



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ

Τμήμα Μηχανικών
Βιομηχανικής Σχεδίασης & Παραγωγής

Ενσωματωμένη Μηχανική Μάθηση

Διπλωματική Εργασία

από:

Αλιφιέρη Μαρία

Επιβλέπων Καθηγητής:

Νικολάου Γρηγόρης

Αθήνα, Σεπτέμβριος

2023

Εξεταστική επιτροπή

Όνοματεπώνυμο	Βαθμίδα	Ψηφιακή Υπογραφή
Νικολάου Γρηγόρης	Λέκτορας	
Βασιλειάδου Σουλτάνα	Επίκουρη Καθηγήτρια	
Δρόσος Χρήστος	ΕΔΙΠ	

Δήλωση Συγγραφέα Διπλωματικής Εργασίας

Η κάτωθι υπογεγραμμένη Αλιφιέρη Μαρία του Ευαγγέλου, με αριθμό μητρώου 18389162, φοιτήτρια του Πανεπιστημίου Δυτικής Αττικής της Σχολής Μηχανικών του Τμήματος Μηχανικών Βιομηχανικής σχεδίασης και παραγωγής, δηλώνω υπεύθυνα ότι:

«Είμαι συγγραφέας αυτής της διπλωματικής εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, οι όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε ακριβώς είτε παραφρασμένες, αναφέρονται στο σύνολό τους, με πλήρη αναφορά στους συγγραφείς, τον εκδοτικό οίκο ή το περιοδικό, συμπεριλαμβανομένων και των πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο.

Επίσης, βεβαιώνω ότι αυτή η εργασία έχει συγγραφεί από μένα αποκλειστικά και αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο δικής μου, όσο και του Ιδρύματος. Παράβαση της ανωτέρω ακαδημαϊκής μου ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του πτυχίου μου».

Η δηλούσα:



Αλιφιέρη Μαρία

Περίληψη

Η παρούσα διπλωματική εργασία με τίτλο "Ενσωματωμένη μηχανική μάθηση" ερευνά τα πεδία της τεχνητής νοημοσύνης (TN) και της εφαρμογής της σε ενσωματωμένα συστήματα. Η διπλωματική εργασία εκτυλίσσεται σε τέσσερα κεφάλαια, καθένα από τα οποία συμβάλλει στη βαθύτερη κατανόηση της χρήσης της TN σε ενσωματωμένες συσκευές.

Το πρώτο κεφάλαιο χρησιμεύει ως βάση για ολόκληρη τη διατριβή. Εμβαθύνει στις βασικές έννοιες της TN και διευκρινίζει τη σημασία της ενσωματωμένης μηχανικής μάθησης σε αυτό το πλαίσιο. Το κεφάλαιο αυτό παρέχει μια ολοκληρωμένη επισκόπηση της εξέλιξης της TN και του τρόπου με τον οποίο η ενσωματωμένη μηχανική μάθηση αυξάνει τις δυνατότητες των συσκευών.

Το δεύτερο κεφάλαιο περνά σε μια πρακτική εξερεύνηση των εργαλείων και του υλικού που είναι απαραίτητα για την υλοποίηση ενός συστήματος μέσω της ενσωματωμένης μηχανικής μάθησης. Παρουσιάζει το Edge Impulse, μια ευέλικτη πλατφόρμα τεχνητής νοημοσύνης, και την αναπτυξιακή πλακέτα Arduino Nano 33 BLE Sense, μία βασική πλακέτα στο τοπίο του Διαδικτύου των Πραγμάτων (IoT).

Το τρίτο κεφάλαιο παρουσιάζει στον αναγνώστη τη διαδικασία κατασκευής ενός μοντέλου μηχανικής μάθησης, που είναι ικανό να αναγνωρίζει ανθρώπινες φωνές και να διακρίνει τέσσερις συγκεκριμένες λέξεις-κλειδιά. Περιγράφει λεπτομερώς τις μεθοδολογίες που χρησιμοποιήθηκαν για τη συλλογή δεδομένων, την προεπεξεργασία, την επιλογή μοντέλου και την εκπαίδευση.

Τέλος, το τέταρτο και τελευταίο κεφάλαιο καθοδηγεί τους αναγνώστες στη διαδικασία μεταφόρτωσης του προηγουμένου κατασκευασμένου μοντέλου αναγνώρισης φωνής στο Arduino Nano 33 BLE Sense. Επιπλέον, επιδεικνύει τον

τρόπο με τον οποίο το ενσωματωμένο σύστημα επεξεργάζεται τα εισερχόμενα δεδομένα ήχου, αναγνωρίζει λέξεις-κλειδιά και ανταποκρίνεται σε πραγματικό χρόνο στις αλληλεπιδράσεις με το περιβάλλον.

Λέξεις – Κλειδιά

Τεχνητή Νοημοσύνη, Ενσωματωμένη Μηχανική Μάθηση, Edge Impulse, Arduino, Αισθητήρες, Φωνητικές Εντολές

Abstract

This thesis entitled "Embedded machine learning" investigates the fields of artificial intelligence (AI) and its application in embedded systems. The thesis unfolds in four chapters, each of which contributes to a deeper understanding of the use of AI in embedded devices.

The first chapter serves as the basis for the entire thesis. It delves into the basic concepts of AI and clarifies the importance of embedded machine learning in this context. This chapter provides a comprehensive overview of the evolution of AI and how embedded machine learning increases the capabilities of devices.

The second chapter moves into a practical exploration of the tools and hardware necessary to implement a system through embedded machine learning. It introduces Edge Impulse, a flexible AI platform, and the Arduino Nano 33 BLE Sense development board, a key board in the Internet of Things (IoT) landscape.

The third chapter presents the reader with the process of building a machine learning model that is capable of recognizing human voices and distinguishing four specific keywords. It details the methodologies used for data collection, pre-processing, model selection and training.

Finally, the fourth and final chapter guides readers through the process of uploading the previously constructed voice recognition model to the Arduino Nano 33 BLE Sense. In addition, it demonstrates how the embedded system processes incoming audio data, recognizes keywords, and responds in real-time to interactions with the environment.

Key – Words

Artificial Intelligence, Embedded Machine Learning, Edge Impulse, Arduino, Sensors,
Voice Commands

Ευχαριστίες

Θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες προς τον επιβλέπον καθηγητή μου, τα μέλη της οικογένειάς μου και τους φίλους μου για την ανεκτίμητη υποστήριξη που μου προσέφεραν κατά τη διάρκεια της διπλωματικής μου εργασίας. Αυτό το ταξίδι ήταν απαιτητικό και πολλές φορές προκλητικό, αλλά χάρη στην πίστη, την εμπιστοσύνη και την αφοσίωσή τους, κατάφερα να το ολοκληρώσω με επιτυχία.

Ο επιβλέπον καθηγητής μου με καθοδήγησε καθ' όλη τη διαδρομή, και χωρίς την εμπειρία και τις γνώσεις του, δεν θα ήμουν σε θέση να προχωρήσω. Η οικογένειά μου πάντα ήταν εκεί για μένα, προσφέροντας όχι μόνο συναισθηματική στήριξη αλλά και την αναγκαία υποστήριξη και ενθάρρυνση. Οι φίλοι μου μοιράστηκαν το βάρος αυτής της πορείας μαζί μου και με υποστήριξαν με τον δικό τους μοναδικό τρόπο.

Περιεχόμενα

Περίληψη	4
Λέξεις – Κλειδιά	5
Abstract	6
Key – Words	7
Ευχαριστίες	8
Πίνακας εικόνων	11
Εισαγωγή.....	13
ΚΕΦΑΛΑΙΟ 1° « ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ ΚΑΙ ΕΝΣΩΜΑΤΩΜΕΝΗ ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ ».....	15
1.1 Ιστορική Αναδρομή.....	15
1.2 Τεχνητή νοημοσύνη.....	16
1.3 Μηχανική μάθηση	18
1.4 Βαθιά Μάθηση	20
1.5 Τεχνητά νευρωνικά δίκτυα (ANN)	21
1.6 Ενσωματωμένη μηχανική μάθηση	23
1.6.1 Πλεονεκτήματα και μειονεκτήματα ενσωματωμένης μηχανικής μάθησης	25
1.7 Μεταφορά μάθησης (Transfer learning)	26
1.7.1 Οφέλη Ενσωματωμένης μηχανικής μάθησης	27
1.8 Υπερπροσαρμογή και υποπροσαρμογή (Overfitting and underfitting)	28
ΚΕΦΑΛΑΙΟ 2° « EDGE IMPULSE ΚΑΙ ΜΙΚΡΟΕΛΕΓΚΤΗΣ ARDUINO »	31
2.1 Πλατφόρμα Edge Impulse.....	31
2.1.1 Δυνατότητες της πλατφόρμας.....	32
2.1.2 Πλεονεκτήματα του Edge Impulse.....	33
2.2 Μικροελεγκτές.....	35
2.3 Arduino	36
2.3.1 Πλεονεκτήματα Arduino	37
2.3.2 Λειτουργία Πλακέτας.....	38
2.3.3 Βασικά μέρη περιβάλλοντος ανάπτυξης Arduino	40
2.4 Arduino Nano 33 Ble Sense.....	42

2.4.1 Τεχνικά χαρακτηριστικά	43
ΚΕΦΑΛΑΙΟ 3° « ΑΝΑΠΤΥΞΗ ΜΟΝΤΕΛΟΥ ΑΝΑΓΝΩΡΙΣΗΣ ΕΝΤΟΛΩΝ »	46
3.1 Επιλογή Συσκευής και Φωνητικών Εντολών	47
3.2 Συλλογή δεδομένων	48
3.3 Επεξεργασία δεδομένων	52
3.3.1 Επιλογή χαρακτηριστικών	55
3.4 Εκπαίδευση Μοντέλου	58
3.5 Έλεγχος απόδοσης εκπαίδευσης	63
3.6 Ανάπτυξη βιβλιοθήκης μοντέλου	67
Κεφαλαίο 4° «ΑΝΑΠΤΗΞΗ ΣΥΣΤΗΜΑΤΟΣ».....	69
4.1 Δημιουργία κώδικα μέσω Arduino IDE.....	69
4.2 Υλικό συσκευής – Hardware	77
Επίλογος.....	81
Βιβλιογραφία	82

Πίνακας εικόνων

Εικόνα 1: Υποκατηγορίες Τεχνητή Νοημοσύνης	17
Εικόνα 2 : Στάδια μηχανικής μάθησης	18
Εικόνα 3:κατηγορίες μηχανικής μάθησης	20
Εικόνα 4:Αρχιτεκτονική Νευρωνικού Δικτύου	22
Εικόνα 5:Συνδεση Μηχανικής μάθηση-Ενσωματωμένων Συστημάτων	24
Εικόνα 6:transfer learning	28
Εικόνα 7: Παραδείγματα υπερπροσαρμογής - υποπροσαρμογής.....	30
Εικόνα 8: Πλατφόρμα Edge impulse.....	34
Εικόνα 9: Data explorer	35
Εικόνα 10: Δαίφορα είδη Arduino	37
Εικόνα 11: Arduino IDE	41
Εικόνα 12:Arduino Nano 33 BLE Sense - Hardware	42
Εικόνα 13:Τεχνικά Χαρακτηριστικά Πλακέτας	44
Εικόνα 14:Ακίδες (Pins) Arduino Nano 33 BLE Sense	45
Εικόνα 15:Αλυσίδα machine learning.....	46
Εικόνα 16: Μετατροπή ομιλίας σε σήμα	48
Εικόνα 17: Σύνδεση Arduino με τον υπολογιστή	49
Εικόνα 18: Προετοιμασία συλλογής δεδομένων.....	50
Εικόνα 19: Σύνολο δεδομένων	51
Εικόνα 20:Πρώτη εικόνα ενός δείγματος	52
Εικόνα 21:Εικόνα δείγματος πριν περικοπεί	53
Εικόνα 22: Εικόνα ενός έτοιμου δείγματος.....	53
Εικόνα 23: Ολοκλήρωση συλλογής δεδομένων	54
Εικόνα 24: Καρτέλες ρύθμισης χαρακτηριστικών	57
Εικόνα 25: Χωρικά ταξινομημένα δεδομένα.....	58
Εικόνα 26:Ρυθμίσεις νευρωνικού.....	59
Εικόνα 27: Αρχιτεκτονική μοντέλου	63

Εικόνα 28: Εικόνα διαδικασίας εκπαίδευσης.....	64
Εικόνα 29: Απόδοση εκπαίδευσης	65
Εικόνα 30: Χωρικός πίνακας κατηγοριοποιημένων δεδομένων	66
Εικόνα 31: Χαρακτηριστικά μοντέλου στη συσκευή	66
Εικόνα 32:Επιλογή τύπου βιβλιοθήκης	68
Εικόνα 33: Λογισμικό συσκευής	76
Εικόνα 34:Arduino Nano BLE Sense Rev 2	77
Εικόνα 35:Δίοδος εκπομπής φωτός LED.....	77
Εικόνα 36: Αντίσταση.....	78
Εικόνα 37: Οθόνη LCD μπροστά όψη	79
Εικόνα 38: Οθόνη LCD πίσω όψη.....	79
Εικόνα 39: Ολοκληρωμένο σύστημα αναγνώρισης εντολών.....	80

Εισαγωγή

Η ενσωματωμένη μηχανική μάθηση αποτελεί έναν σημαντικό και αναπτυσσόμενο τομέα στον χώρο της τεχνητής νοημοσύνης. Η χρησιμότητά της εκδηλώνεται σε πολλά πεδία εφαρμογών και έχει αποκτήσει ιδιαίτερη σημασία στην ανάπτυξη προηγμένων ενσωματωμένων συστημάτων. Η παρούσα διπλωματική εργασία επικεντρώνεται στην αξιοποίηση της ενσωματωμένης μηχανικής μάθησης για τη δημιουργία ενός μοντέλου αναγνώρισης φωνητικών εντολών.

Η αναγνώριση φωνής αποτελεί μια τεχνολογία που έχει κατακτήσει μια ευρεία διάδοση σε διάφορους τομείς της τεχνολογίας και της επικοινωνίας. Η ικανότητα ενός συστήματος να αναγνωρίζει και να επεξεργάζεται την ανθρώπινη φωνή έχει φέρει σημαντικές εξελίξεις και πλεονεκτήματα στην αλληλεπίδραση ανθρώπου-μηχανής.

Υπάρχουν αρκετοί λόγοι για την ευρεία διάδοση της χρήσης συσκευών αναγνώρισης φωνής. Πρώτον, η φωνητική αλληλεπίδραση προσφέρει μια φυσική, άνετη και ευχάριστη εμπειρία για τους χρήστες. Αντί να αλληλοεπιδρούν με τις συσκευές μέσω πληκτρολογίου ή οθόνης αφής, οι χρήστες μπορούν απλά να εκφράσουν τις εντολές ή τα αιτήματά τους με τη φωνή τους. Αυτό καθιστά τη χρήση συσκευών πιο ευχάριστη, εύκολη και βολική, ειδικά σε καταστάσεις κατά τις οποίες οι χρήστες έχουν περιορισμένη πρόσβαση σε άλλες μεθόδους εισόδου, όπως κατά την οδήγηση ή την καταγραφή πληροφοριών κατά τη διάρκεια εργασιών.

Επιπλέον, η αναγνώριση φωνής έχει επεκταθεί σε πολλές εφαρμογές και τομείς. Από τις εικονικές προσωπικές βοηθούς σε έξυπνα τηλέφωνα και ηχοσυστήματα στο σπίτι, μέχρι τη χρήση σε ιατρικούς εξοπλισμούς και αυτοκίνητα, η αναγνώριση φωνής

προσφέρει νέες δυνατότητες και βελτιώνει την ευκολία, την αποδοτικότητα και την ασφάλεια των συστημάτων.

Παρά την ευρεία χρήση συσκευών αναγνώρισης φωνητικών εντολών, συχνά αντιμετωπίζονται ορισμένα προβλήματα, όπως η εξάρτηση από τη σύνδεση στο διαδίκτυο, η καθυστέρηση στην απόκριση και η απώλεια ιδιωτικότητας. Αυτοί είναι μερικοί από τους λόγους που ανακύπτει η ανάγκη για δημιουργία ενός μοντέλου αναγνώρισης λέξεων με χρήση μικροελεγκτή. Η χρήση του επιτρέπει την αυτόνομη λειτουργία του συστήματος αναγνώρισης φωνής, χωρίς την ανάγκη για σύνδεση στο διαδίκτυο. Αυτό είναι ιδιαίτερα χρήσιμο σε περιβάλλοντα όπου η σύνδεση στο διαδίκτυο είναι περιορισμένη, ακριβή ή αναξιόπιστη. Έτσι, μπορεί να βοηθήσει στην αντιμετώπιση αυτών των προβλημάτων και να παρέχει μια πιο ολοκληρωμένη και αποτελεσματική εμπειρία χρήστη.

Ως αποτέλεσμα, η δημιουργία ενός μοντέλου αναγνώρισης λέξεων με χρήση μικροελεγκτή, ως ενσωματωμένη μηχανική μάθηση δηλαδή, προσφέρει πολλά πλεονεκτήματα και δυνατότητες, επιτρέποντας την ανάπτυξη ευφυών συστημάτων αλληλεπίδρασης και τη βελτίωση της εμπειρίας των χρηστών.

ΚΕΦΑΛΑΙΟ 1^ο « ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ ΚΑΙ ΕΝΣΩΜΑΤΩΜΕΝΗ ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ »

1.1 Ιστορική Αναδρομή

Η έρευνα στον τομέα της ενσωματωμένης μηχανικής μάθησης έχει αναπτυχθεί σταδιακά κατά τη διάρκεια των τελευταίων δεκαετιών. Ακολουθεί μια συνοπτική ιστορική αναδρομή με κάποιες σημαντικές ημερομηνίες:

1990: Εισαγωγή του όρου "ενσωματωμένη μηχανική μάθηση" (embedded machine learning) από τον Michael L. Tsetlin και την Elaine M. Tsetlin στην έρευνά τους για την εκπαίδευση νευρωνικών δικτύων σε ενσωματωμένα συστήματα. Έτσι αρχίζουν να εμφανίζονται πρώτες εφαρμογές στον τομέα των ενσωματωμένων συστημάτων. Οι αλγόριθμοι μηχανικής μάθησης περιορίζονται σε απλές τεχνικές.

2000-2010: Με την αύξηση της υπολογιστικής ισχύος των ενσωματωμένων συσκευών, επιτρέπεται η εκτέλεση πιο προηγμένων αλγορίθμων μηχανικής μάθησης. Εμφανίζονται πρώτες εφαρμογές ενσωματωμένης μηχανικής μάθησης σε έξυπνες κάμερες ασφαλείας και αναγνώρισης φωνής.

2010-2015: Η έρευνα επικεντρώνεται στην εξέλιξη αλγορίθμων μηχανικής μάθησης που είναι αποδοτικοί από άποψη ενέργειας και μνήμης, καθώς οι ενσωματωμένες συσκευές έχουν αυστηρούς περιορισμούς σε αυτούς τους τομείς.

