00%	0.002
4	100	82.76%	90.00%	37.50%	63.75%	74.75%	0.052
	200	82.76%	88.00%	50.00%	69.00%	73.37%	0.255
5	100	74.14%	78.00%	50.00%	64.00%	74.00%	0.788
	200	82.76%	90.00%	37.50%	63.75%	74.25%	0.041
6	**100**	**79.31%**	**84.00%**	**50.00%**	**67.00%**	**78.50%**	**0.751**
	200	**75.86%**	**80.00%**	**50.00%**	**65.00%**	**78.25%**	**0.706**

Table 4.15 illustrates higher sensitivity but slightly lower specificity due to different threshold settings when compared to Table 4.13. The significant improvements in sensitivity, indicate a better detection of positive cases with the application of the new thresholds. LRM 6 shows a slight decrease in sensitivity and balanced accuracy at 200 epochs in Table 4.15 when compared to Table 4.13, while at 100 epochs the metrics are identical. Once more, the result of optimizing the classification threshold generally

improves sensitivity, while decreasing specificity. The balanced accuracy metric varies by model.

5

Conclusion – Feature Work

5.1 Conclusion

The current thesis' aim is: (1) the development of Machine Learning and Deep Learning methods for COVID-19 diagnosis from audio data and (2) the implementation of drift adaptation methods that would maintain the developed model's accuracy, in nonstationary environments, throughout time. To that effort, multiple models were tested using both traditional implementations, namely Random Forests and Multilayer Perceptron, and CNN architectures. In order to handle the problem stemming from the limited data available, the Transfer Learning technique was utilized through the usage of the VGG-16, ResNet-50 and Inception-ResNet-V2 models, which were pertained on the ImageNet dataset. Also, the RF and MLP models were trained with and without the application of SMOTE. In addition, Zero-Shot Learning was employed in this study, using the OpenAI CLIP model, to examine how this learning method would perform in a medical classification task and to compare its results against models specifically trained or fine-tuned for this task.

The data used were cough samples from the Coswara dataset containing healthy and COVID-19 infected individuals. The data transformations performed were both feature extractions from the audio recordings and audio to image transformations. The best model using extracted features as input, without incorporating SMOTE, was the random forest model with a threshold of 0.367, which resulted in 66.47% accuracy, 47.83% sensitivity, 71.22% specificity, 59.52% balanced accuracy and 69.91% AUROC score.

The application of SMOTE on the training data resulted in the random forest model, with a threshold value of 0.387, being the best performing one. The model achieved 65.00% accuracy, 59.42% sensitivity, 66.42% specificity, 62.92% balanced accuracy 69.42% AUROC score. The application of SMOTE and threshold optimization from the validation set generally improved the models' metrics, although with a marginal decrease in AUROC. On the extreme case where the models' sensitivity values were both 5.80%, the two aforementioned techniques achieved a sensitivity value 10 times greater and an 21% increase in the balanced accuracy metric, which leads to a better balance between sensitivity and specificity.

With regards to the CNN models tested using Mel – Spectrograms the best performance was achieved by the architecture using the VGG – 16 as base – model and a hidden Gaussian Noise layer. The results are an accuracy value of 78.53%, sensitivity of 62.32%, specificity of 82.66%, balanced accuracy of 72.49% and AUROC value of 80.21%. The utilization of a Gaussian Noise layer as a hidden layer leads to higher AUROC (13% improvement to CNNs using VGG-16 and Inception-ResNet-V2 as base models.) and sensitivity values when compared to using Gaussian Noise as an input layer. The usage of a classification threshold optimization improved the model's sensitivity score, while marginally decreasing the balanced accuracy score. The resulting scores using a threshold of 0.860 were 74.41% accuracy, sensitivity of 68.12%, specificity of 76.01%, balanced accuracy of 72.07% and AUROC value of 80.21%. Overall, optimizing the decision threshold did not achieve the same overall metric improvement as it did to the MLP and RF models. In the case of CNN models there seems to be a trade-off between sensitivity and specificity, but due to the application domain of our models an improved sensitivity is favored to an increased specificity.