2016-2018: Εμφανίζονται οι πρώτοι επεξεργαστές με ενσωματωμένη υποστήριξη για αλγορίθμους μηχανικής μάθησης, όπως οι επεξεργαστές της σειράς Google Tensor Processing Units (TPUs) και οι NVIDIA Jetson. Τα TPUs είναι επεξεργαστικές

μονάδες σχεδιασμένες ειδικά για ενσωματωμένη μηχανική μάθηση. Εκπαιδεύονται με τη χρήση νευρωνικών δικτύων και αποτελούν σημαντικό επίτευγμα στον τομέα της επιτάχυνσης των αλγορίθμων μηχανικής μάθησης σε ενσωματωμένες συσκευές. Αντίστοιχα, οι NVIDIA Jetson είναι μια σειρά ενσωματωμένων πλατφορμών που επιτρέπουν την εκτέλεση προηγμένων αλγορίθμων μηχανικής μάθησης απευθείας στις συσκευές.

2022-σήμερα: Η έρευνα επικεντρώνεται στην ανάπτυξη αλγορίθμων μηχανικής μάθησης που είναι πιο αποδοτικοί και ακριβείς, ενσωματώνονται περισσότερες λειτουργίες μηχανικής μάθησης σε ενσωματωμένες συσκευές και εξελίσσονται προηγμένες τεχνικές όπως η αυτόνομη μάθηση και η μάθηση με ενισχυτές.

1.2 Τεχνητή νοημοσύνη

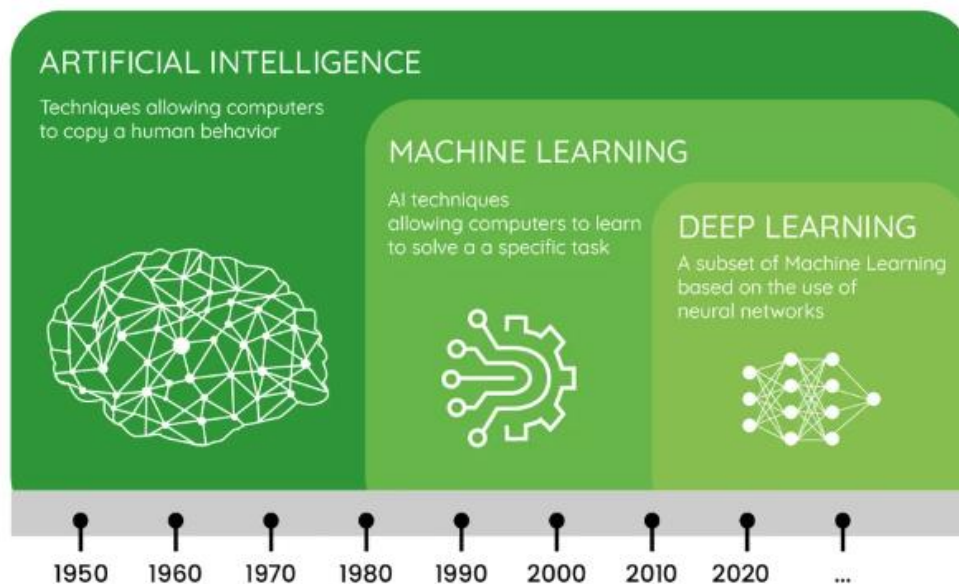
Η Τεχνητή Νοημοσύνη (TN) αναφέρεται στη δημιουργία υπολογιστικών συστημάτων που μπορούν να εκτελούν εργασίες που απαιτούν ανθρώπινη νοημοσύνη. Στην ουσία, η TN αποσκοπεί στην ανάπτυξη υπολογιστικών συστημάτων που μπορούν να αντιλαμβάνονται, να μαθαίνουν, να λαμβάνουν αποφάσεις και να επιλύουν προβλήματα με παρόμοιο τρόπο με τον ανθρώπινο εγκέφαλο.

Η τεχνητή νοημοσύνη χρησιμοποιείται σε πολλούς τομείς και εφαρμογές. Ένας από τους κύριους τομείς εφαρμογής είναι η αυτοματοποίηση και η βελτιστοποίηση διαδικασιών. Μπορεί να εφαρμοστεί για να αυτοματοποιήσει εργασίες που απαιτούν ανθρώπινη παρέμβαση ή είναι χρονοβόρες, όπως η αυτόνομη οδήγηση

αυτοκινήτων, η ρομποτική, οι αυτόματες γραμμές παραγωγής και οι ευφυείς συστοιχίες δικτύων ενέργειας.

Η τεχνητή νοημοσύνη είναι επίσης απαραίτητη για την ανάλυση και την επεξεργασία μεγάλου όγκου δεδομένων. Με τη χρήση αλγορίθμων μηχανικής μάθησης και βαθιών νευρωνικών δικτύων, μπορεί να εξάγει πληροφορίες από δεδομένα και να ανακαλύψει κρυμμένα μοτίβα και συσχετίσεις. Αυτό έχει εφαρμογές σε πολλούς τομείς, όπως η αναγνώριση φωνής και εικόνας, η εξόρυξη γνώσης από μεγάλες βάσεις δεδομένων και η πρόβλεψη τάσεων και συμπεριφορών.

Με τη χρήση της μπορούμε να επιλύουμε πολύπλοκα προβλήματα, να βελτιώσουμε την αποτελεσματικότητα και την απόδοση σε διάφορους τομείς και να δημιουργήσουμε νέες ευκαιρίες. Για αυτούς τους λόγους, η επαναστατική αυτή τεχνολογία είναι τόσο ευρέως διαδεδομένη.

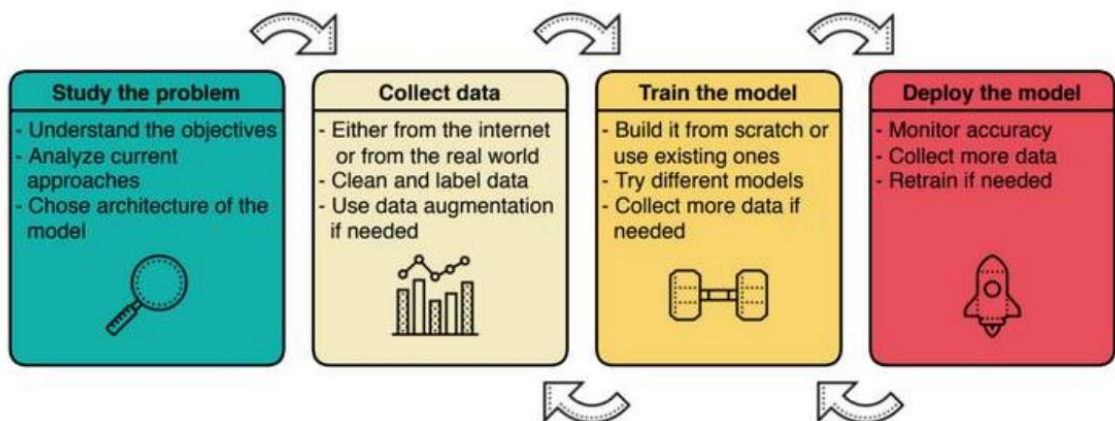


Εικόνα 1: Υποκατηγορίες Τεχνητή Νοημοσύνης

1.3 Μηχανική μάθηση

Η μηχανική μάθηση (machine learning) αναφέρεται σε έναν κλάδο της τεχνητής νοημοσύνης που ασχολείται με την ανάπτυξη αλγορίθμων και μοντέλων που επιτρέπουν στους υπολογιστές να μαθαίνουν από τα δεδομένα και να προβλέπουν, χωρίς να προγραμματίζονται αυστηρά για κάθε πιθανή περίπτωση. Στην ουσία, η μηχανική μάθηση επιτρέπει στους υπολογιστές να "μάθουν" από παραδείγματα και εμπειρίες, και να προβλέπουν και να προσαρμόζουν τη συμπεριφορά τους ανάλογα με τα νέα δεδομένα.

Οι αλγόριθμοι μηχανικής μάθησης εξετάζουν τα δεδομένα, αναγνωρίζουν μοτίβα και κατασκευάζουν μοντέλα που μπορούν να χρησιμοποιηθούν για να προβλέψουν ή να κάνουν αποφάσεις για νέα δεδομένα. Τα δεδομένα που χρησιμοποιούνται μπορεί να είναι δομημένα ή άτακτα, όπως πίνακες, εικόνες, κείμενο, ήχος και άλλα.



Εικόνα 2 : Στάδια μηχανικής μάθησης

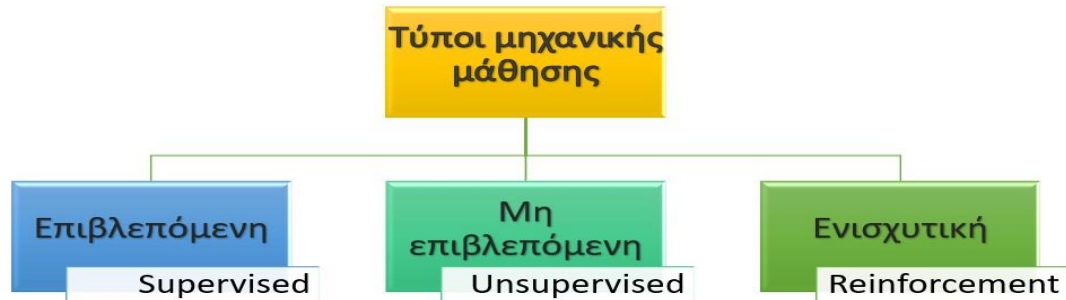
Η διαδικασία που ακολουθείται είναι αρχικά η μελέτη του προβλήματος, ώστε να κατανοηθούν οι στόχοι και να διαπιστωθούν ήδη υπάρχουσες λύσεις και έτσι να

αποφασιστεί η καλύτερη δυνατή προσέγγιση που θα χρησιμοποιηθεί. Δεύτερο βήμα είναι η δημιουργία μια συλλογής δεδομένων (dataset) με στοιχεία είτε από το διαδίκτυο είτε από δειγματοληψία που πραγματοποιείται. Τρίτο βήμα είναι η εκπαίδευση του μοντέλου σε διαφορετικές αρχιτεκτονικές επιλέγοντας αυτή που ανταποκρίνεται καλύτερα στα δεδομένα. Επαναλαμβάνουμε το συγκεκριμένο βήμα αρκετές φορές ώστε να βελτιώσουμε την ακρίβεια του μοντέλου. Τέλος ενσωματώνουμε το μοντέλο στην συσκευή και ελέγχουμε την πραγματική ακρίβεια του μοντέλου.

Οι τεχνικές μηχανικής μάθησης χωρίζονται συνήθως σε τρεις βασικές κατηγορίες:

1. Επιβλεπόμενη μάθηση (supervised learning): Τα μοντέλα εκπαιδεύονται χρησιμοποιώντας δεδομένα με ετικέτα, όπου γνωρίζουμε τη σωστή απάντηση για κάθε παράδειγμα. Στόχος είναι η κατασκευή ενός μοντέλου που μπορεί να προβλέπει σωστά για νέα δεδομένα, χωρίς ετικέτα.
2. Μη επιβλεπόμενη μάθηση (unsupervised learning): Τα μοντέλα εκπαιδεύονται χρησιμοποιώντας δεδομένα χωρίς ετικέτα, χωρίς γνώση της σωστής απάντησης. Στόχος είναι η ανακάλυψη κρυμμένων μοτίβων και δομών στα δεδομένα.
3. Ενισχυτική μάθηση (reinforcement learning): Τα μοντέλα αλληλοεπιδρούν με ένα περιβάλλον και εκπαιδεύονται μέσω δοκιμών και λαθών, με τον στόχο να μάθουν την καλύτερη συμπεριφορά για μια συγκεκριμένη εργασία.

Η μηχανική μάθηση έχει εφαρμογές σε πολλούς τομείς, όπως η αναγνώριση φωνής, η ανίχνευση αντικειμένων, η πρόβλεψη σεισμών, η ιατρική διάγνωση, η ρομποτική και πολλές άλλες εφαρμογές.



Εικόνα 3:κατηγορίες μηχανικής μάθησης

1.4 Βαθιά Μάθηση

Η βαθιά μάθηση είναι μια υποκατηγορία της μηχανικής μάθησης που επικεντρώνεται στην ανάπτυξη αλγορίθμων που μπορούν να μάθουν αναπαραστάσεις υψηλού επιπέδου από πολύπλοκα δεδομένα. Αυτή η τεχνική είναι εμπνευσμένη από τον τρόπο λειτουργίας του ανθρώπινου εγκεφάλου.

Η βαθιά μάθηση χρησιμοποιεί τεχνητά νευρωνικά δίκτυα, γνωστά και ως βαθιά νευρωνικά δίκτυα, για να εκπαιδεύσει μοντέλα που μπορούν να εξάγουν σύνθετες αναπαραστάσεις από τα δεδομένα. Ο όρος "βαθιά" αναφέρεται στο γεγονός τα δίκτυα αποτελούνται από πολλαπλά επίπεδα νευρώνων, γνωστά ως κρυφά επίπεδα, που συνεργάζονται για την επεξεργασία των δεδομένων εισόδου.

Κατά τη διάρκεια της εκπαίδευσης, τα βαθιά νευρωνικά δίκτυα προσαρμόζουν τις παραμέτρους τους ώστε να μπορούν να αναγνωρίσουν χαρακτηριστικά και να κάνουν προβλέψεις στα δεδομένα εκπαίδευσης.

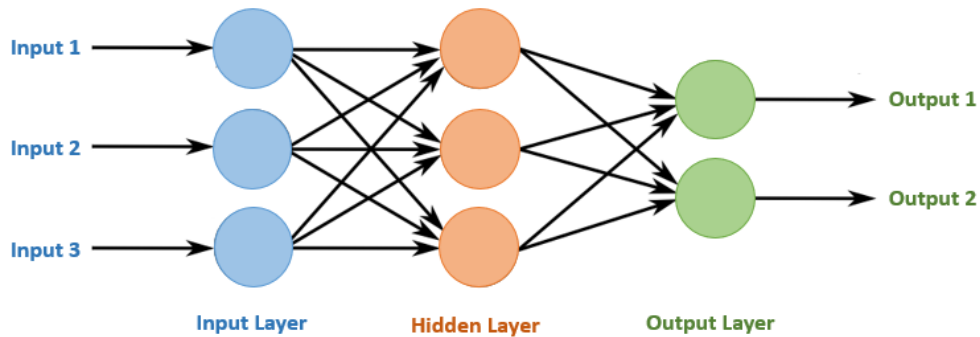
1.5 Τεχνητά νευρωνικά δίκτυα (ANN)

Τα τεχνητά νευρωνικά δίκτυα (artificial neural networks) είναι μια κατηγορία αλγορίθμων μηχανικής μάθησης που εμπνέονται από τη δομή και λειτουργία του ανθρώπινου εγκεφάλου. Τα πρώτα μοντέλα νευρωνικών δικτύων εμφανίστηκαν στο 1943 όπου οι McCulloch και Pitts λειτούργησαν ως τους πρωτεργάτες για την τεχνητή νοημοσύνη. Ωστόσο χρειάστηκε να περάσουν κάποιες δεκαετίες έρευνας και προόδου ώστε να μπορέσουν να εφαρμοστούν σε πρακτικά καθημερινά προβλήματα.

Τα τεχνητά νευρωνικά δίκτυα αποτελούνται από διάφορα επίπεδα νευρώνων, τα οποία συνδέονται μεταξύ τους με βαρύτητες (weights). Κάθε νευρώνας λαμβάνει εισόδους, εφαρμόζει έναν υπολογισμό και παράγει μια έξοδο. Οι επικοινωνίες μεταξύ των νευρώνων προσδίδουν στο δίκτυο την ικανότητα να αναγνωρίζει πρότυπα και να προβλέπει αποτελέσματα.

Το κύριο πλεονέκτημα των τεχνητών νευρωνικών δικτύων είναι η ικανότητά τους να μάθουν από τα δεδομένα και να προσαρμοστούν σε νέες πληροφορίες, καθιστώντας τα ιδιαίτερα ισχυρά στην επίλυση πολύπλοκων προβλημάτων. Τα τεχνητά νευρωνικά δίκτυα έχουν ευρεία εφαρμογή σε πολλούς τομείς. Χρησιμοποιούνται με στόχο την αναγνώριση φωνής, εικόνας και κειμένου, πρόβλεψη και πρόγνωση, αυτόματη

μετάφραση, αυτόνομη οδήγηση οχημάτων, ρομποτική, βιοϊατρική και πολλές άλλες εφαρμογές.



Εικόνα 4: Αρχιτεκτονική Νευρωνικού Δικτύου

Υπάρχουν διάφορα είδη νευρωνικών δικτύων, τα οποία είναι κατάλληλα για διάφορες εφαρμογές και προβλήματα. Ορισμένα από τα κύρια είδη νευρωνικών δικτύων περιλαμβάνουν:

1. Πολυεπίπεδα Πλήρως Συνδεδεμένα Νευρωνικά Δίκτυα (Fully Connected Neural Networks): Είναι ο πιο απλός τύπος νευρωνικού δικτύου, όπου κάθε νευρώνας σε ένα επίπεδο συνδέεται με κάθε νευρώνα στο επόμενο επίπεδο. Χρησιμοποιούνται κυρίως για προβλήματα ταξινόμησης.
2. Συνελικτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks - CNNs): Είναι σχεδιασμένα ειδικά για την επεξεργασία εικόνων και αναγνώριση μοτίβων. Χρησιμοποιούν φίλτρα συνέλιξης για την ανίχνευση χαρακτηριστικών σε διάφορα μέρη της εικόνας.
3. Αναδραστικά Νευρωνικά Δίκτυα (Recurrent Neural Networks - RNNs): Είναι κατάλληλα για την επεξεργασία ακολουθιών δεδομένων, όπως κείμενο, ήχος και

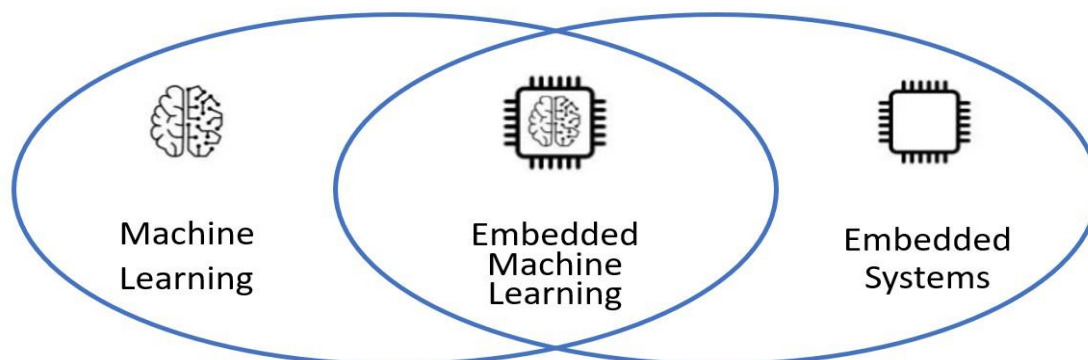
χρονοσειρές. Έχουν ενσωματωμένη μνήμη για την αντιμετώπιση των σχέσεων μεταξύ των προηγούμενων και των τρέχων εισόδων.

4. Αυτονομούμενα Νευρωνικά Δίκτυα (Autoencoder Neural Networks): Χρησιμοποιούνται για τη μείωση της διάστασης των δεδομένων και την εξαγωγή χαρακτηριστικών. Αποτελούνται από έναν κωδικοποιητή και έναν αποκωδικοποιητή.

Κάθε είδος νευρωνικού δικτύου έχει τις δικές του μοναδικές ιδιότητες και εφαρμογές. Η επιλογή του κατάλληλου τύπου εξαρτάται από τα δεδομένα, την εφαρμογή και τους στόχους της μηχανικής μάθησης.

1.6 Ενσωματωμένη μηχανική μάθηση

Η ενσωματωμένη μηχανική μάθηση (embedded machine learning) αναφέρεται στην εφαρμογή τεχνικών μηχανικής μάθησης (machine learning) σε ενσωματωμένα συστήματα και συσκευές. Γνωστή και ως Tiny ML ενσωματώνει μηχανική μάθηση με μικροελεγκτές (microcontrollers) και άλλες συσκευές με μειωμένους πόρους υπολογιστικής ισχύος και ενέργειας. Η Tiny ML προέκυψε από την ανάγκη για εκτέλεση μοντέλων μηχανικής μάθησης απευθείας σε αυτόνομες συσκευές, όπως αισθητήρες, wearables, έξυπνα αντικείμενα και άλλες συσκευές του Internet of Things (IoT). Η λειτουργία ενσωματωμένων συστημάτων μηχανικής μάθησης (ML) σε συσκευές αιχμής έχει γίνει όλο και πιο ελκυστική τα τελευταία χρόνια, καθώς η αποφυγή της χρήσης συστημάτων μέσω υπολογιστικού νέφους (cloud) μπορεί να μειώσει την διαρροή δεδομένων, τον λανθάνοντα χρόνο και να βελτιώσει την ασφάλεια και την προσβασιμότητα.



Εικόνα 5:Συνδεση Μηχανικής μάθησης-Ενσωματωμένων Συστημάτων

Οι μικροελεγκτές έχουν περιορισμένη υπολογιστική ισχύ και μνήμη, και συνήθως λειτουργούν με συγκεκριμένη ενέργεια, από μπαταρίες ή άλλες πηγές. Ωστόσο, η τεχνολογία Tiny ML επιτρέπει την εκτέλεση μικρών και ελαφρών μοντέλων μηχανικής μάθησης σε αυτές τις συσκευές, επιτρέποντας την επεξεργασία δεδομένων και την λήψη αποφάσεων στον τοπικό χώρο, χωρίς την ανάγκη σύνδεσης με το cloud ή το διαδίκτυο.

Η τεχνολογία Tiny ML έχει ευρύ φάσμα εφαρμογών, όπως ανίχνευση και αναγνώριση προτύπων, παρακολούθηση υγείας, επιτήρηση περιβάλλοντος, αυτόνομη πλοήγηση, έξυπνη ενεργειακή διαχείριση και άλλες προηγμένες εφαρμογές. Η χρήση της μηχανικής μάθησης σε τέτοιου είδους συσκευές ανοίγει νέες προοπτικές για την επέκταση των δυνατοτήτων των αυτόνομων συστημάτων και των έξυπνων αντικειμένων.

1.6.1 Πλεονεκτήματα και μειονεκτήματα ενσωματωμένης μηχανικής μάθησης

Η ενσωματωμένη μηχανική μάθηση παρουσιάζει αρκετά πλεονεκτήματα που την καθιστούν ιδιαίτερα αξιόλογη για διάφορες εφαρμογές. Ανάμεσα σε αυτά τα πλεονεκτήματα περιλαμβάνονται:

1. **Αυτοματοποίηση Αποφάσεων:** Η ενσωματωμένη μηχανική μάθηση επιτρέπει στις συσκευές και τα συστήματα να λαμβάνουν αυτόνομες αποφάσεις βασισμένες σε δεδομένα και εμπειρία, απαλλαγμένα από συνεχή ανθρώπινη παρέμβαση.
2. **Συλλογή Δεδομένων:** Η χρήση αισθητήρων μέσω ενσωματωμένης μηχανικής μάθησης μπορεί να βοηθήσει στη συλλογή και την αξιοποίηση δεδομένων από το περιβάλλον.
3. **Συστήματα σε Πραγματικό Χρόνο:** Η ενσωματωμένη μηχανική μάθηση επιτρέπει την άμεση ανταπόκριση των συστημάτων σε διαφορετικές καταστάσεις, επιτρέποντας τη γρήγορη ανταπόκριση.
4. **Απόδοση:** Με την ικανότητά της να εξάγει πρότυπα από τα δεδομένα, μπορεί να βελτιώσει την ακρίβεια και την αποδοτικότητα του συστήματος.
5. **Τιμή:** Η χρήση ενσωματωμένης μηχανικής μάθησης μπορεί να είναι οικονομικά αποδοτική. Οι σύγχρονες τεχνικές επιτρέπουν την υλοποίηση σε οικονομικά προσιτές πλατφόρμες και συστήματα, εξοικονομώντας πόρους.
6. **Μέγεθος:** Οι μικροελεγκτές και οι μικροεπεξεργαστές που χρησιμοποιούνται για ενσωματωμένα συστήματα συχνά έχουν μικρότερο μέγεθος.

Η ενσωματωμένη μηχανική μάθηση προσφέρει πολλά πλεονεκτήματα, όμως όπως και κάθε τεχνολογία, φέρει και ορισμένα μειονεκτήματα που πρέπει να ληφθούν

υπόψιν. Το βασικότερο από αυτά είναι η μνήμη, ο χώρος δηλαδή που η συσκευές διαθέτουν για την ενσωμάτωση των μοντέλων.

1.7 Μεταφορά μάθησης (Transfer learning)

Η μεταφορά μάθησης (transfer learning) είναι μια τεχνική στον τομέα της μηχανικής μάθησης όπου τα μοντέλα που έχουν εκπαιδευτεί σε ένα πρόβλημα μεγάλης κλίμακας αξιοποιούνται και προσαρμόζονται για την επίλυση ενός διαφορετικού, αλλά σχετικού προβλήματος. Σε αντίθεση με την παραδοσιακή μηχανική μάθηση που απαιτεί την εκπαίδευση ενός μοντέλου από την αρχή, η μεταφορά μάθησης εκμεταλλεύεται την πληροφορία που έχει ήδη αποκτηθεί από παρόμοια προβλήματα. Έτσι αναδεικνύεται η ιδέα της αξιοποίησης της γνώσης που έχει αποκτηθεί από ένα μοντέλο σε μια περίπτωση για τη βελτίωση της απόδοσης σε μια άλλη περίπτωση. Αυτό μπορεί να συμβεί με τη μεταφορά των βαρών, των χαρακτηριστικών ή ακόμα και των γενικών εμπειριών που έχουν αποκτηθεί από το πρώτο πρόβλημα στο δεύτερο. Κυρίαρχος σκοπός της μεταφοράς μάθησης είναι η εξαγωγή μεγαλύτερων αποδόσεων και γενίκευσης από τα μοντέλα, ενώ ταυτόχρονα μειώνει τον χρόνο και τους πόρους που απαιτούνται για την εκπαίδευση ενός μοντέλου από το μηδέν.

Η μεταφορά μάθησης μπορεί να επιτευχθεί με διάφορους τρόπους, όπως την εκπαίδευση ενός μοντέλου σε ένα μεγάλο σύνολο δεδομένων και την προσαρμογή του σε ένα μικρότερο σύνολο, τη χρήση προεκπαιδευμένων μοντέλων που έχουν εκπαιδευτεί σε μεγάλα δεδομένα, και την εξαγωγή χαρακτηριστικών από ένα μοντέλο και τη χρήση τους σε ένα άλλο. Έτσι επιτρέπεται η ανάπτυξη πιο

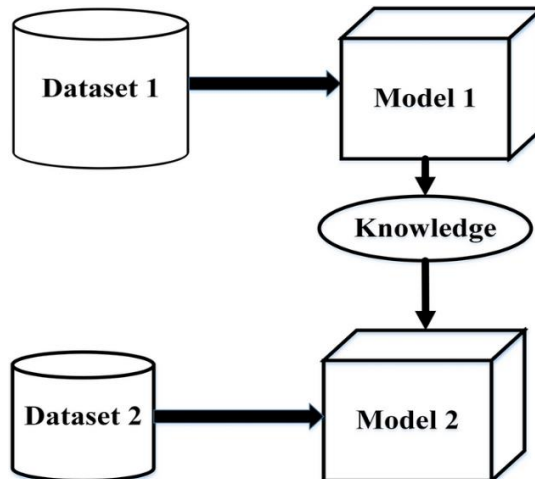
αποτελεσματικών και γενικευμένων μοντέλων μηχανικής μάθησης, που μπορούν να εφαρμοστούν σε ποικίλα προβλήματα με μεγαλύτερη αξιοπιστία και απόδοση.

1.7.1 Οφέλη Ενσωματωμένης μηχανικής μάθησης

Η μεταφορά μάθησης έχει και άλλα σημαντικά οφέλη:

1. **Αποτελεσματικότητα σε περιορισμένα δεδομένα:** Σε περιπτώσεις όπου δεν διαθέτουμε αρκετά δεδομένα για να εκπαιδύσουμε ένα μοντέλο από το μηδέν, η μεταφορά μάθησης μπορεί να βοηθήσει στην ανάπτυξη αποδοτικών μοντέλων με βάση τα δεδομένα που ήδη υπάρχουν.
2. **Γενίκευση γνώσης:** Με την μεταφορά μάθησης, τα μοντέλα μπορούν να αποκτήσουν γενικευμένη γνώση πάνω σε κάποιο θέμα. Αυτό σημαίνει ότι αν ένα μοντέλο έχει εκπαιδευτεί σε μία μεγάλη ποικιλία προβλημάτων, μπορεί να ανταποκριθεί καλύτερα σε νέα, μη γνωστά προβλήματα.
3. **Εξοικονόμηση χρόνου και πόρων:** Η εκπαίδευση ενός μοντέλου από το μηδέν μπορεί να απαιτεί αρκετό χρόνο, υπολογιστικούς πόρους και δεδομένα. Η μεταφορά μάθησης επιτρέπει τη χρήση των υπαρχόντων μοντέλων και δεδομένων, εξοικονομώντας χρόνο και πόρους.
4. **Βελτιωμένη σταθερότητα και αξιοπιστία:** Ένα μοντέλο που έχει εκπαιδευτεί σε μεγάλα και ποικίλα δεδομένα έχει συνήθως βελτιωμένη αντοχή σε διάφορες συνθήκες και ατυχίες, καθιστώντας το πιο αξιόπιστο σε πραγματικές συνθήκες.

5. **Επιτάχυνση εκπαίδευσης:** Αν έχουμε ήδη ένα προεκπαιδευμένο μοντέλο, μπορούμε να χρησιμοποιήσουμε τα βάρη του ως αρχικές τιμές για το νέο μας μοντέλο. Αυτό μπορεί να επιταχύνει την εκπαίδευση και να αυξήσει την πιθανότητα εύρεσης καλύτερων λύσεων.



Εικόνα 6:transfer learning

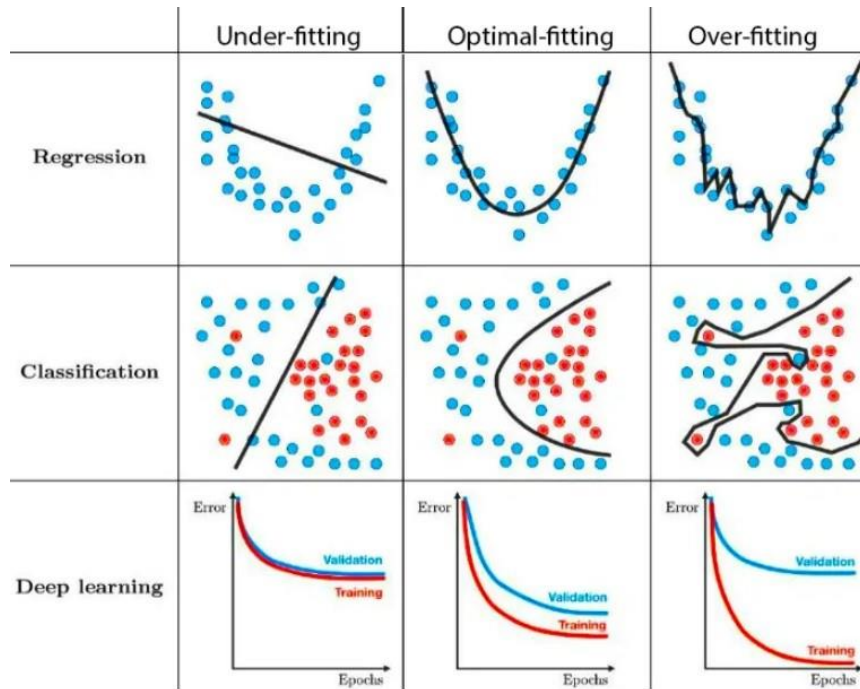
1.8 Υπερπροσαρμογή και υποπροσαρμογή (Overfitting and underfitting)

Η επιβλεπόμενη μάθηση, στο πλαίσιο της μηχανικής μάθησης, αναφέρεται στην εκπαίδευση ενός μοντέλου με τη χρήση δεδομένων εισόδου με ετικέτα και αντίστοιχων ετικετών εξόδου. Ωστόσο, κατά τη διάρκεια της εκπαίδευσης, μπορούν να παρουσιαστούν δύο βασικές κατηγορίες προβλημάτων: το overfitting (υπερπροσαρμογή) και το underfitting (υποπροσαρμογή).

1. Overfitting: Το overfitting συμβαίνει όταν το μοντέλο μάθει να "μνημονεύει" τα δεδομένα εκπαίδευσης αντί να γενικεύει τα πρότυπα. Αποτέλεσμα αυτού είναι ότι το μοντέλο παρουσιάζει πολύ καλή απόδοση στα δεδομένα εκπαίδευσης, αλλά αποτυγχάνει να γενικεύσει σε νέα δεδομένα. Αυτό μπορεί να οδηγήσει σε υπερβολική πολυπλοκότητα του μοντέλου ή στην ύπαρξη πολλών παραμέτρων που είναι υψηλά προσαρμοσμένες στα δεδομένα εκπαίδευσης.

2. Underfitting: Το underfitting συμβαίνει όταν το μοντέλο δεν είναι ικανό να αποκτήσει αρκετή γενικότητα για να μάθει τα πραγματικά πρότυπα των δεδομένων. Αποτέλεσμα αυτού είναι ότι το μοντέλο δεν μπορεί να προσαρμοστεί καλά στα δεδομένα εκπαίδευσης ή να γενικεύσει σε νέα δεδομένα. Αυτό μπορεί να οφείλεται στην ανεπαρκή πολυπλοκότητα του μοντέλου ή στην ανεπαρκή διάρκεια της εκπαίδευσης.

Για να αντιμετωπιστούν αυτά τα προβλήματα, μπορούν να χρησιμοποιηθούν διάφορες τεχνικές. Κάποια παραδείγματα περιλαμβάνουν τη χρήση διαίρεσης των δεδομένων σε σύνολα εκπαίδευσης, επικύρωσης και ελέγχου, τη ρύθμιση της πολυπλοκότητας του μοντέλου, την εφαρμογή τεχνικών απόκλισης (regularization) και τη χρήση συνόλων δεδομένων μεγαλύτερης έκτασης. Είναι σημαντικό να βρεθεί μια ισορροπία μεταξύ του overfitting και του underfitting για να επιτευχθεί η βέλτιστη απόδοση του μοντέλου σε νέα δεδομένα.



Εικόνα 7: Παραδείγματα υπερπροσαρμογής - υποπροσαρμογής

ΚΕΦΑΛΑΙΟ 2^ο « EDGE IMPULSE ΚΑΙ ΜΙΚΡΟΕΛΕΓΚΤΗΣ ARDUINO »

2.1 Πλατφόρμα Edge Impulse

Το Edge Impulse αποτελεί μία προηγμένη πλατφόρμα ανάπτυξης και δοκιμής μοντέλων μηχανικής μάθησης σε ενσωματωμένα συστήματα. Δημιουργήθηκε το 2019 από μια ομάδα προγραμματιστών και ερευνητών με σκοπό να παρέχει ένα εύκολο και αποτελεσματικό εργαλείο για την ανάπτυξη σε ενσωματωμένα συστήματα. Από τότε, η πλατφόρμα έχει εξελιχθεί και εμπλουτιστεί με περισσότερες δυνατότητες και υποστηρίζει μια ποικιλία από μικροελεγκτές και αισθητήρες, καθιστώντας την ένα σημαντικό εργαλείο για την πρόοδο προηγμένων ενσωματωμένων εφαρμογών μηχανικής μάθησης. Αυτή η πλατφόρμα αποτελεί έναν σημαντικό εργαλείο για την ανάπτυξη και τον έλεγχο προηγμένων λειτουργιών ενσωματωμένης μηχανικής μάθησης, παρέχοντας ταυτόχρονα τη δυνατότητα άμεσης εκτέλεσης των μοντέλων σε πραγματικά ενσωματωμένα συστήματα. Μέσα από αυτήν την πλατφόρμα, οι ερευνητές και οι προγραμματιστές μπορούν να εξερευνήσουν τις δυνατότητες της ενσωματωμένης μηχανικής μάθησης, να αναπτύξουν προηγμένα μοντέλα, και να τα δοκιμάσουν σε πραγματικές εφαρμογές που λειτουργούν σε περιβάλλοντα περιορισμένων πόρων. Έτσι, το Edge Impulse προσφέρει έναν ολοκληρωμένο και πρακτικό τρόπο για την υλοποίηση μηχανικής μάθησης σε μικροελεγκτές, ανοίγοντας νέους ορίζοντες στην ανάπτυξη ενσωματωμένων συστημάτων.

2.1.1 Δυνατότητες της πλατφόρμας

Η πλατφόρμα Edge Impulse παρέχει μια σειρά από δυνατότητες που επιτρέπουν την ανάπτυξη και τον έλεγχο μοντέλων μηχανικής μάθησης για ενσωματωμένα συστήματα. Αρχικά, παρέχει εύκολη συλλογή δεδομένων αφού επιτρέπει τη συλλογή δεδομένων από διάφορες πηγές, όπως αισθητήρες, κινητά τηλέφωνα και άλλες συσκευές. Η δημιουργία και η εκπαίδευση μοντέλων γίνεται επίσης ευκολότερη αφού οι χρήστες μπορούν να δημιουργήσουν και να εκπαιδεύσουν μοντέλα μηχανικής μάθησης χρησιμοποιώντας διάφορες αρχιτεκτονικές.