Lastly, zero – shot classification utilizing the CLIP model has the worst performance overall on the test set, with a considerable gap between the best performing CLIP text – label pair and the worst performing fully trained model.

When it comes to concept drift adaptation, retraining the last two dense layers of the model using layer regularization method 6 showed an improvement of up to 5% in AUROC value (when compared to no layer regularization method being used) and required 100 epochs of retraining on the drift training data to achieve that score, instead of the 200 epochs needed for the initial model training. The results achieved were

79.31% accuracy, sensitivity of 84.00%, specificity of 50.00%, balanced accuracy of 67.00% and AUROC value of 78.50%. The use of an optimized decision threshold model did not alter the values of that particular model but overall, a trade-off between sensitivity and specificity was showcased, where sensitivity increased and specificity decreased. The balanced accuracy's behavior varied depending on the LRM used. Furthermore, it can be observed that the lower the value of the parameter c of the layer regularization method, the better the model is performing. This could be attributed to the fact that due to the change in concept, a new balance needs to be found between the learned knowledge that should be maintained, and the new knowledge that must be acquired in order for the model to perform the classification task accurately. Because of the fewer data available in the drift set, and the significant change in class imbalance, lower values of the regularization value 'c' (0.01) lead the retrained model to rely more on the new data, whereas higher 'c' values (1.0) lead the model to rely more on the already seen data.

From the aforementioned results, it is evident that the CNN models utilizing transfer learning can be effectively employed in the domain of Covid – 19 detection. They outperform the Random Forest and MLP models and despite the limited and imbalanced training data, they achieve AUROC scores north of 75%, even reaching up to 80.21%. If these models were to be made publicly accessible, they could significantly increase the speed of testing and decrease the pressure pout upon hospitals, health organizations and testing centers.

5.2 Future Work

Future research may include the combination of cough, voice and speech recordings that are provided in the Coswara dataset. The use of different sources of respiratory sounds could improve the classifier's performance since the CNN would have a greater variety of possible features to extract. An effective application of this method could produce a model with better discrimination abilities. Furthermore, the prospect of multimodal classification that combines Mel – spectrograms with the text data provided by the users (gender, age, symptoms, etc.) could potentially increase the model's performance, while also utilizing already provided patient information that remained unused in the current study. Additionally, employing ensemble methods, such as model aggregation or stacking, could aggregate the strengths of multiple classifiers, enhancing overall performance and diagnostic accuracy. Moreover, conducting multiple trainings on different COVID-19 cough datasets, including resources like COUGHVID and Sarcos, could provide richer training data, enhancing the model's ability to generalize across diverse populations and conditions.

Furthermore, worth investigating are also few-shot learning techniques, which enable models to generalize from a limited number of examples, potentially benefiting scenarios with scarce labelled data. Combining few-shot learning with accurate phrase engineering stemming from the relevant literature could further enhance the technique's potential. Another area that could provide significant research potential is federated learning. Allowing models to be trained across decentralized data sources without the need for a centralized data aggregation could enable the preservation of data privacy, enhance data security, and allow model training from data across different populations.

Finally, concept drift adaptation combined with detection methods could ensure model robustness over time, enabling the system to detect and adapt to evolving data distributions with minimal external interaction. Leveraging and combining such techniques could pave the way for more effective COVID-19 diagnostic systems, thus ultimately improving patient outcomes and alleviating pressure from healthcare organizations.

6

Bibliography

[1] N. Bostanghadiri, P. Ziaeefar, M. G. Mofrad, P. Yousefzadeh, A. Hashemi, and D. Darban-Sarokhalil, ‘COVID-19: An Overview of SARS-CoV-2 Variants-The Current Vaccines and Drug Development’, *BioMed Res. Int.*, vol. 2023, p. 1879554, 2023, doi: 10.1155/2023/1879554.