Η πλατφόρμα παρέχει επίσης εργαλεία για τη βελτιστοποίηση των μοντέλων, ώστε να επιτυγχάνονται καλύτερες επιδόσεις. Επιπλέον, οι χρήστες μπορούν να εκτελέσουν προεπεξεργασία δεδομένων, όπως απομάκρυνση θορύβου και ανάλυση χαρακτηριστικών.

Αφού δημιουργηθεί το μοντέλο μηχανικής μάθησης, μπορεί να ενσωματωθεί στη συσκευή. Τα μοντέλα που αναπτύσσονται μπορούν να ενσωματωθούν σε συστήματα και να εκτελούνται σε πραγματικό χρόνο. Αφού το μοντέλο ενσωματωθεί γίνονται οι δοκιμές και η αξιολόγησή του. Οι χρήστες μπορούν να δοκιμάσουν τα μοντέλα τους σε διάφορες συνθήκες και να τα αξιολογήσουν σε πραγματικό περιβάλλον. Αυτές οι δυνατότητες επιτρέπουν στους προγραμματιστές και τους ερευνητές να δημιουργήσουν και να εκπαιδεύσουν μοντέλα μηχανικής μάθησης, να τα ενσωματώσουν σε συστήματα και να τα δοκιμάσουν σε πραγματικά περιβάλλοντα, επιτρέποντας έτσι την ανάπτυξη προηγμένων εφαρμογών που βασίζονται σε ενσωματωμένη μηχανική μάθηση.

2.1.2 Πλεονεκτήματα του Edge Impulse

Αναλυτικότερα μερικά από τα πλεονεκτήματα της πλατφόρμας είναι:

Εύκολη Ανάπτυξη: Το Edge Impulse παρέχει ένα φιλικό περιβάλλον για την ανάπτυξη μοντέλων μηχανικής μάθησης. Έχει γραφική διεπαφή που επιτρέπει τη δημιουργία και την διαμόρφωση πειραμάτων εύκολα και γρήγορα.

Ενσωματωμένη Επεξεργασία Δεδομένων: Το Edge Impulse παρέχει εργαλεία για την επεξεργασία και τον καθαρισμό των δεδομένων που συλλέγονται από αισθητήρες. Εφαρμόζοντας φίλτρα, δίνεται η δυνατότητα να αφαιρείται ο θόρυβος και να διορθώνονται ανωμαλίες.

Δημιουργία και Εκπαίδευση Μοντέλων: Η δημιουργία προσαρμοσμένων μοντέλων μηχανικής μάθησης γίνεται εφικτή χρησιμοποιώντας τα δεδομένα που έχουν συλλεχθεί. Η πλατφόρμα Edge Impulse υποστηρίζει διάφορες αλγοριθμικές προσεγγίσεις, όπως τα νευρωνικά δίκτυα.

Εξαγωγή για Ενσωματωμένα Συστήματα: Μόλις εκπαιδευτεί το μοντέλο μηχανικής μάθησης, το Edge Impulse σας επιτρέπει να το εξαγάγετε σε μορφές που μπορούν να ενσωματωθούν σε μικροελεγκτές. Αυτό σημαίνει ότι μπορείτε να το εκτελέσετε σε πραγματικό χρόνο.

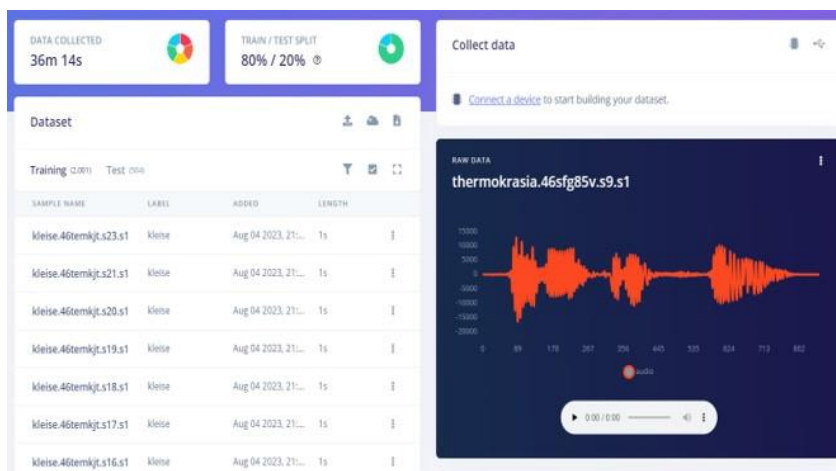
Υποστήριξη Συσκευών και Αισθητήρων: Το Edge Impulse υποστηρίζει μια ποικιλία από αισθητήρες και συσκευές, όπως αισθητήρες κίνησης, περιβάλλοντος, ήχου και

βιομετρικά δεδομένα. Αυτό επιτρέπει τη συλλογή από πολλαπλά είδη δεδομένων για την ανάπτυξη μοντέλου.

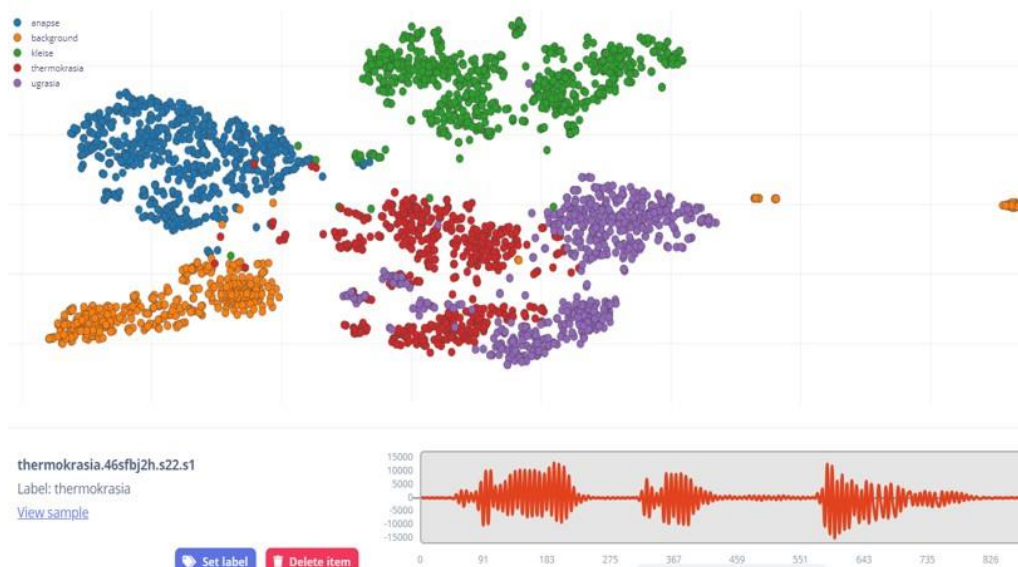
Ανάλυση και Οπτικοποίηση : Η πλατφόρμα επιτρέπει την ανάλυση των δεδομένων που έχουν συλλεχθεί από τους αισθητήρες. Επιτρέποντας την εξέταση διαφόρων πτυχών των δεδομένων, όπως διακυμάνσεις, διακριτικά χαρακτηριστικά, και τάσεις. Η πλατφόρμα παρέχει επίσης διάφορα γραφήματα και οπτικές αναπαραστάσεις για να βοηθήσει στην απεικόνιση των δεδομένων σε μια κατανοητή μορφή.

Ενσωματωμένη Αξιολόγηση: Μετά την ανάπτυξη του μοντέλου, στην πλατφόρμα μπορεί να γίνει αξιολόγηση της απόδοσής του στο περιβάλλον που προορίζεται να λειτουργήσει. Αυτό μπορεί να γίνει μέσω αυτής και τους ενσωματωμένους αλγορίθμους αξιολόγησης.

Συνολικά, το Edge Impulse παρέχει ένα ολοκληρωμένο σετ εργαλείων για την ανάπτυξη, την εκπαίδευση και την εκτέλεση μοντέλων μηχανικής μάθησης σε ενσωματωμένα συστήματα.



Εικόνα 8: Πλατφόρμα Edge impulse



Εικόνα 9: Data explorer

2.2 Μικροελεγκτές

Ένας μικροελεγκτής είναι ένα ηλεκτρονικό κύκλωμα ή μικροσίπ που περιλαμβάνει έναν ενσωματωμένο επεξεργαστή (μικροεπεξεργαστή) και άλλα στοιχεία όπως μνήμη, αισθητήρες εισόδου/εξόδου και περιφερειακά κυκλώματα. Ο μικροελεγκτής λειτουργεί ως κέντρο ελέγχου για ένα ηλεκτρονικό σύστημα ή μια συσκευή, εκτελώντας προγραμματιζόμενες λειτουργίες και αλγορίθμους. Είναι σχεδιασμένος για να εκτελεί επεξεργασία και ελεγχόμενες εργασίες σε εφαρμογές όπου απαιτείται χαμηλή κατανάλωση ενέργειας, μικρό μέγεθος, χαμηλό κόστος και αξιοπιστία.

Οι μικροελεγκτές παρέχουν μια πλατφόρμα για την υλοποίηση λειτουργιών εισόδου/εξόδου, επεξεργασίας δεδομένων, ελέγχου και επικοινωνίας με άλλες

συσκευές. Οι πιο συνηθισμένοι μικροελεγκτές που χρησιμοποιούνται στον τομέα της ενσωματωμένης ηλεκτρονικής είναι οι Arduino, Raspberry Pi, PIC, STM32 και ESP32.

2.3 Arduino

Το Arduino είναι μια πλατφόρμα ανοιχτού κώδικα για την ανάπτυξη πρωτοτύπων και έργων ηλεκτρονικής. Αποτελείται από έναν μικροελεγκτή (microcontroller) και ένα περιβάλλον ανάπτυξης λογισμικού (IDE), που επιτρέπει σε μηχανικούς, ερασιτέχνες και φοιτητές να δημιουργήσουν διάφορες διαδραστικές και έξυπνες ηλεκτρονικές συσκευές. Βασίζεται στο εύχρηστο υλικό και λογισμικό που χρησιμοποιεί.

Τα Arduino boards είναι μικρές πλακέτες που περιλαμβάνουν έναν μικροελεγκτή, όπως τον ATmega328 στην περίπτωση του Arduino Uno. Αυτοί οι μικροελεγκτές είναι προγραμματιζόμενοι, που σημαίνει ότι μπορείτε να γράψετε κώδικα και να τους προγραμματίσετε να εκτελέσουν συγκεκριμένες λειτουργίες και να αλληλοεπιδράσουν με αισθητήρες, κινητήρες, οθόνες και άλλες συσκευές.

Το περιβάλλον ανάπτυξης Arduino IDE παρέχει έναν απλό τρόπο για τη γραφή και τη μεταφόρτωση του κώδικα στο Arduino board καθώς κώδικας γράφεται σε μια παραλλαγή της γλώσσας προγραμματισμού C++.

Με το Arduino, μπορείτε να δημιουργήσετε διάφορα έργα, όπως ρομπότ, έξυπνα συστήματα ελέγχου και πολλά άλλα. Η απλότητα και η ευελιξία του το καθιστούν ευρέως αποδεκτό και δημοφιλές στην κοινότητα των επαγγελματιών και των ερασιτεχνών τεχνολογίας.



Εικόνα 10: Διάφορα είδη Arduino

2.3.1 Πλεονεκτήματα Arduino

Η πλακέτα Arduino έχει πολλά πλεονεκτήματα που την καθιστούν δημοφιλή και ευρέως χρησιμοποιούμενη στον χώρο της ενσωματωμένης ηλεκτρονικής. Ανάμεσα στα πλεονεκτήματά της περιλαμβάνονται:

1. Ευκολία Χρήσης: Η πλακέτα Arduino είναι σχεδιασμένη με έμφαση στην ευκολία χρήσης ακόμα και για αρχάριους. Έχει φιλικό προς τον χρήστη περιβάλλον ανάπτυξης και απλή σύνδεση με την υπολογιστική μονάδα.
2. Ευελιξία: Η πλακέτα Arduino παρέχει πολλές ψηφιακές και αναλογικές ακροδέκτες εισόδου/εξόδου για τη σύνδεση αισθητήρων, ενεργοποιητών

και άλλων περιφερειακών συσκευών. Αυτή η ευελιξία επιτρέπει την ανάπτυξη διάφορων εφαρμογών και τον έλεγχο διαφόρων συσκευών.

3. Κοινότητα και Υποστήριξη: Η πλατφόρμα Arduino έχει μια ζωντανή κοινότητα χρηστών και προγραμματιστών που μοιράζονται ιδέες, κώδικα και πληροφορίες μεταξύ τους. Αυτή η κοινότητα παρέχει υποστήριξη και λύσεις σε προβλήματα που μπορεί να αντιμετωπίσει κάποιος χρήστης.
4. Ευκολία Επέκτασης: Η πλακέτα υποστηρίζει τη χρήση επιπλέον προσθέτων (shields) που επεκτείνουν τις δυνατότητές της. Αυτά τα πρόσθετα μπορούν να παρέχουν επιπλέον λειτουργίες όπως ασύρματη επικοινωνία, ελέγχους μοτέρ, οθόνες και πολλά άλλα.
5. Χαμηλό κόστος: Η πλακέτα Arduino και τα περιφερειακά της είναι συνήθως οικονομικά, κάνοντας την κατάλληλη για προσωπικά ή εκπαιδευτικά έργα.

2.3.2 Λειτουργία Πλακέτας

Η πλακέτα Arduino βασίζεται σε έναν μικροελεγκτή, ο οποίος είναι ένας μικρός υπολογιστής που μπορεί να προγραμματιστεί για να εκτελεί διάφορες εργασίες. Διαθέτει αναλογικούς και ψηφιακούς ακροδέκτες (analog / digital pins) που μπορούν να χρησιμοποιηθούν για τη σύνδεση αισθητήρων, ενεργοποιητών και άλλων ηλεκτρονικών εξαρτημάτων. Η πλακέτα διαθέτει επίσης θύρα USB που της επιτρέπει να επικοινωνεί με έναν υπολογιστή για προγραμματισμό και μεταφορά δεδομένων.

Η Πλακέτα Arduino όντας ένα σύστημα ανοιχτού κώδικα περιλαμβάνει μια πλακέτα μικροελεγκτή και ένα περιβάλλον ανάπτυξης λογισμικού. Ο μικροελεγκτής είναι υπεύθυνος για την εκτέλεση του προγράμματος, ενώ το περιβάλλον ανάπτυξης λογισμικού παρέχει έναν επεξεργαστή κώδικα, έναν μεταγλωττιστή και έναν φορτωτή εκκίνησης. Προγραμματίζεται με τη χρήση της γλώσσας προγραμματισμού Arduino, η οποία βασίζεται στη γλώσσα C++ και περιλαμβάνει έναν επεξεργαστή κώδικα (code editor), έναν μεταγλωττιστή (compiler) και έναν φορτωτή εκκίνησης (bootloader).

Ο επεξεργαστής κώδικα παρέχει μια χρήσιμη διεπαφή για τη συγγραφή κώδικα, ο μεταγλωττιστής μεταφράζει τον κώδικα σε γλώσσα μηχανής που μπορεί να κατανοήσει ο μικροελεγκτής και ο φορτωτής εκκίνησης είναι υπεύθυνος για τη μεταφόρτωση του κώδικα στην πλακέτα Arduino μέσω της σύνδεσης USB. Για να χρησιμοποιηθεί μια πλακέτα Arduino, πρέπει να τη συνδεθεί στον υπολογιστή σας μέσω καλωδίου USB. Στη συνέχεια, γράφεται το πρόγραμμα, γνωστό και ως σκίτσο (sketch), χρησιμοποιώντας το περιβάλλον ανάπτυξης λογισμικού Arduino IDE. Το σκίτσο ελέγχει τη συμπεριφορά της πλακέτας, όπως την ανάγνωση αισθητήρων και την ενεργοποίηση των εξόδων.

Αφού ολοκληρωθεί ο κώδικας, μεταφορτώνεται στην πλακέτα Arduino μέσω της σύνδεσης USB. Ο φορτωτής εκκίνησης της πλακέτας αναλαμβάνει να παραλάβει το σκίτσο και να το αποθηκεύσει στη μνήμη του μικροελεγκτή. Έπειτα, ο μικροελεγκτής εκτελεί τις εντολές που του έχουν δοθεί, ελέγχοντας τη συμπεριφορά της πλακέτας και των συνδεδεμένων ηλεκτρονικών εξαρτημάτων. Συνολικά, η πλακέτα Arduino σας επιτρέπει να διαβάζετε εισόδους από αισθητήρες ή κουμπιά και να τις μετατρέπετε σε έξοδο, ελέγχοντας την κατάσταση των LED, την ενεργοποίηση μοτέρ ή την αποστολή δεδομένων σε άλλες συσκευές.

2.3.3 Βασικά μέρη περιβάλλοντος ανάπτυξης Arduino

Τα παρακάτω είναι σημαντικά στοιχεία του περιβάλλοντος ανάπτυξης Arduino, και κάθε ένα από αυτά έχει έναν συγκεκριμένο ρόλο στη διαδικασία του προγραμματισμού και της ανάπτυξης:

Επεξεργαστής Πηγαίου Κώδικα (Code Editor)

Ο επεξεργαστής πηγαίου κώδικα είναι η κύρια περιοχή όπου γράφεται ο προγραμματιστικός κώδικας της εφαρμογής. Εδώ μπορούν να δημιουργηθούν οι λειτουργίες της εφαρμογής, να δηλωθούν μεταβλητές και να δομηθεί η λογική του προγράμματος.

Εργαλεία Προεπισκόπησης Κώδικα

Αυτά τα εργαλεία επιτρέπουν μια προεπισκόπηση του κώδικα πριν ανέβει στον μικροελεγκτή. Μπορεί να παρέχουν επίσης χρήσιμες πληροφορίες για συντακτικά λάθη ή προειδοποιήσεις.

Σειριακή Κονσόλα

Η σειριακή κονσόλα (Serial Monitor) είναι ένα εργαλείο που σας επιτρέπει να επικοινωνείτε με το Arduino μέσω της σειριακής θύρας. Μπορείτε να χρησιμοποιήσετε αυτό το εργαλείο για να παρακολουθείτε τα μηνύματα που στέλνει και λαμβάνει το Arduino, καθώς και για να αποσφαλματώνετε τον κώδικά σας.

Board Manager (Διαχείριση Πλακετών)

Το Board Manager είναι ένα εργαλείο που σας επιτρέπει να προσθέτετε νέες πλακέτες (boards) στο περιβάλλον ανάπτυξης Arduino. Μπορείτε να επιλέξετε το μοντέλο του μικροελεγκτή που θέλετε να χρησιμοποιήσετε και το περιβάλλον θα

εγκαταστήσει τις απαραίτητες βιβλιοθήκες και ρυθμίσεις για να μπορείτε να προγραμματίσετε τη συγκεκριμένη πλακέτα.

Library Manager (Διαχείριση Βιβλιοθηκών)

Το Library Manager σας επιτρέπει να προσθέτετε και να διαχειρίζεστε βιβλιοθήκες κώδικα που δημιουργήθηκαν από την κοινότητα ή από τους κατασκευαστές συσκευών. Οι βιβλιοθήκες αυτές περιλαμβάνουν προ-γραμμένο κώδικα που μπορείτε να χρησιμοποιήσετε για να απλοποιήσετε τον προγραμματισμό και να προσθέσετε λειτουργίες στη συσκευή σας.

Εργαλεία μεταφόρτωσης(Upload Tools)

Τα Upload Tools είναι κρίσιμα για τον προγραμματισμό του μικροελεγκτή. Χρησιμοποιούνται για να μεταφέρετε τον προγραμματισμένο κώδικα από τον υπολογιστή στον μικροελεγκτή, προετοιμάζοντας τον για την εκτέλεση του προγράμματος.

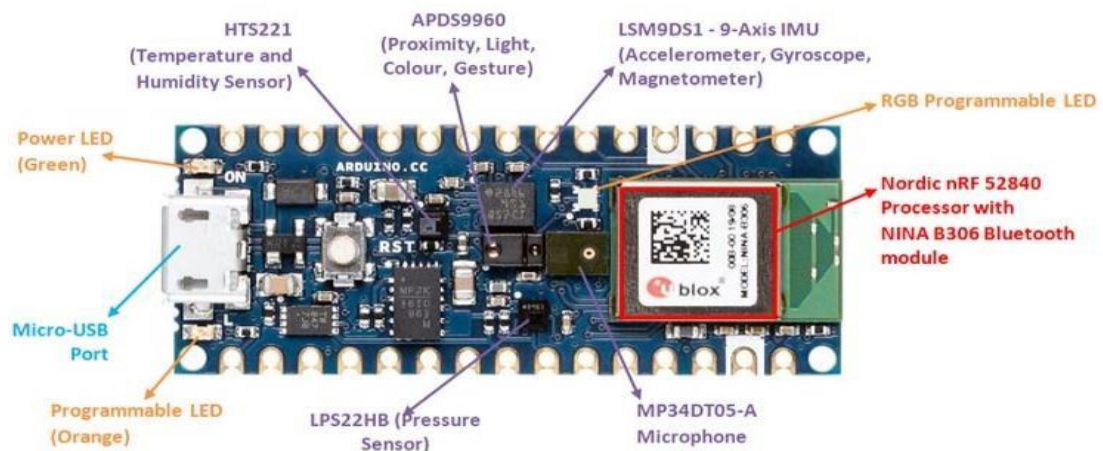


Εικόνα 11: Arduino IDE

2.4 Arduino Nano 33 Ble Sense

Το Arduino Nano 33 BLE Sense αντιπροσωπεύει μια σημαντική πρόοδο στον τομέα της μικροηλεκτρονικής και των ενσωματωμένων συστημάτων. Αυτός ο μικροελεγκτής προσφέρει ένα ευρύ φάσμα δυνατοτήτων που συνδυάζει την ισχύ της υψηλής τεχνολογίας με την ευκολία χρήσης.

Η συσκευή διαθέτει έναν ενσωματωμένο αισθητήρα που μπορεί να μετρήσει διάφορες παραμέτρους όπως θερμοκρασία, υγρασία, κίνηση, πίεση και ποιότητα αέρα. Οι αισθητήρες αυτοί προσφέρουν μια πολυδιάστατη αίσθηση του περιβάλλοντος, κάνοντας το Arduino Nano 33 BLE Sense ιδανικό για εφαρμογές που απαιτούν ανίχνευση και αναγνώριση περιβαλλοντικών συνθηκών. Η ισχυρή απόδοση του μικροελεγκτή σε συνδυασμό με την ποικιλία των αισθητήρων του και την ευκολία προγραμματισμού του παρέχουν τον ιδανικό χώρο για την υλοποίηση του μοντέλου μηχανικής μάθησης που έχει επιλεγεί. Τα δεδομένα που παρέχονται από τους αισθητήρες μπορούν να χρησιμοποιηθούν για την εκπαίδευση και την αξιολόγηση του μοντέλου, επιτρέποντας την ανάπτυξη μιας εφαρμογής που μπορεί να αναγνωρίσει και να ανταποκριθεί σε συγκεκριμένες περιβαλλοντικές συνθήκες.



Εικόνα 12: Arduino Nano 33 BLE Sense - Hardwar

2.4.1 Τεχνικά χαρακτηριστικά

Το Arduino Nano 33 BLE Sense αντλεί τη δύναμή του από τον εξαιρετικά ισχυρό μικροελεγκτή Nordic nRF52840. Αυτός ο μικροελεγκτής λειτουργεί στα 64 MHz, προσφέροντας εξαιρετική απόδοση για εφαρμογές πραγματικού χρόνου. Το Nordic nRF52840 διαθέτει εντυπωσιακές τεχνικές προδιαγραφές, συμπεριλαμβανομένης της μνήμης flash 1 MB και της μνήμης RAM 256 KB. Με αυτήν τη μνήμη, ο μικροελεγκτής μπορεί να υποστηρίξει πολύπλοκες εφαρμογές και αναλύσεις δεδομένων.

Είναι ενδιαφέρον να σημειώσουμε ότι η πλακέτα μπορεί να προγραμματιστεί με το Arduino IDE, παρέχοντας ένα φιλικό περιβάλλον ανάπτυξης για προγραμματισμό και δοκιμές. Επιπλέον, μπορεί να χρησιμοποιηθεί και με άλλες γλώσσες προγραμματισμού όπως η Python, η JavaScript και η C++, επεκτείνοντας τις δυνατότητες ανάπτυξης. Όσον αφορά τους αισθητήρες, το Arduino Nano 33 BLE Sense διαθέτει αισθητήρες κίνησης 9 αξόνων (3 αξόνων επιτάχυνσης, 3 αξόνων γυροσκοπίου BMI270 και 3 αξόνων μαγνητόμετρου BMM150), οπτικούς αισθητήρες για την ανίχνευση χειρονομιών, εγγύτητας, ανίχνευση φωτός περιβάλλοντος (ALS) και ανίχνευση χρώματος (RGBC) APDS9960, ενσωματωμένο μικρόφωνο MP34DT06JTR, και αισθητήρες θερμοκρασίας και υγρασίας HS3003.

Το Bluetooth Low Energy (BLE) δίνει τη δυνατότητα ασύρματης σύνδεσης με άλλες συσκευές και εφαρμογές. Αυτή η τεχνολογία επιτρέπει τη μεταφορά δεδομένων με χαμηλή κατανάλωση ενέργειας, καθιστώντας το Arduino Nano 33 BLE Sense ιδανικό για εφαρμογές που απαιτούν ασύρματη σύνδεση και ανταλλαγή δεδομένων με άλλες συσκευές.

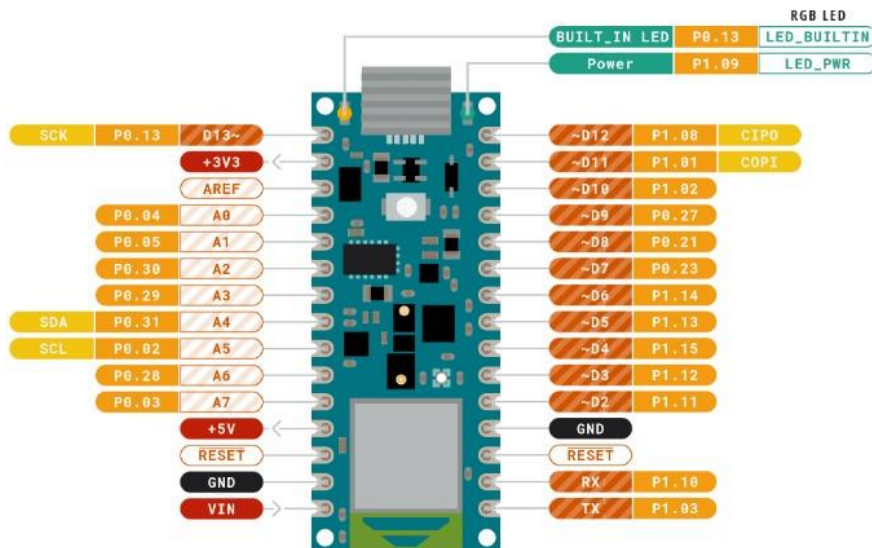
Sensors	IMU	BMI270 (3-axis accelerometer + 3-axis gyroscope) + BMM150 (3-axis Magnetometer)
	Microphone	MP34DT06JTR
	Gesture, light, proximity	APDS9960
	Barometric pressure	LPS22HB
	Temperature, humidity	HS3003

Εικόνα 13:Τεχνικά Χαρακτηριστικά Πλακέτας

Σχετικά με τις επιλογές συνδεσμολογίας της πλακέτας), το Arduino Nano 33 BLE Sense περιλαμβάνει, 8 αναλογικές (analog) και 14 ψηφιακές (digital), ενώ 5 από αυτές είναι PWM (Pulse Width Modulation), για την λήψη αναλογικών αποτελεσμάτων με ψηφιακά μέσα. Επιπροσθέτως, συναντώνται 2 ακίδες γείωσης (GND), μια ακίδα για είσοδο 3,3V, μία για 5V, καθώς και η Vin, η οποία μπορεί να λάβει τιμές από 4,5V, έως 21V.

Για να συνδέσεις το Arduino με έναν υπολογιστή ή άλλο μικροελεγκτή, χρησιμοποιείται η σειριακή επικοινωνία. Αυτό συμβαίνει μέσω ενός ειδικού καλωδίου που συνδέει δύο ακροδέκτες των συσκευών. Στον κόσμο της σειριακής επικοινωνίας, αυτοί οι δύο ακροδέκτες παίζουν το ρόλο του πομπού και του δέκτη.

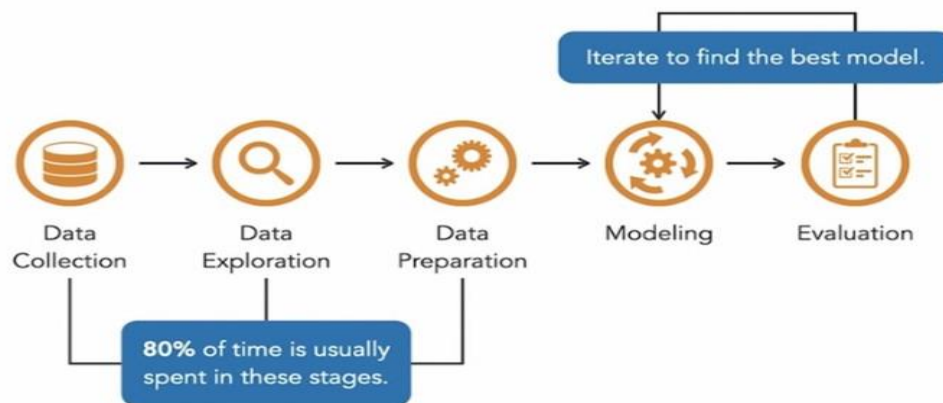
Ο ακροδέκτης TX (Transmitter) είναι υπεύθυνος για την αποστολή δεδομένων, ενώ ο ακροδέκτης RX (Receiver) για τη λήψη δεδομένων.



Εικόνα 14:Ακίδες (Pins) Arduino Nano 33 BLE Sense

ΚΕΦΑΛΑΙΟ 3^ο « ΑΝΑΠΤΥΞΗ ΜΟΝΤΕΛΟΥ ΑΝΑΓΝΩΡΙΣΗΣ ΕΝΤΟΛΩΝ »

Για να δημιουργηθεί ένα σύστημα που μπορεί να αναγνωρίζει φωνητικές εντολές με βάση τη μηχανική μάθηση και να ενσωματωθεί σε μια πλακέτα, όπως το Arduino, χρειάζονται μερικά συγκεκριμένα βήματα. Αρχικά, επιλέγεται η συσκευή και οι εντολές που θα χρησιμοποιηθούν για τον έλεγχο της. Αφού αυτές επιλεγούν, το επόμενο βήμα είναι η συλλογή δεδομένων που θα χρησιμοποιηθούν για την εκπαίδευση του μοντέλου. Μόλις συλλεχθούν τα απαραίτητα δεδομένα, θα πρέπει να υποστούν κάποιες διαδικασίες προεπεξεργασίας για να είναι έτοιμα για την εκπαίδευση του μοντέλου. Στη συνέχεια, θα ακολουθήσει η φάση της εκπαίδευσης του μοντέλου μηχανικής μάθησης, κατά την οποία το μοντέλο θα μάθει από τα δεδομένα που του παρέχονται. Θα αξιολογηθεί η απόδοση του μοντέλου σε σύνολο δοκιμών, προκειμένου να μετρηθεί πόσο καλά λειτουργεί. Τέλος, θα πρέπει να μετατραπεί το μοντέλο σε μορφή που μπορεί να εκτελεστεί στη συγκεκριμένη συσκευή. Για να μπορέσει το μοντέλο να φορτωθεί στη συσκευή και να ενσωματωθεί με τους αισθητήρες της.



Εικόνα 15: Αλυσίδα machine learning

3.1 Επιλογή Συσκευής και Φωνητικών Εντολών

Πριν ξεκινήσει η διαδικασία συλλογής και η προετοιμασία των δεδομένων πρέπει να γίνει η επιλογή της συσκευής που θα χρησιμοποιηθεί όπως επίσης και των εντολών που θα επικαλεστούν για τον χειρισμό του συστήματος.

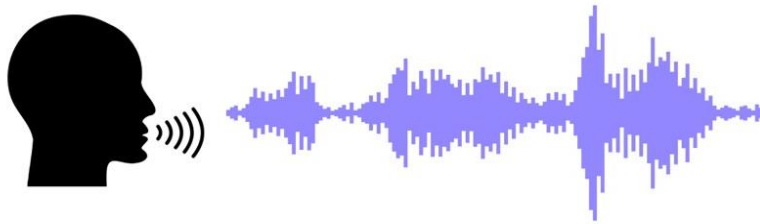
Η επιλογή της συσκευής αποτέλεσε κρίσιμο παράγοντα στην υλοποίηση του πρότζεκτ μου για τη διπλωματική εργασία. Μετά από έρευνα, κατέληξα στην πλακέτα Arduino Nano 33 BLE Sense, καθώς διαθέτει όλα τα απαραίτητα χαρακτηριστικά που χρειαζόμουν. Συγκεκριμένα, οι ενσωματωμένοι αισθητήρες συνέβαλαν σημαντικά στην υλοποίηση της αρχικής μου ιδέας. Επιπλέον, το μικρό μέγεθος της πλακέτας μου προσέφερε την ευελιξία να την ενσωματώσω και να την χρησιμοποιήσω όπως επιθυμούσα.

Οι φωνητικές εντολές που επέλεξα για τον χειρισμό της πλακέτας ήθελα να συμβαδίζουν με λειτουργίες που πρόκειται να ενεργοποιεί. Οι λειτουργίες που επιλέχθηκαν είναι:

1. Το άναμμα μίας διόδου εκπομπής φωτός (LED, light-emitting diode).
2. Το σβήσιμο της ίδιας διόδου εκπομπής φωτός (LED, light-emitting diode).
3. Η ανίχνευση και η εμφάνιση της θερμοκρασίας του χώρου σε μία οθόνη υγρών κρυστάλλων (LCD, liquid crystal display).
4. Η ανίχνευση και η εμφάνιση της υγρασίας του χώρου στην ίδια οθόνη υγρών κρυστάλλων (LCD, liquid crystal display).

Έχοντας σκοπό η χρήση της συσκευής να είναι εύκολη και προσιτή σε όλους, οι φωνητικές εντολές που επιστρατεύτηκαν είναι αντιπροσωπευτικές για της

παραπάνω λειτουργίες. Για την πρώτη επιλέχθηκε η εντολή «άναψε», για την δεύτερη «κλείσε», ενώ για την τέταρτη και την πέμπτη λειτουργία επιλέχθηκαν οι εντολές «θερμοκρασία» και «υγρασία» αντίστοιχα.



Εικόνα 16: Μετατροπή ομιλίας σε σήμα

3.2 Συλλογή δεδομένων

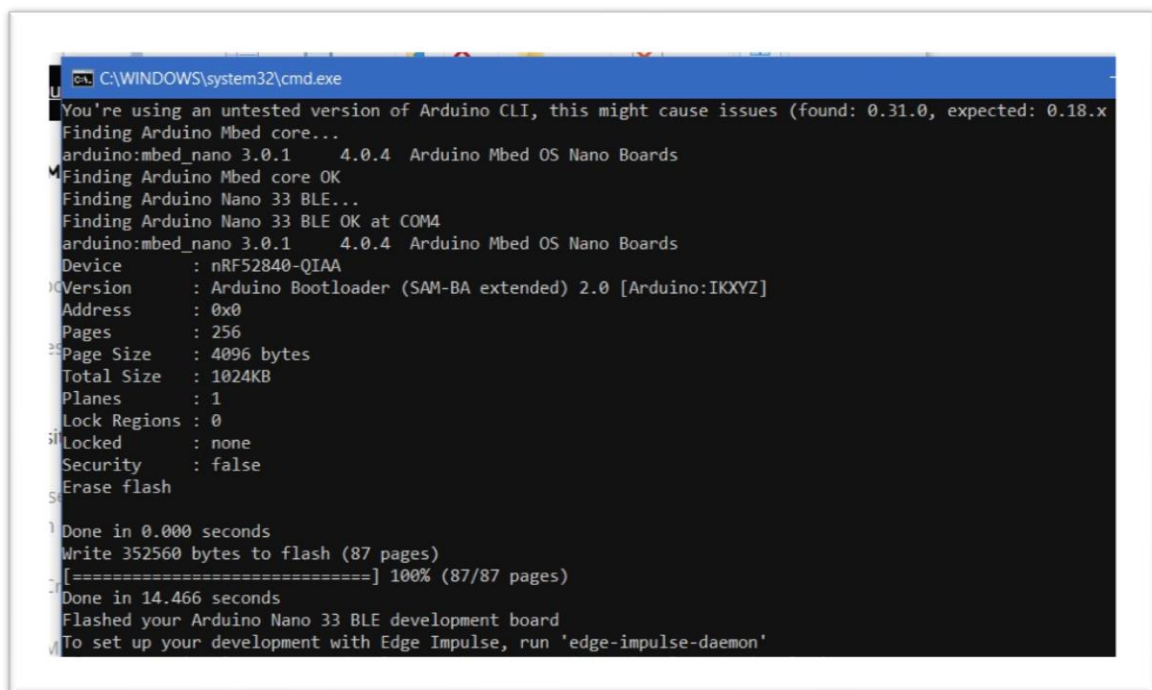
Η συλλογή δεδομένων αποτελεί το πρώτο βήμα για τη δημιουργία οποιουδήποτε μοντέλου μηχανικής μάθησης. Στην περίπτωση τις συγκεκριμένης διπλωματικής τα δεδομένα που συλλέχθηκαν είναι φωνητικές εντολές, δηλαδή ηχητικά σήματα που αποθηκεύονται ως ηλεκτρονικά δεδομένα.