[2] ‘Coronavirus disease (COVID-19): How is it transmitted?’ Accessed: Jan. 19, 2024. [Online]. Available: https://www.who.int/news-room/questions-and-answers/item/coronavirus-disease-covid-19-how-is-it-transmitted

[3] CDC, ‘COVID-19 and Your Health’, Centers for Disease Control and Prevention. Accessed: Jan. 19, 2024. [Online]. Available: https://www.cdc.gov/coronavirus/2019-ncov/prevent-getting-sick/prevention.html

[4] A. Talukder, S. R. Razu, S. M. Alif, M. A. Rahman, and S. M. S. Islam, ‘Association Between Symptoms and Severity of Disease in Hospitalised Novel Coronavirus (COVID-19) Patients: A Systematic Review and Meta-Analysis’, *J. Multidiscip. Healthc.*, vol. 15, pp. 1101–1110, May 2022, doi: 10.2147/JMDH.S357867.

[5] Y. Shi *et al.*, ‘An overview of COVID-19’, *J. Zhejiang Univ. Sci. B*, vol. 21, no. 5, pp. 343–360, May 2020, doi: 10.1631/jzus.B2000083.

[6] J. McNulty, R. B. Reilly, T. E. Taylor, S. M. O’Dwyer, R. W. Costello, and Y. Zigel, ‘Automatic Audio-Based Classification of Patient Inhaler Use: A Pharmacy Based Study’, in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Berlin, Germany: IEEE, Jul. 2019, pp. 2606–2609. doi: 10.1109/EMBC.2019.8857132.

[7] I. Mazić, M. Bonković, and B. Džaja, ‘Two-level coarse-to-fine classification algorithm for asthma wheezing recognition in children’s respiratory sounds’, *Biomed. Signal Process. Control*, vol. 21, pp. 105–118, Aug. 2015, doi: 10.1016/j.bspc.2015.05.002.

[8] J. G. Aleixandre, M. Elgendi, and C. Menon, ‘The Use of Audio Signals for Detecting COVID-19: A Systematic Review’, *Sensors*, vol. 22, no. 21, p. 8114, Oct. 2022, doi: 10.3390/s22218114.

[9] A. Shimoda, Y. Li, H. Hayashi, and N. Kondo, ‘Dementia risks identified by vocal features via telephone conversations: A novel machine learning prediction model’, *PloS One*, vol. 16, no. 7, p. e0253988, 2021, doi: 10.1371/journal.pone.0253988.

[10] J. Korpáš, J. Sadloňová, and M. Vrabec, ‘Analysis of the Cough Sound: an Overview’, *Pulm. Pharmacol.*, vol. 9, no. 5, pp. 261–268, Oct. 1996, doi: 10.1006/pulp.1996.0034.

[11] E. S. Adamidi, K. Mitsis, and K. S. Nikita, ‘Artificial intelligence in clinical care amidst COVID-19 pandemic: A systematic review’, *Comput. Struct. Biotechnol. J.*, vol. 19, pp. 2833–2850, 2021, doi: 10.1016/j.csbj.2021.05.010.

[12] K. Zarkogianni *et al.*, ‘The smarty4covid dataset and knowledge base as a framework for interpretable physiological audio data analysis’, *Sci. Data*, vol. 10, no. 1, p. 770, Nov. 2023, doi: 10.1038/s41597-023-02646-6.

[13] M. Pahar, M. Klopper, R. Warren, and T. Niesler, ‘COVID-19 detection in cough, breath and speech using deep transfer learning and bottleneck features’, *Comput. Biol. Med.*, vol. 141, p. 105153, Feb. 2022, doi: 10.1016/j.compbiomed.2021.105153.

[14] ‘COVID-19 Artificial Intelligence Diagnosis Using Only Cough Recordings’, *IEEE Open J. Eng. Med. Biol.*, vol. 1, pp. 275–281, Sep. 2020, doi: 10.1109/OJEMB.2020.3026928.

[15] A. Imran *et al.*, ‘AI4COVID-19: AI enabled preliminary diagnosis for COVID-19 from cough samples via an app’, *Inform. Med. Unlocked*, vol. 20, p. 100378, Jan. 2020, doi: 10.1016/j.imu.2020.100378.

[16] M. Pahar, M. Klopper, R. Warren, and T. Niesler, ‘COVID-19 cough classification using machine learning and global smartphone recordings’, *Comput. Biol. Med.*, vol. 135, p. 104572, Aug. 2021, doi: 10.1016/j.compbiomed.2021.104572.