Το Arduino Nano 33 BLE Sense είναι μια δημοφιλής πλακέτα μικροελεγκτή που είναι εξοπλισμένη με μια σειρά αισθητήρων, συμπεριλαμβανομένου ενός αισθητήρα ήχου, ο οποίος αξιοποιήθηκε για την συλλογή του μεγαλύτερου συνόλου των δεδομένων.

Το Edge Impulse διαθέτει αρχεία ηχητικών δεδομένων από απλές αγγλικές λέξεις, όπως «yes», «no» και αρχεία θορύβου ελευθέρως για χρήση. Σε περίπτωση που τα αρχεία αυτά δεν μας καλύπτουν μπορούμε να εισάγουμε εκ νέου δεδομένα. Η απόκτηση των δεδομένων μπορεί να γίνει είτε εισάγοντας πακέτα δεδομένων που μπορούμε να βρούμε στο διαδίκτυο, είτε μπορούμε να κάνουμε οι ίδιοι την

δειγματοληψία. Στην συγκεκριμένη διπλωματική η πλειονότητα των δεδομένων συλλέχθηκε από το μικρόφωνο του Arduino Nano 33 BLE Sense μέσω του Edge Impulse.

Για να ξεκινήσει η δειγματοληψία πρέπει το Arduino Nano 33 BLE Sense και το Edge Impulse να συνδεθούν. Συνδέοντας τη συσκευή στην υπολογιστική μονάδα, ανοίγοντας το Command Prompt και δίνοντας συγκεκριμένες εντολές γίνεται η σύνδεση και έπειτα είναι εφικτή η χρήση του μικροφώνου.



```
C:\WINDOWS\system32\cmd.exe
You're using an untested version of Arduino CLI, this might cause issues (found: 0.31.0, expected: 0.18.x)
Finding Arduino Mbed core...
arduino:mbed_nano 3.0.1 4.0.4 Arduino Mbed OS Nano Boards
Finding Arduino Mbed core OK
Finding Arduino Nano 33 BLE...
Finding Arduino Nano 33 BLE OK at COM4
arduino:mbed_nano 3.0.1 4.0.4 Arduino Mbed OS Nano Boards
Device      : nRF52840-QIAA
Version     : Arduino Bootloader (SAM-BA extended) 2.0 [Arduino:IKXYZ]
Address     : 0x0
Pages      : 256
Page Size  : 4096 bytes
Total Size : 1024KB
Planes    : 1
Lock Regions : 0
Locked    : none
Security  : false
Erase flash
Done in 0.000 seconds
Write 352560 bytes to flash (87 pages)
[=====] 100% (87/87 pages)
Done in 14.466 seconds
Flashed your Arduino Nano 33 BLE development board
To set up your development with Edge Impulse, run 'edge-impulse-daemon'
```

Εικόνα 17: Σύνδεση Arduino με τον υπολογιστή

Ανοίγοντας την πλατφόρμα του Edge Impulse διακρίνουμε ότι η συσκευή πλέον εμφανίζεται και είναι επιλέξιμη στις επιλογές καταγραφής.

The image shows a web interface titled "Collect data" with a microphone icon in the top right corner. Below the title, there is a "Device" section with a dropdown menu showing the MAC address "84:EF:2A:B4:08:8A". Below this, there are four input fields: "Label" with the text "kleise", "Sample length (ms.)" with the value "19000", "Sensor" with a dropdown menu showing "Built-in microphone", and "Frequency" with a dropdown menu showing "16000Hz". At the bottom right of the form is a dark blue button labeled "Start sampling".

Εικόνα 18: Προετοιμασία συλλογής δεδομένων

Επόμενο βήμα είναι να θέσουμε την ετικέτα (Label) λέξεων που θέλουμε να ηχογραφήσουμε και τον χρόνο που κάθε δειγματοληψία (Sample length) θα διαρκεί, αναγραφόμενη σε χιλιοστά του δευτερολέπτου (milliseconds, ms)

Μέσω του Edge Impulse, χρησιμοποιώντας το μικρόφωνο της συσκευής Arduino για συλλογή δειγμάτων ο μέγιστος χρόνος που μπορούμε να ηχογραφήσουμε είναι τα 19000 ms., όπως αναγράφεται. Στην διάρκεια των 19 δευτερολέπτων της ηχογράφησης ο ομιλητής επαναλαμβάνει πολλές φορές τη λέξη που θα χρησιμοποιηθεί για τη δημιουργία του μοντέλου.

Πατώντας το κουμπί «Start sampling» θα ξεκινήσει η διαδικασία ηχογράφησης.

Ο ομιλητής θα πρέπει να προσπαθήσει να καταγράψει τη φωνή του με καθαρότητα και διακριτικότητα. Είναι βασικό τα δεδομένα να είναι εύηχα αφού με τη χρήση αυτών θα γίνει η δημιουργία του μοντέλου.

Κατά την ηχογράφηση το περιβάλλον θα πρέπει να είναι ήσυχο χωρίς παρεμβάλλοντες θορύβους, είναι σημαντικό να διασφαλίσουμε ότι η φωνή του ομιλητή καταγράφεται καθαρά και χωρίς ανεπιθύμητες πηγές θορύβου που θα μπορούσαν να επηρεάσουν την ποιότητα των δεδομένων.

Εάν το περιβάλλον είναι θορυβώδες, τα δεδομένα που καταγράφονται κατά τη διάρκεια της ηχογράφησης μπορεί να προσθέσουν παραπλανητικά στοιχεία. Αυτό μπορεί να επηρεάσει την ακρίβεια του μοντέλου μηχανικής μάθησης που εκπαιδεύετε με αυτά τα δεδομένα.

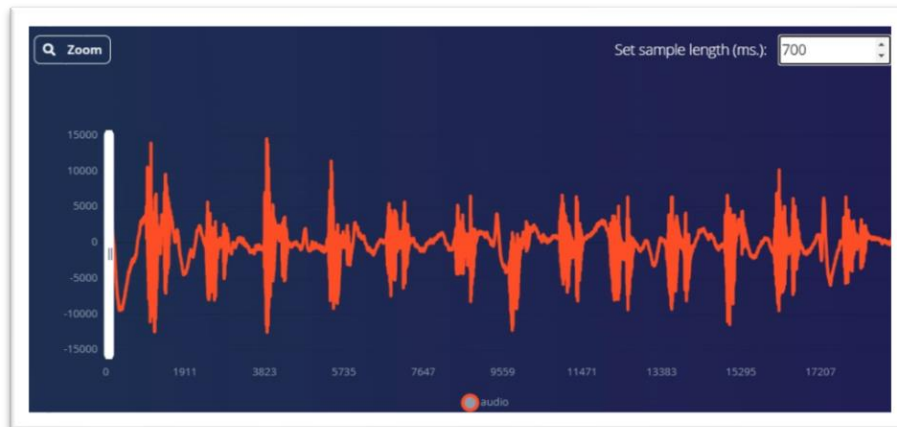
Η ποσότητα των δειγμάτων που χρειάζεται για να είναι ένα μοντέλο αρκετά αποδοτικό και αξιόπιστο, μετά από έρευνα διαπιστώθηκε ότι είναι περίπου πέντε λεπτά δειγμάτων από κάθε λέξη που θέλουμε αυτό να ανιχνεύει. Για αυτό το μοντέλο συλλέχθηκαν συνολικά 36 λεπτά και 13 δευτερόλεπτα δεδομένων. Πιο συγκεκριμένα συλλέχθηκαν για την κάθε λέξη περίπου 7 λεπτά δεδομένων, όπως επίσης συλλέχθηκαν και 7 λεπτά από ήχους που μπορεί να εντοπιστούν στην καθημερινότητα.



Εικόνα 19: Σύνολο δεδομένων

3.3 Επεξεργασία δεδομένων

Αφού γίνει η συλλογή των δεδομένων, περνάμε στο επόμενο βήμα της δημιουργίας του μοντέλου, το οποίο είναι η επεξεργασία των δεδομένων που συλλέχθηκαν.



Εικόνα 20: Πρώτη εικόνα ενός δείγματος

Στην εικόνα του δείγματος, μπορούμε να διακρίνουμε σε ποια σημεία της ηχογράφησης εντοπίζονται τα στοιχεία που μας ενδιαφέρουν. Έτσι θέτοντας ένα αναφορικό μήκος δείγματος και εφαρμόζοντας το, έχουμε αποτέλεσμα την πλαισίωση των κύριων σημείων του δείγματος. Τα μέρη με τις υψηλότερες συχνότητες είναι αυτά που ανιχνεύονται οι λέξεις, ενώ τα υπόλοιπα είναι οι παύσεις ανάμεσά τους.

Πατώντας «Apply» (εφαρμογή) γίνεται διαχωρισμός του δείγματος στα επιλεγμένα μικρότερα γκρι πλαίσια.



Εικόνα 21:Εικόνα δείγματος πριν διαχωριστεί

Με αυτόν τον τρόπο έχουμε μία καθαρή εικόνα από το κάθε δείγμα ξεχωριστά.

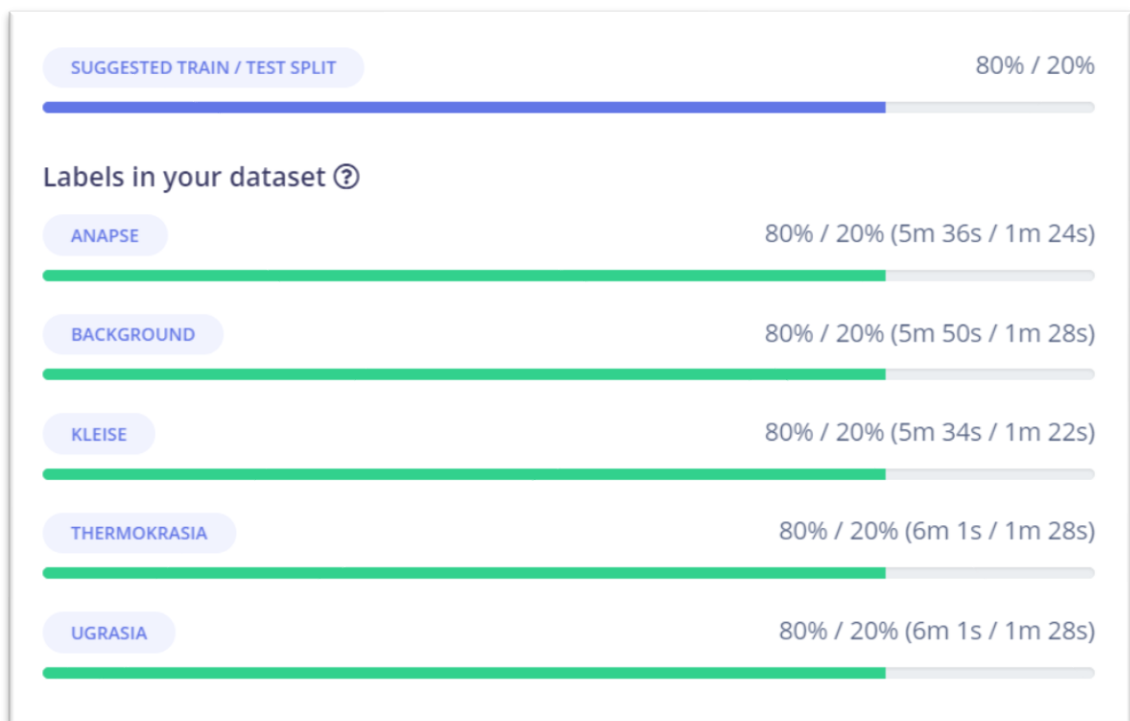
Επόμενο βήμα της επεξεργασίας των δειγμάτων είναι η αφαίρεση των απαραίτητων τμημάτων δεξιά και αριστερά από αυτά, κρατώντας τις λέξεις όσο περισσότερο «καθαρές» γίνεται, χωρίς περιττούς ήχους ή κενό χρόνο.

Στο τέλος, κάθε δείγμα θα πρέπει να μοιάζει κάπως έτσι:



Εικόνα 22: Εικόνα ενός έτοιμου δείγματος

Αφού συγκεντρωθούν τα απαιτούμενα δεδομένα, χωρίζονται σε δεδομένα εκπαίδευσης (train data) και δεδομένα δοκιμής (test data) σε μία κλίμακα 80% - 20% αντίστοιχα. Αυτό συμβαίνει ώστε να μπορούμε να αξιολογήσουμε την απόδοση και την ικανότητα γενίκευσης των μοντέλων μηχανικής μάθησης. Με τον διαχωρισμό των δεδομένων μπορούμε να αξιολογήσουμε πόσο καλά το μοντέλο μας έχει μάθει από τα δεδομένα εκπαίδευσης και πόσο αποτελεσματικά μπορεί να κάνει προβλέψεις σε αθέατα δεδομένα.



Εικόνα 23: Ολοκλήρωση συλλογής δεδομένων

3.3.1 Επιλογή χαρακτηριστικών

Επόμενη φάση της δημιουργίας του μοντέλου είναι η επιλογή των χαρακτηριστικών των δεδομένων. Το Edge impulse παρέχει τη δυνατότητα διαμόρφωσης τους μέσω καρτελών.

Η πρώτη καρτέλα, «Δεδομένα χρονοσειρών» (Time series data) αναφέρεται σε δεδομένα που καταγράφουν παρατηρήσεις, μετρήσεις ή αλλαγές μεταξύ διαφόρων χρονικών στιγμών.

Εκεί ρυθμίζουμε το μέγεθος του παραθύρου (Window size) που αναφέρεται στο μέγεθος των τμημάτων των δεδομένων που χρησιμοποιούνται για την εκπαίδευση. Στην περίπτωση του έργου το «Window size» ρυθμίζεται στα 1000 ms. (milliseconds) που είναι ένας μέσος όρος της διάρκειας όλων των δειγμάτων που χρησιμοποιήθηκαν.

Η αύξηση του παραθύρου (Window increase) χρησιμοποιείται για την τεχνητή δημιουργία περισσότερων χαρακτηριστικών (features) την τροφοδότηση, δηλαδή του μπλοκ μάθησης με περισσότερες πληροφορίες.

Η συχνότητα (frequency) υπολογίζεται αυτόματα βάσει των δειγμάτων εκπαίδευσης, η οποία ισούται με 16.000 hz.

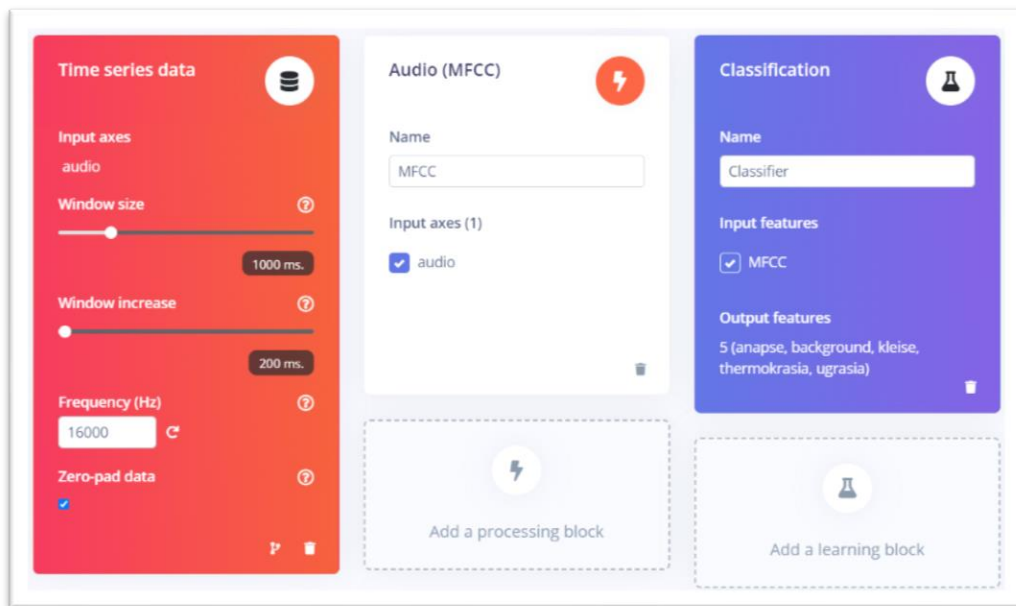
Η μηδενική συμπλήρωση (Zero-pad data) επιτρέπει το διάβασμα των δεδομένων όταν αυτά είναι μικρότερα σε μήκος από αυτό που ορίζεται στο Window size.

Έπειτα, πρέπει να επιλέξουμε στο «Μπλοκ επεξεργασίας» (Processing block) ένα προεκπαιδευμένο μοντέλο που θα χρησιμοποιηθεί για να ενσωματωθούν και να εκπαιδευτούν τα δεδομένα. Το μοντέλο Mel-frequency cepstral coefficients (MFCC)

αποδείχθηκε η αποδοτικότερη επιλογή. Το συγκεκριμένο χρησιμοποιεί μία τεχνική εξαγωγής χαρακτηριστικών που χρησιμοποιείται ευρέως στην επεξεργασία ομιλίας και ήχου. Ο αλγόριθμος MFCC βασίζεται στην αντίληψη του ήχου από το ανθρώπινο ακουστικό σύστημα, το οποίο αναλύει τα ηχητικά σήματα σε ζώνες συχνότητας.

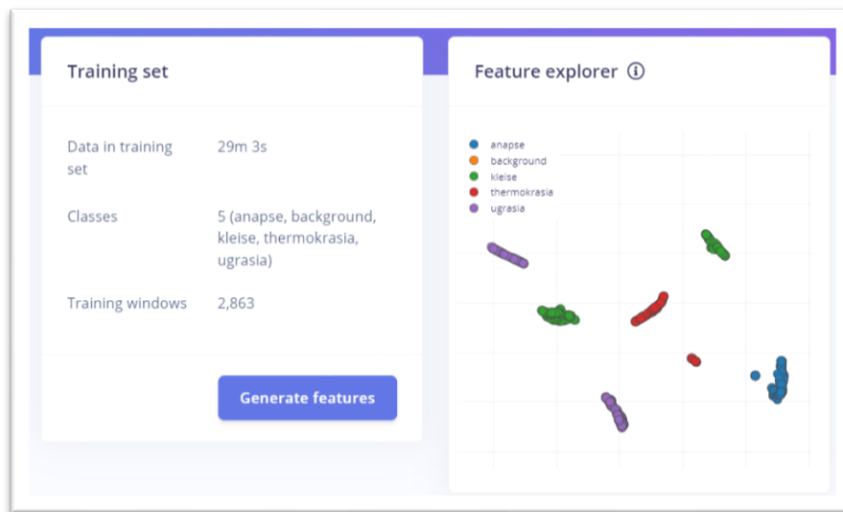
Οι MFCC λαμβάνονται εφαρμόζοντας πρώτα τον μετασχηματισμό Fourier μικρού χρόνου (STFT) σε ένα σήμα για να ληφθεί η φασματική του αναπαράσταση. Στη συνέχεια, το φάσμα ισχύος του σήματος αντιστοιχίζεται στην κλίμακα mel. Η κλίμακα mel είναι μια μη γραμμική κλίμακα συχνοτήτων που προσεγγίζει περισσότερο την ανθρώπινη αντίληψη του ήχου που χωρίζεται σε μια σειρά τριγωνικών επικαλυπτόμενων ζωνών συχνοτήτων.

Αφού εξαγάγουμε τα χρήσιμα χαρακτηριστικά από το ακατέργαστο σήμα χρησιμοποιώντας την επεξεργασία σήματος, θα εκπαιδεύσουμε το μοντέλο χρησιμοποιώντας ένα μπλοκ εκμάθησης (Learning block). Επιλέγοντας το μπλοκ εκμάθησης «Ταξινόμησης» (Classification) το νευρωνικού δικτύου θα λάβει κάποια δεδομένα εισόδου και θα εξάγει ένα σκορ πιθανότητας που δείχνει πόσο πιθανό είναι τα δεδομένα εισόδου να ανήκουν σε μια συγκεκριμένη κατηγορία.



Εικόνα 24: Καρτέλες ρύθμισης χαρακτηριστικών

Τέλος αποθηκεύουμε όλα νέα χαρακτηριστικά που έχουμε δώσει στο μοντέλο πριν προχωρήσουμε στην επόμενη φάση. Η επόμενη καρτέλα που βλέπουμε αποτελείται από τις παραμέτρους που το προεκπαιδευμένο μοντέλο θα χρησιμοποιεί. Επιλέγοντας διαδοχικά «Αποθήκευση παραμέτρων» (Save parameters) και «Δημιουργία χαρακτηριστικών» (Generate features) έχουμε μια εικόνα για το πως μοιάζουν τα δεδομένα εκπαίδευσης.



Εικόνα 25: Χωρικά ταξινομημένα δεδομένα

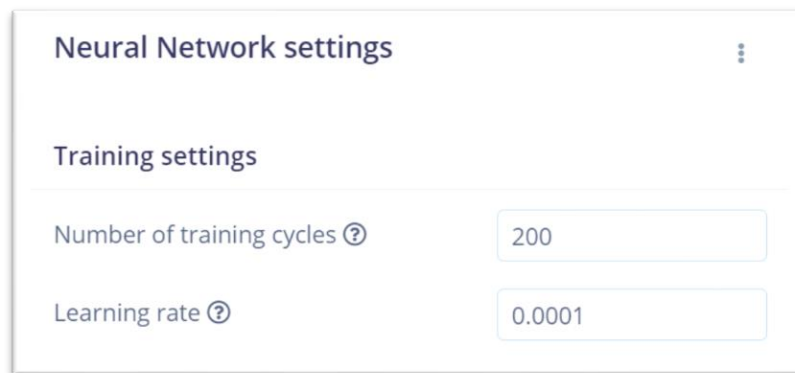
3.4 Εκπαίδευση Μοντέλου

Επόμενη και τελευταία φάση είναι η εκπαίδευση του μοντέλου. Για την διαδικασία αυτή συμπληρώνεται η καρτέλα «ταξινομητής» (Classifier). Σε αυτή την καρτέλα, θα ρυθμιστεί το μοντέλο και η διαδικασία εκπαίδευσης και αφού εφαρμοστούν στα δεδομένα και η εκπαίδευση τελειώσει θα έχω μια επισκόπηση των επιδόσεων του μοντέλου.

3.4.1 Ρύθμιση παραμέτρων μοντέλου

Στην συνέχεια ρυθμίζονται ο αριθμός κύκλων εκπαίδευσης και ο συντελεστής μάθησης του μοντέλου. Στο συγκεκριμένο μοντέλο εφαρμόστηκαν 200 κύκλοι

εκπαίδευσης και ο συντελεστής μάθησης είναι 0,0001, τα οποία μετά από διάφορες δοκιμές μου έδωσαν την καλύτερη απόδοση.



The image shows a screenshot of a 'Neural Network settings' dialog box. Under the 'Training settings' section, there are two input fields: 'Number of training cycles' with the value '200' and 'Learning rate' with the value '0.0001'. Each field has a question mark icon to its left.

Εικόνα 26: Ρυθμίσεις νευρωνικού

Αριθμός κύκλων εκπαίδευσης : ένας κύκλος (Epoch) εκπαίδευσης θεωρείται η μία φορά που ο αλγόριθμος εκπαίδευσης πραγματοποιεί ένα πλήρες πέρασμα σε όλα τα δεδομένα εκπαίδευσης με διάδοση και ενημερώνει τις παραμέτρους του μοντέλου καθώς προχωρά. Κατά τη διάρκεια κάθε epoch, το μοντέλο υπολογίζει την απόδοσή του (όπως το σφάλμα ή την ακρίβεια) στα δεδομένα εκπαίδευσης και ενημερώνει τα βάρη του με σκοπό τη βελτίωση της απόδοσής του. Συνήθως, η εκπαίδευση ενός μοντέλου περιλαμβάνει πολλές επαναλήψεις των epochs, ώστε το μοντέλο να μπορεί να μάθει από τα δεδομένα και να βελτιώσει την απόδοσή του με τον χρόνο.

Η επιλογή του αριθμού των epochs είναι μια κύρια παράμετρος που πρέπει να ρυθμιστεί κατάλληλα κατά την εκπαίδευση ενός μοντέλου. Πολλοί κύκλοι εκπαίδευσης μπορούν να οδηγήσουν σε υπερ-εκπαίδευση (overfitting), ενώ λίγοι μπορεί να μην επαρκούν για να επιτύχουν συγκεκριμένη απόδοση (underfitting). Επομένως, η επιλογή του κατάλληλου αριθμού τους είναι μια σημαντική διαδικασία κατά την εκπαίδευση του μοντέλου.

Συντελεστής μάθησης: Ο συντελεστής ή ρυθμός μάθησης ελέγχει πόσο ενημερώνονται οι εσωτερικές παράμετροι των μοντέλων σε κάθε βήμα της διαδικασίας εκπαίδευσης. Μπορείτε επίσης να το δείτε ως το πόσο γρήγορα θα μάθει το νευρωνικό δίκτυο. Ο συγκεκριμένος συντελεστής καθορίζει πόσο γρήγορα θα συγκλίνει ένα μοντέλο κατά τη διάρκεια της εκπαίδευσης και πόσο μεγάλα βήματα θα κάνει στον χώρο των παραμέτρων κατά την ανανέωσή τους.

Η καλή επιλογή του learning rate είναι κρίσιμη για την επιτυχία της εκπαίδευσης ενός μοντέλου. Ένα πολύ υψηλό learning rate μπορεί να οδηγήσει σε αστάθεια και αδυναμία σύγκλισης, ενώ ένα πολύ χαμηλό learning rate μπορεί να καθυστερήσει τη σύγκλιση του μοντέλου.

Μία ακόμα δυνατότητα που παρέχει το Edge impulse είναι η αύξηση των δεδομένων (data augmentation). Η αύξηση των δεδομένων είναι μια τεχνική που χρησιμοποιείται στη μηχανική μάθηση, κυρίως στην εκπαίδευση μοντέλων, όπως νευρικά δίκτυα, για να βελτιώσει την απόδοσή τους. Η αύξηση δεδομένων ήχου μπορεί να βελτιώσει την ανθεκτικότητα των μοντέλων σε διάφορες καταστάσεις και επιδόσεις, βοηθώντας το να γενικεύει καλύτερα και να αντιμετωπίζει το overfitting.

3.4.2 Αρχιτεκτονική μοντέλου

Ανάλογα με τον τύπο του έργου, το Edge impulse προσφέρει επιλογές μεταξύ διαφορετικών προτύπων αρχιτεκτονικής νευρωνικών δικτύων. Η αρχιτεκτονική του νευρωνικού δικτύου λαμβάνει ως είσοδο τα εξαγόμενα χαρακτηριστικά σας και τα μεταβιβάζει σε κάθε επίπεδο της. Στην περίπτωση της ταξινόμησης (classification), το τελευταίο επίπεδο που χρησιμοποιείται είναι ένα στρώμα εξόδου με 5 κατηγορίες (Output layer (5 classes)) δηλαδή μία κατηγορία για κάθε περίπτωση. Με αυτόν τον

τρόπο μετά από την εκπαίδευση για όλα τα δείγματα δίνεται μία πιθανότητα για το σε ποια κατηγορία ανήκουν.

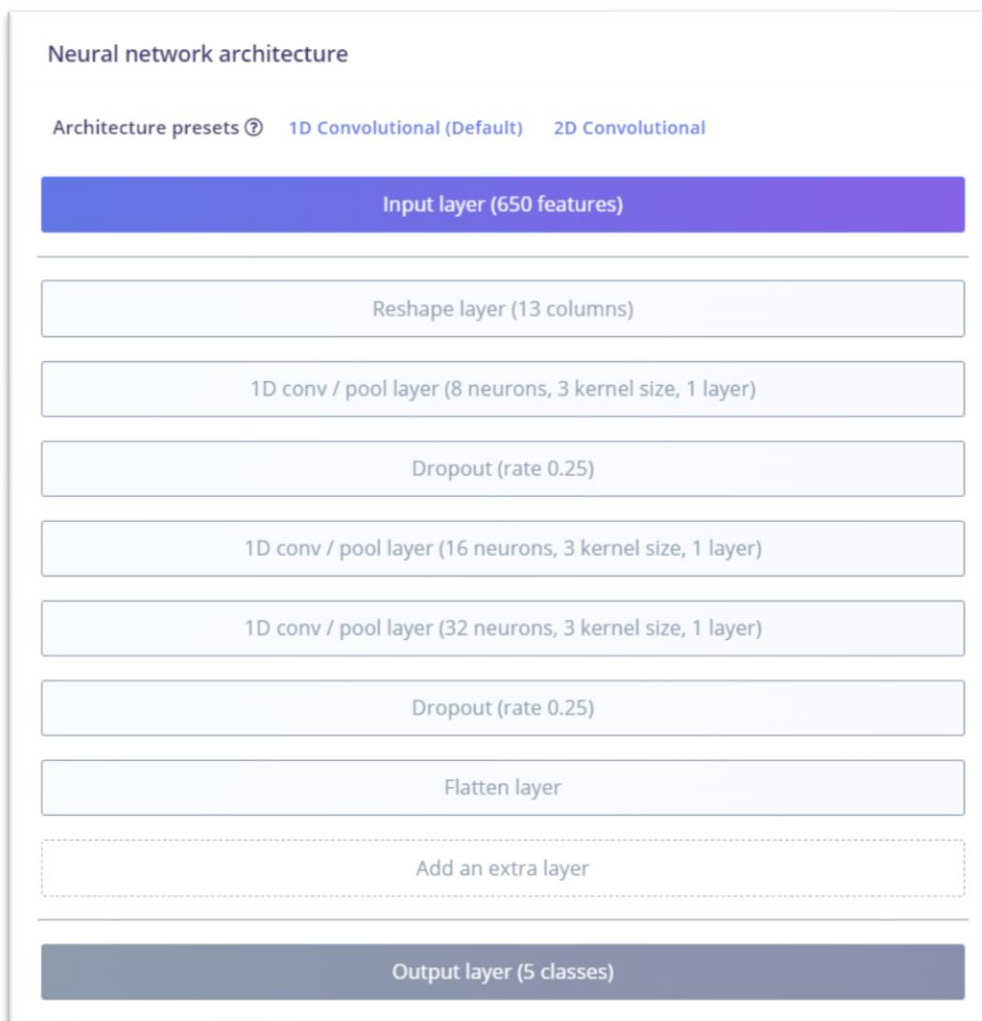
Το νευρωνικό δίκτυο που χρησιμοποιήθηκε αποτελείται από την αρχιτεκτονική του προεκπαιδευμένου μοντέλου, στο οποίο προστέθηκαν 8 επιπλέον επίπεδα.

Το πρώτο στρώμα (input (650 features)) αναφέρεται στα χαρακτηριστικά του αρχικού προεκπαιδευμένου μοντέλου που χρησιμοποιήθηκε (MFCC). Τα επόμενα επίπεδα αποτελούνται από:

- Επίπεδο ανασχηματισμού (Reshape Layer): Ο σκοπός του επιπέδου αυτού είναι να αλλάξει τη μορφή των δεδομένων χωρίς να αλλάξει την πραγματική τους πληροφορία.
 - Επίπεδα συνέλιξης (convolution) και συγκέντρωσης (pooling) με 1 διάσταση (1D): Αυτά τα επίπεδα είναι σχεδιασμένα για να κάνουν συνέλιξη σε δεδομένα με μια διάσταση. Συνδυάζοντας τη την εξαγωγή χαρακτηριστικών και τη μείωση των διαστάσεων τους για τη δημιουργία ενός συνόλου νευρώνων με σκοπό την αναπαράσταση των χαρακτηριστικών αυτών. Αυτό επιτρέπει στο δίκτυο να ανιχνεύσει χαρακτηριστικά στα δεδομένα εισόδου που έχουν χρονική συσχέτιση. Τα επίπεδα αυτά χαρακτηρίζονται από τον αριθμό των νευρώνων (neurons), το μέγεθος του πυρήνα (kernel size) και των επιπέδων στην αρχιτεκτονική του δικτύου.
- i. Οι νευρώνες λειτουργούν ως μονάδες επεξεργασίας για τα χαρακτηριστικά που εξάγονται από τη συνέλιξη και τη συγκέντρωση. Κάθε νευρώνας είναι υπεύθυνος για την εκμετάλλευση συγκεκριμένων πληροφοριών από τα δεδομένα εισόδου.

- ii. Το μέγεθος του πυρήνα ορίζει πόσα δείγματα του ήχου θα συμπεριληφθούν κάθε φορά κατά την εφαρμογή της συνέλιξης. Το μέγεθος του πυρήνα (kernel) μπορεί να είναι διαφορετικό ανάλογα με τις απαιτήσεις της εφαρμογής σας. Για παράδειγμα, μεγαλύτερα παράθυρα μπορεί να αντλήσουν περισσότερη χρονική πληροφορία από τον ήχο, ενώ μικρότερα παράθυρα μπορεί να επιτρέψουν την ανίχνευση πιο λεπτών λεπτομερειών.
- Επίπεδο εγκατάλειψης (Dropout): Το επίπεδο εγκατάλειψης είναι μια τεχνική που χρησιμοποιείται κατά την εκπαίδευση νευρικών δικτύων για τη μείωση του overfitting. Η χρήση της τεχνικής Dropout βοηθά στη γενίκευση του δικτύου, καθώς επιτρέπει σε κάθε νευρώνα να μάθει ανεξάρτητα χαρακτηριστικά από τα δεδομένα εκπαίδευσης.
- Επίπεδο εξομάλυνσης (Flatten layer): Το επίπεδο εξομάλυνσης μετατρέπει την έξοδο του προηγούμενου επιπέδου σε ένα διάνυσμα 1D που μπορεί να τροφοδοτηθεί σε ένα πλήρως συνδεδεμένο επίπεδο.

Αναλυτικότερα φαίνεται πως έχει διαμορφωθεί το νευρωνικό δίκτυο έπειτα από πολλές προσπάθειες. Έτσι αφού όλες οι παράμετροι έχουν διαμορφωθεί όπως θα έπρεπε, το μοντέλο είναι έτοιμο να εκπαιδευτεί.



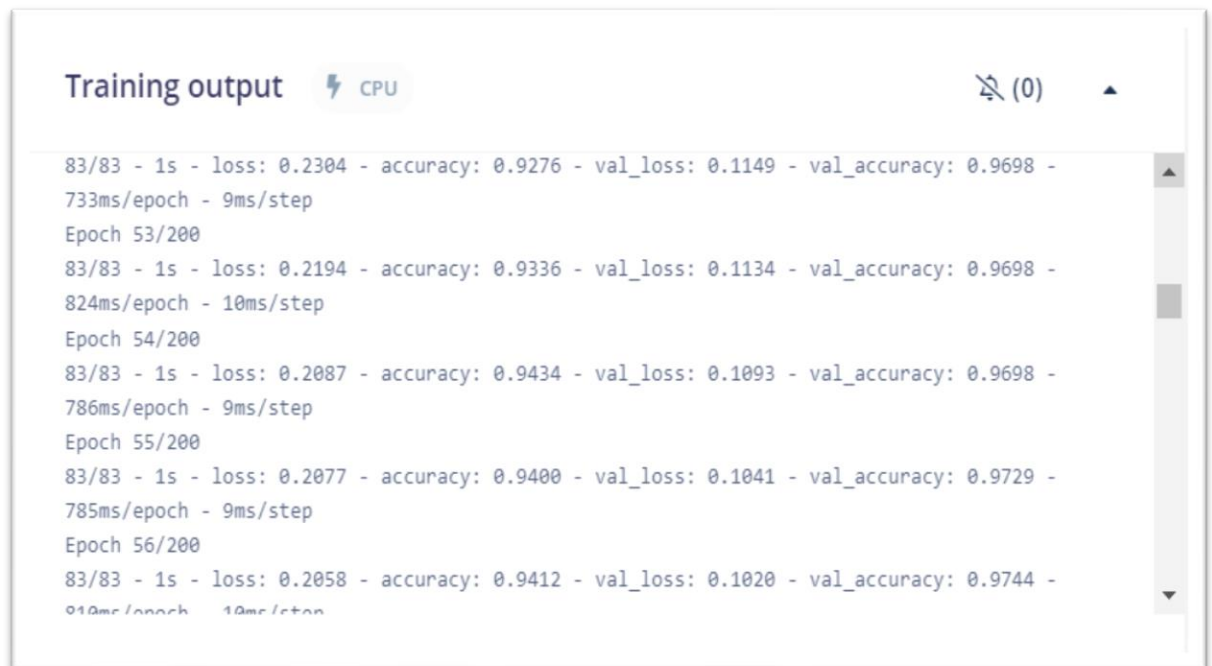
Εικόνα 27: Αρχιτεκτονική μοντέλου

3.5 Έλεγχος απόδοσης εκπαίδευσης

Πατώντας το «Εκκίνησης Εκπαίδευσης» (Start Training) μπαίνει σε εφαρμογή η διαδικασία της εκπαίδευσης του μοντέλου. Κατά τη διάρκεια της, σε ένα νέο

παράθυρο στην δεξιά μεριά της πλατφόρμας εμφανίζεται η εξέλιξη των τιμών της εκπαίδευσης και της απώλειας σε κάθε επανάληψη.