[17] C. Duckworth *et al.*, ‘Using explainable machine learning to characterise data drift and detect emergent health risks for emergency department admissions during COVID-19’, *Sci. Rep.*, vol. 11, no. 1, Art. no. 1, Nov. 2021, doi: 10.1038/s41598-021-02481-y.

[18] S. Disabato and M. Roveri, ‘Learning Convolutional Neural Networks in presence of Concept Drift’, presented at the 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary: IEEE, Jul. 2019, pp. 1–8. doi: 10.1109/IJCNN.2019.8851731.

[19] Ö. Gözüaçık, A. Büyükçakır, H. Bonab, and F. Can, ‘Unsupervised Concept Drift Detection with a Discriminative Classifier’, in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, in CIKM ’19. New York, NY, USA: Association for Computing Machinery, Aug. 2019, pp. 2365–2368. doi: 10.1145/3357384.3358144.

[20] A. Bifet and R. Gavaldà, ‘Learning from Time-Changing Data with Adaptive Windowing’, presented at the Proceedings of the 7th SIAM International Conference on Data Mining, Apr. 2007. doi: 10.1137/1.9781611972771.42.

[21] J. Gama, P. Medas, G. Castillo, and P. Rodrigues, ‘Learning with Drift Detection’, presented at the Intelligent Data Analysis, Sep. 2004, pp. 286–295. doi: 10.1007/978-3-540-28645-5_29.

[22] M. Baena-García, J. Campo-Ávila, R. Fidalgo-Merino, A. Bifet, R. Gavald, and R. Morales-Bueno, ‘Early Drift Detection Method’, Jan. 2006.

[23] Z. Yang, S. Al-Dahidi, P. Baraldi, E. Zio, and L. Montelatici, ‘A Novel Concept Drift Detection Method for Incremental Learning in Nonstationary Environments’, *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 1, pp. 309–320, Jan. 2020, doi: 10.1109/TNNLS.2019.2900956.

[24] N.-Y. Liang, G.-B. Huang, P. Saratchandran, and N. Sundararajan, ‘A fast and accurate online sequential learning algorithm for feedforward networks’, *IEEE Trans. Neural Netw.*, vol. 17, no. 6, pp. 1411–1423, Nov. 2006, doi: 10.1109/TNN.2006.880583.

[25] J. Moor, ‘The Dartmouth College Artificial Intelligence Conference: The Next Fifty Years.’, *AI Mag.*, vol. 27, pp. 87–91, Jan. 2006.

[26] H. Larochelle, D. Erhan, and Y. Bengio, ‘Zero-data Learning of New Tasks’.

[27] M. Rezaei and M. Shahidi, ‘Zero-shot learning and its applications from autonomous vehicles to COVID-19 diagnosis: A review’, *Intell.-Based Med.*, vol. 3, p. 100005, Dec. 2020, doi: 10.1016/j.ibmed.2020.100005.

[28] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata, ‘Zero-Shot Learning -- A Comprehensive Evaluation of the Good, the Bad and the Ugly’. arXiv, Sep. 23, 2020. Accessed: Feb. 06, 2024. [Online]. Available: http://arxiv.org/abs/1707.00600

[29] J. C. Schlimmer and R. H. Granger, ‘Incremental learning from noisy data’, *Mach. Learn.*, vol. 1, no. 3, pp. 317–354, Sep. 1986, doi: 10.1007/BF00116895.

[30] N. Sharma *et al.*, ‘Coswara -- A Database of Breathing, Cough, and Voice Sounds for COVID-19 Diagnosis’, in *Interspeech 2020*, Oct. 2020, pp. 4811–4815. doi: 10.21437/Interspeech.2020-2768.

[31] D. Bhattacharya *et al.*, ‘Coswara: A website application enabling COVID-19 screening by analysing respiratory sound samples and health symptoms’. arXiv, Jun. 09, 2022. Accessed: May 08, 2023. [Online]. Available: http://arxiv.org/abs/2206.05053

[32] D. Bhattacharya *et al.*, ‘Coswara: A respiratory sounds and symptoms dataset for remote screening of SARS-CoV-2 infection’, *Sci. Data*, vol. 10, no. 1, Art. no. 1, Jun. 2023, doi: 10.1038/s41597-023-02266-0.

[33] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, ‘Hyperband: A Novel Bandit-Based Approach to Hyperparameter Optimization’.

[34] 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*, Jun. 2009, pp. 248–255. doi: 10.1109/CVPR.2009.5206848.

[35] J. Cui, F. Li, and Z.-L. Shi, ‘Origin and evolution of pathogenic coronaviruses’, *Nat. Rev. Microbiol.*, vol. 17, no. 3, pp. 181–192, 2019, doi: 10.1038/s41579-018-0118-9.

[36] ‘COVID-19 cases | WHO COVID-19 dashboard’, datadot. Accessed: Jan. 19, 2024. [Online]. Available: https://data.who.int/dashboards/covid19/cases

[37] Y. Panahi *et al.*, ‘An overview on the treatments and prevention against COVID-19’, *Virol. J.*, vol. 20, no. 1, p. 23, Feb. 2023, doi: 10.1186/s12985-023-01973-9.

[38] A. Sharma, I. Ahmad Farouk, and S. K. Lal, ‘COVID-19: A Review on the Novel Coronavirus Disease Evolution, Transmission, Detection, Control and Prevention’, *Viruses*, vol. 13, no. 2, p. 202, Jan. 2021, doi: 10.3390/v13020202.

[39] N. van Doremalen *et al.*, ‘Aerosol and Surface Stability of SARS-CoV-2 as Compared with SARS-CoV-1’, *N. Engl. J. Med.*, vol. 382, no. 16, pp. 1564–1567, Apr. 2020, doi: 10.1056/NEJMc2004973.

[40] ‘Isolation and Precautions for People with COVID-19 | CDC’. Accessed: Jan. 23, 2024. [Online]. Available: https://www.cdc.gov/coronavirus/2019-ncov/your-health/isolation.html

[41] ‘Advice for the public on COVID-19 – World Health Organization’. Accessed: Jan. 23, 2024. [Online]. Available: https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public

[42] S. B. Brosnahan, A. H. Jonkman, M. C. Kugler, J. S. Munger, and D. A. Kaufman, ‘COVID-19 and Respiratory System Disorders: Current Knowledge, Future Clinical and Translational Research Questions’, *Arterioscler. Thromb. Vasc. Biol.*, vol. 40, no. 11, pp. 2586–2597, Nov. 2020, doi: 10.1161/ATVBAHA.120.314515.

[43] S. J. Yong, ‘Long COVID or post-COVID-19 syndrome: putative pathophysiology, risk factors, and treatments’, *Infect. Dis. Lond. Engl.*, vol. 53, no. 10, pp. 737–754, Oct. 2021, doi: 10.1080/23744235.2021.1924397.

[44] A.-L. Beaumont, S. Rozencwajg, N. Peiffer-Smadja, and P. Montravers, ‘COVID-19: Brief overview of therapeutic strategies’, *Anaesth. Crit. Care Pain Med.*, vol. 42, no. 1, p. 101181, Feb. 2023, doi: 10.1016/j.accpm.2022.101181.

[45] F. E *et al.*, ‘Advances in COVID-19 mRNA vaccine development’, *Signal Transduct. Target. Ther.*, vol. 7, no. 1, Mar. 2022, doi: 10.1038/s41392-022-00950-y.

[46] R. Patel, M. Kaki, V. S. Potluri, P. Kahar, and D. Khanna, ‘A comprehensive review of SARS-CoV-2 vaccines: Pfizer, Moderna & Johnson & Johnson’, *Hum. Vaccines Immunother.*, vol. 18, no. 1, p. 2002083, doi: 10.1080/21645515.2021.2002083.

[47] ‘SARS-CoV-2, COVID-19, &amp; vaccine side effects | Oncotarget’. Accessed: Jan. 24, 2024. [Online]. Available: https://www.oncotarget.com/news/pr/sars-cov-2-covid-19-vaccine-side-effects/

[48] Y. M. Arabi, S. N. Myatra, and S. M. Lobo, ‘Surging ICU during COVID-19 pandemic: an overview’, *Curr. Opin. Crit. Care*, vol. 28, no. 6, pp. 638–644, Dec. 2022, doi: 10.1097/MCC.0000000000001001.

[49] Q. Rafique *et al.*, ‘Reviewing methods of deep learning for diagnosing COVID-19, its variants and synergistic medicine combinations’, *Comput. Biol. Med.*, vol. 163, p. 107191, Sep. 2023, doi: 10.1016/j.compbiomed.2023.107191.

[50] A. H. Sfayyih, N. Sulaiman, and A. H. Sabry, ‘A review on lung disease recognition by acoustic signal analysis with deep learning networks’, *J. Big Data*, vol. 10, no. 1, p. 101, 2023, doi: 10.1186/s40537-023-00762-z.

[51] J. Knocikova, J. Korpas, M. Vrabec, and M. Javorka, ‘Wavelet analysis of voluntary cough sound in patients with respiratory diseases’, *J. Physiol. Pharmacol. Off. J. Pol. Physiol. Soc.*, vol. 59 Suppl 6, pp. 331–340, Dec. 2008.

[52] K. F. Chung and I. D. Pavord, ‘Prevalence, pathogenesis, and causes of chronic cough’, *Lancet Lond. Engl.*, vol. 371, no. 9621, pp. 1364–1374, Apr. 2008, doi: 10.1016/S0140-6736(08)60595-4.

[53] M. Athanasiou, K. Zarkogianni, K. Karytsas, and K. S. Nikita, ‘An LSTM-based Approach Towards Automated Meal Detection from Continuous Glucose Monitoring in Type 1 Diabetes Mellitus’, in *2021 IEEE 21st International Conference on Bioinformatics and Bioengineering (BIBE)*, Oct. 2021, pp. 1–5. doi: 10.1109/BIBE52308.2021.9635246.

[54] M. Athanasiou, G. Fragkozidis, K. Zarkogianni, and K. S. Nikita, ‘Long Short-term Memory–Based Prediction of the Spread of Influenza-Like Illness Leveraging Surveillance, Weather, and Twitter Data: Model Development and Validation’, *J. Med. Internet Res.*, vol. 25, no. 1, p. e42519, Feb. 2023, doi: 10.2196/42519.

[55] W. Han, C.-F. Chan, C.-S. Choy, and K.-P. Pun, ‘An efficient MFCC extraction method in speech recognition’, in *2006 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2006, p. 4 pp.-. doi: 10.1109/ISCAS.2006.1692543.

[56] R. Y. Choi, A. S. Coyner, J. Kalpathy-Cramer, M. F. Chiang, and J. P. Campbell, ‘Introduction to Machine Learning, Neural Networks, and Deep Learning’, *Transl. Vis. Sci. Technol.*, vol. 9, no. 2, p. 14, doi: 10.1167/tvst.9.2.14.

[57] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. in Springer Series in Statistics. New York, NY: Springer, 2009. doi: 10.1007/978-0-387-84858-7.

[58] S. Hochreiter, ‘The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions’, *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, vol. 6, pp. 107–116, Apr. 1998, doi: 10.1142/S0218488598000094.

[59] ‘Deep Learning’. Accessed: Feb. 09, 2024. [Online]. Available: https://www.deeplearningbook.org/

[60] A. Graves, ‘Practical Variational Inference for Neural Networks’, in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2011. Accessed: Feb. 09, 2024. [Online]. Available: https://papers.nips.cc/paper_files/paper/2011/hash/7eb3c8be3d411e8ebfab08eba5f49632-Abstract.html

[61] A. Graves, ‘Generating Sequences With Recurrent Neural Networks’. arXiv, Jun. 05, 2014. Accessed: Feb. 09, 2024. [Online]. Available: http://arxiv.org/abs/1308.0850

[62] S. J. Pan and Q. Yang, ‘A Survey on Transfer Learning’, *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010, doi: 10.1109/TKDE.2009.191.

[63] H. E. Kim, A. Cosa-Linan, N. Santhanam, M. Jannesari, M. E. Maros, and T. Ganslandt, ‘Transfer learning for medical image classification: a literature review’, *BMC Med. Imaging*, vol. 22, no. 1, p. 69, Apr. 2022, doi: 10.1186/s12880-022-00793-7.

[64] M. Toseef *et al.*, ‘Deep transfer learning for clinical decision-making based on high-throughput data: comprehensive survey with benchmark results’, *Brief. Bioinform.*, vol. 24, no. 4, p. bbad254, Jul. 2023, doi: 10.1093/bib/bbad254.

[65] A. Radford *et al.*, ‘Learning Transferable Visual Models From Natural Language Supervision’. arXiv, Feb. 26, 2021. Accessed: Feb. 06, 2024. [Online]. Available: http://arxiv.org/abs/2103.00020

[66] T. Uchida and K. Yoshida, ‘Concept Drift in Japanese COVID-19 Infection Data’, *Procedia Comput. Sci.*, vol. 207, pp. 380–387, 2022, doi: 10.1016/j.procs.2022.09.072.

[67] I. Žliobaitė, M. Pechenizkiy, and J. Gama, ‘An Overview of Concept Drift Applications’, in *Big Data Analysis: New Algorithms for a New Society*, N. Japkowicz and J. Stefanowski, Eds., in Studies in Big Data. , Cham: Springer International Publishing, 2016, pp. 91–114. doi: 10.1007/978-3-319-26989-4_4.

[68] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang, ‘Learning under Concept Drift: A Review’, *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 12, pp. 2346–2363, Sep. 2019, doi: 10.1109/TKDE.2018.2876857.

[69] A. R. M. S., N. C. R., S. B. R., H. Lahza, and H. F. M. Lahza, ‘A survey on detecting healthcare concept drift in AI/ML models from a finance perspective’, *Front. Artif. Intell.*, vol. 5, 2023, Accessed: Feb. 09, 2024. [Online]. Available: https://www.frontiersin.org/articles/10.3389/frai.2022.955314

[70] N. Oza and S. Russell, ‘Experimental Comparisons of Online and Batch Versions of Bagging and Boosting’, *Proc. Seventh ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, Oct. 2001, doi: 10.1145/502512.502565.

[71] A. Bifet, G. Holmes, and B. Pfahringer, ‘Leveraging Bagging for Evolving Data Streams’, in *Machine Learning and Knowledge Discovery in Databases*, vol. 6321, J. L. Balcázar, F. Bonchi, A. Gionis, and M. Sebag, Eds., in Lecture Notes in Computer Science, vol. 6321. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 135–150. doi: 10.1007/978-3-642-15880-3_15.

[72] H. M. Gomes *et al.*, ‘Adaptive random forests for evolving data stream classification’, *Mach. Learn.*, vol. 106, no. 9, pp. 1469–1495, Oct. 2017, doi: 10.1007/s10994-017-5642-8.

[73] P. Domingos and G. Hulten, ‘Mining high-speed data streams’, in *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, Boston Massachusetts USA: ACM, Aug. 2000, pp. 71–80. doi: 10.1145/347090.347107.

[74] J. Gama, R. Rocha, and P. Medas, ‘Accurate decision trees for mining high-speed data streams’, in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, in KDD ’03. New York, NY, USA: Association for Computing Machinery, Aug. 2003, pp. 523–528. doi: 10.1145/956750.956813.

[75] ‘Decision Trees for Mining Data Streams Based on the Gaussian Approximation | IEEE Journals & Magazine | IEEE Xplore’. Accessed: Feb. 11, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/6466324

[76] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, ‘Optuna: A Next-generation Hyperparameter Optimization Framework’, in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Anchorage AK USA: ACM, Jul. 2019, pp. 2623–2631. doi: 10.1145/3292500.3330701.

[77] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, ‘Algorithms for Hyper-Parameter Optimization’, in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2011. Accessed: Dec. 28, 2023. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2011/hash/86e8f7ab32cfd12577bc2619bc635690-Abstract.html