Η σχέση μεταξύ των επαναλήψεων μας δίνει πολλές και σημαντικές πληροφορίες για το πως ανταποκρίνεται το μοντέλο, με τα δεδομένα. Μελετώντας αυτή τη σχέση, μπορεί να παρατηρηθούν αστοχίες ή λάθη που προκύπτουν κατά τη διαδικασία της εκπαίδευσης. Έτσι γίνεται ευκολότερα αντιληπτό αν πρέπει να διορθωθεί κάποια ρύθμιση.



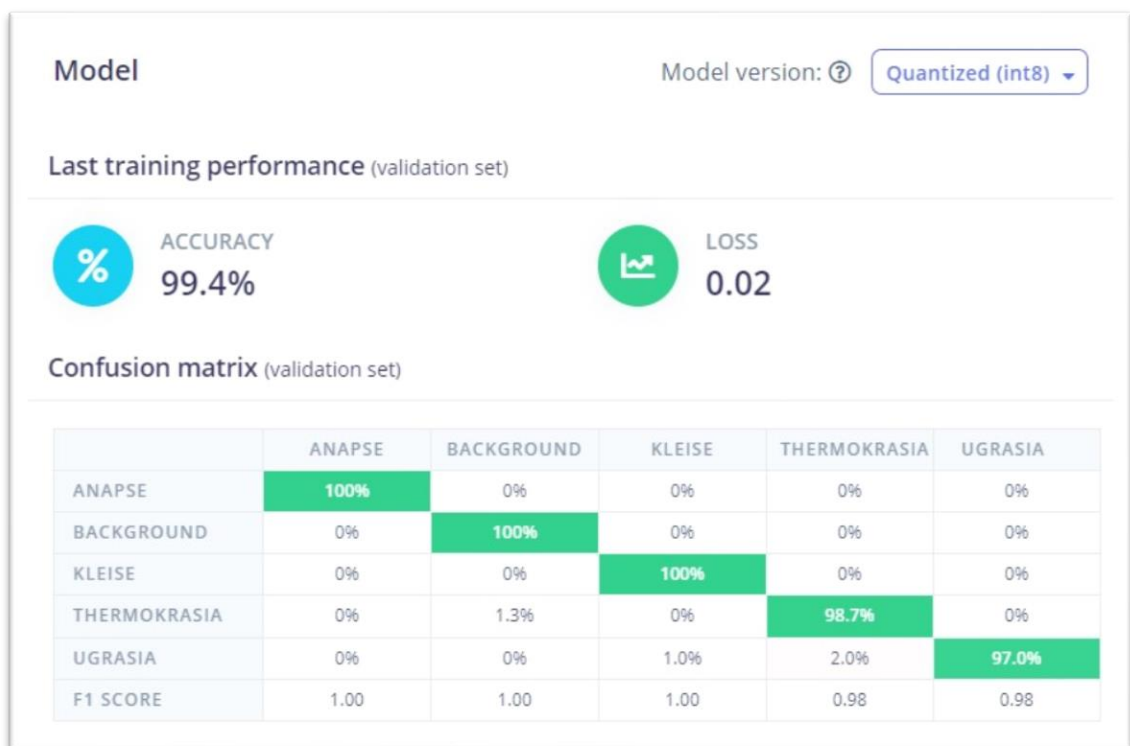
```
Training output CPU (0) ▲
83/83 - 1s - loss: 0.2304 - accuracy: 0.9276 - val_loss: 0.1149 - val_accuracy: 0.9698 -
733ms/epoch - 9ms/step
Epoch 53/200
83/83 - 1s - loss: 0.2194 - accuracy: 0.9336 - val_loss: 0.1134 - val_accuracy: 0.9698 -
824ms/epoch - 10ms/step
Epoch 54/200
83/83 - 1s - loss: 0.2087 - accuracy: 0.9434 - val_loss: 0.1093 - val_accuracy: 0.9698 -
786ms/epoch - 9ms/step
Epoch 55/200
83/83 - 1s - loss: 0.2077 - accuracy: 0.9400 - val_loss: 0.1041 - val_accuracy: 0.9729 -
785ms/epoch - 9ms/step
Epoch 56/200
83/83 - 1s - loss: 0.2058 - accuracy: 0.9412 - val_loss: 0.1020 - val_accuracy: 0.9744 -
910ms/epoch - 10ms/step
```

Εικόνα 28: Εικόνα διαδικασίας εκπαίδευσης

Όταν ολοκληρωθεί η εκπαίδευση εμφανίζεται μία νέα ενότητα η οποία περιέχει ένα πίνακα σύγχυσης(Confusion Matrix) και έναν χωρικά ταξινομημένο πίνακα των δεδομένων (Features Explorer).

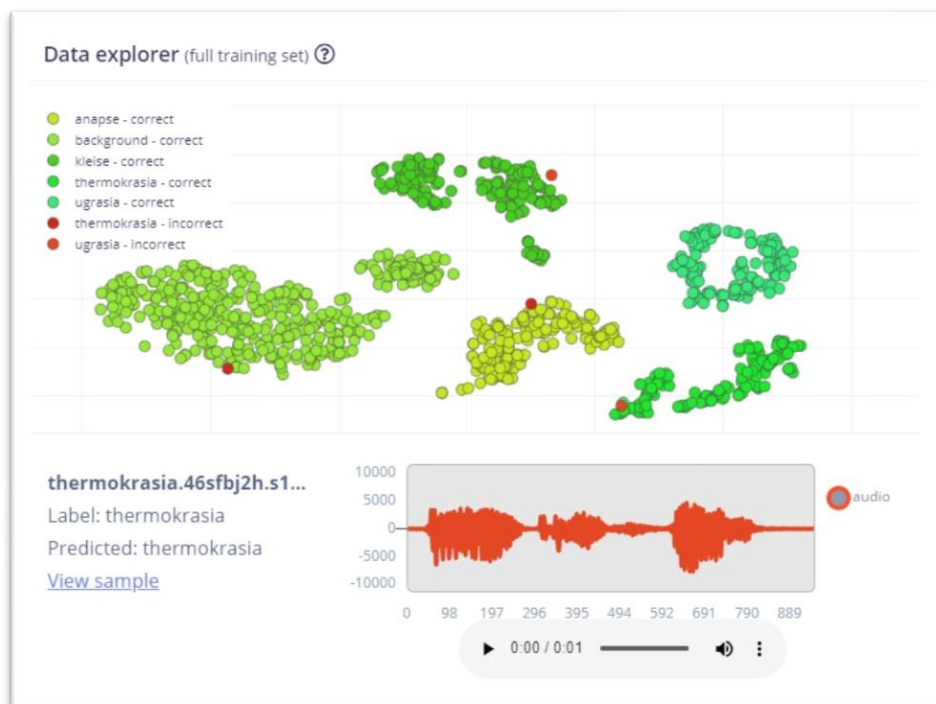
Ο πίνακας σύγχυσης είναι ένα από τα πιο χρήσιμα εργαλεία για την αξιολόγηση ενός μοντέλου. Καταγράφει όλες τις σωστές και λανθασμένες απαντήσεις ταξινόμησης

που παράγει το μοντέλο σε σύνολο δεδομένων που του έχει δοθεί. Οι ετικέτες στο πλάι αντιστοιχούν στις πραγματικές ετικέτες σε κάθε δείγμα και οι ετικέτες στην κορυφή αντιστοιχούν στις προβλεπόμενες ετικέτες από το μοντέλο. Αναλύοντας προσεγγίζεται η άποψη, αν το μοντέλο είναι ικανό να καλύψει τις ανάγκες σας ή αν πρέπει να δοκιμαστούν άλλες παράμετροι και αρχιτεκτονικές.



Εικόνα 29: Απόδοση εκπαίδευσης

Ο χωρικά ταξινομημένος πίνακας χαρακτηριστικών, οπτικοποιεί τη χωρική κατανομή των δειγμάτων εκπαίδευσης. Σε αυτόν, απεικονίζεται ποια από αυτά έχουν ταξινομηθεί σωστά και ποια όχι.



Εικόνα 30: Χωρικός πίνακας κατηγοριοποιημένων δεδομένων

Το μοντέλο μηχανικής μάθησης που αναπτύχθηκε για αυτή τη διπλωματική έχει ποσοστό ακρίβειας 99,4% και απώλεια επαλήθευσης 0,2%.

Στο τέλος της σελίδας της εκπαίδευσης αναφέρονται επίσης εκτιμήσεις σχετικά με την επίδοση που θα έχει το μοντέλο όταν ενσωματωθεί στη συσκευή. Αυτές θα αφορούν τον χρόνο εξαγωγής συμπερασμάτων, τη μέγιστη χρήση RAM και τη χρήση flash. Αυτές οι πληροφορίες θα επιβεβαιώσουν ότι το μοντέλο θα είναι σε θέση να λειτουργήσει στη συσκευή με βάση τους περιορισμούς της.



Εικόνα 31: Χαρακτηριστικά μοντέλου στη συσκευή

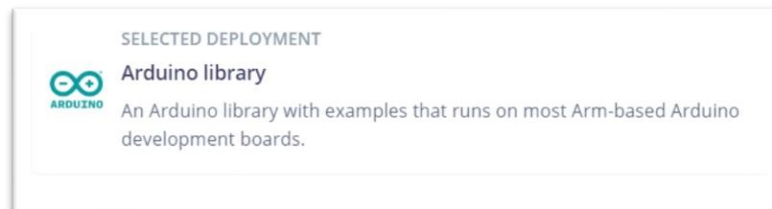
- Ο «χρόνος εξαγωγής συμπερασμάτων»(inferencing time) αναφέρεται στον χρόνο που απαιτείται για να προβλέψει ένα μοντέλο μηχανικής μάθησης το αποτέλεσμα για ένα συγκεκριμένο είσοδο δεδομένων.
- Η μνήμη Τυχαίας Προσπέλασης RAM (Random Access Memory) λειτουργεί ως προσωρινή αποθήκευση δεδομένων και κώδικα προγράμματος κατά την εκτέλεση ενός μοντέλου μηχανικής μάθησης. Αν και δεν αποθηκεύει τα δεδομένα μόνιμα όπως η μνήμη flash, είναι ουσιώδης για την άμεση πρόσβαση σε δεδομένα κατά την εκτέλεση των μοντέλων.
- Η μνήμη flash χρησιμοποιείται για την αποθήκευση των παραμέτρων του μοντέλου μηχανικής μάθησης, των δεδομένων εκπαίδευσης, του κώδικα λογισμικού και άλλων στοιχείων που απαιτούνται για τη λειτουργία του μοντέλου. Η μνήμη αυτή μπορεί να διατηρήσει τα δεδομένα ακόμη και όταν η συσκευή είναι απενεργοποιημένη.

3.6 Ανάπτυξη βιβλιοθήκης μοντέλου

Στη συνέχεια μέσα στην πλατφόρμα Edge impulse περνώντας στην σελίδα «Ανάπτυξη» (deployment) γίνεται η διαδικασία προετοιμασίας του μοντέλου σε μορφή κατάλληλη προκειμένου να ενσωματωθεί σε μία συσκευή. Κατά την πραγματοποίηση της διαδικασίας του μοντέλου για ανάπτυξη, το Edge Impulse δίνει τη δυνατότητα προσθήκης ενός ακόμα επιπέδου βελτιστοποίησης στο μοντέλο χρησιμοποιώντας τον μεταγλωττιστή EON. Αυτός, επιτρέπει την εκτέλεση των λειτουργιών των νευρωνικών δικτύων σε 25-55% λιγότερη μνήμη RAM και έως 35% λιγότερη flash, διατηρώντας την ίδια ακρίβεια, σε σύγκριση με το TensorFlow Lite για μικροελεγκτές.

Το πλεονέκτημα της χρήσης της παραδοσιακής προσέγγισης Tflite Micro είναι πολύ ευέλικτη και ευπροσάρμοστη. Το μειονέκτημα είναι ότι όλος ο κώδικας για την ετοιμασία του μοντέλου στη συσκευή είναι αρκετά βαρύς για ενσωματωμένα συστήματα. Ο μεταγλωττιστής EON επιτυγχάνει τις ενέργειες του μεταγλωττίζοντας τα νευρωνικά σας δίκτυα στον πηγαίο κώδικα C++. Αυτό διαφέρει από άλλα προγράμματα εκτέλεσης ενσωματωμένων νευρωνικών δικτύων, όπως το TensorFlow Lite για μικροελεγκτές, που διαθέτουν έναν γενικό διερμηνέα και στη συνέχεια φορτώνεται το μοντέλο κατά την εκτέλεση.

Μέσω της πλατφόρμας το μοντέλο μετατρέπεται σε έναν πλήρως βελτιστοποιημένο πηγαίο κώδικα που μπορεί να προσαρμοστεί περαιτέρω και να ενσωματωθεί στην συσκευή. Η προσαρμόσιμη βιβλιοθήκη πακετάρει όλα τα βήματα επεξεργασίας σήματος, τις διαμορφώσεις και των κώδικα μηχανικής μάθησης σε ένα ενιαίο πακέτο.



Εικόνα 32:Επιλογή τύπου βιβλιοθήκης

Αφού καθορίζεται ο τυπος βιβλιοθήκης που ταιριάζει περισσότερο στις απαιτήσεις της συσκευής πατώντας «Δημιουργία» (Built) κατεβάνει στην υπολογιστική μονάδα, το αρχείο που περιέχει όλα τα παραπάνω. Το Αρχείο αυτό, βιβλιοθήκη τύπου .h, είναι έτοιμο να εισαχθεί στον κύριο κώδικα του arduino και έπιτα να περαστεί στην συσκευή.

Κεφάλαιο 4^ο «ΑΝΑΠΤΗΞΗ ΣΥΣΤΗΜΑΤΟΣ»

Τελικό στάδιο της ανάπτυξης του συστήματος αναγνώρισης φωνητικών εντολών, αποτελεί εισαγωγή του μοντέλου στο Arduino IDE, για την ταξινόμηση αυτών σε πραγματικό χρόνο, ο σχεδιασμός και η κατασκευή του συστήματος.

4.1 Δημιουργία κώδικα μέσω Arduino IDE

Ο παρεχόμενος κώδικας Arduino είναι για μια επίδειξη εξαγωγής συμπερασμάτων Edge Impulse που χρησιμοποιεί μια πλακέτα Arduino με ένα μικρόφωνο και διάφορους αισθητήρες για την εκτέλεση συμπερασμάτων σε πραγματικό χρόνο.

Ο κώδικας έχει σχεδιαστεί για την εκτέλεση ενός μοντέλου μηχανικής μάθησης στο Arduino για την ταξινόμηση διαφόρων εισόδων ήχου και την εκτέλεση ενεργειών με βάση τα αποτελέσματα της ταξινόμησης. Ακολουθεί μια ανάλυση των βασικών συστατικών και λειτουργιών του κώδικα.

Αρχικά, όσον αφορά τις βιβλιοθήκες, ο κώδικας περιλαμβάνει βιβλιοθήκες για διάφορες λειτουργίες, συμπεριλαμβανομένης της εισόδου ήχου PDM, της οθόνης LCD και των ενδείξεων αισθητήρων, όπως επίσης και την βιβλιοθήκη του μοντέλου μηχανικής μάθησης.

Στον κώδικα, εισάγονται επίσης παγκόσμιες μεταβλητές (Global variables) δηλώνοντας διάφορα χαρακτηριστικά για την αποθήκευση ενδείξεων αισθητήρων, κατάστασης LED και άλλων σημαντικών πληροφοριών.

Περνώντας στην λειτουργία `setup()`, όπου ο κώδικα εκτελείται μία φορά, αρχικοποιείται η οθόνη LCD και ο αισθητήρας HS300x για μετρήσεις θερμοκρασίας

και υγρασίας. Επιπλέον, ρυθμίζεται η σειριακή επικοινωνία για αποσφαλμάτωση, αρχικοποιείται ο ταξινομητής Edge Impulse και προετοιμάζεται το μικρόφωνο για εγγραφή ήχου.

Στην Λειτουργία loop(), όπου ο κώδικας εκτελείται σε επανάληψη, διαβάζονται τα δεδομένα θερμοκρασίας και υγρασίας από τον αισθητήρα HS3003 και ξεκινά η εγγραφή ήχου με τη χρήση του μικροφώνου.

Στην συνέχεια, δίνεται ο καταγεγραμμένος ήχος στον ταξινομητή Edge Impulse για εξαγωγή συμπερασμάτων και εκτελούνται ενέργειες με βάση τα αποτελέσματα της ταξινόμησης για τις τέσσερις διαφορετικές κλάσεις:

1. Ανάβει μια λυχνία LED όταν ανιχνεύεται ένα συγκεκριμένο συμβάν ("άναψε").
2. Σβήνει μια λυχνία LED όταν ανιχνεύεται ένα συγκεκριμένο συμβάν ("κλείσε").
3. Εμφανίζονται πληροφορίες θερμοκρασίας σε οθόνη LCD όταν ανιχνεύεται συμβάν ("θερμοκρασία").
4. Εμφανίζονται πληροφορίες υγρασίας σε μια οθόνη LCD όταν ανιχνεύεται ένα συμβάν ("υγρασία").

Τέλος, δημιουργούνται κάποιες λειτουργίες (Functions) για την πραγματοποίηση διαδικασιών:

- `pdm_data_ready_inference_callback()` Λειτουργία:
Χειρίζεται την ανάκληση δεδομένων ήχου PDM κατά τη διάρκεια της εγγραφής ήχου.
- `microphone_inference_start()` Λειτουργία:
Αρχικοποιεί και διαμορφώνει το μικρόφωνο για εγγραφή ήχου.

- `microphone_inference_record()` Λειτουργία:
Περιμένει να γίνουν διαθέσιμα νέα δεδομένα ήχου για εξαγωγή συμπερασμάτων.
- Λειτουργία `microphone_audio_signal_get_data()`:
Ανάκτηση ακατέργαστων δεδομένων ηχητικού σήματος για εξαγωγή συμπερασμάτων.
- `microphone_inference_end()` Λειτουργία:
Σταματάει το μικρόφωνο και απελευθερώνει την εκχωρημένη μνήμη.

Οδηγίες προεπεξεργαστή:

Ορισμένες οδηγίες προεπεξεργαστή χρησιμοποιούνται για να ελέγξουν αν ο επιλεγμένος αισθητήρας (μικρόφωνο) ταιριάζει με τις απαιτήσεις του μοντέλου.

Αυτός ο κώδικας ουσιαστικά καταγράφει δεδομένα ήχου από το μικρόφωνο, τα εκτελεί μέσω του μοντέλου μηχανικής μάθησης Edge Impulse (δεν εμφανίζεται σε αυτόν τον κώδικα) και λαμβάνει συγκεκριμένες ενέργειες με βάση τα αποτελέσματα ταξινόμησης για διάφορες κλάσεις.

Διαβάζει επίσης δεδομένα θερμοκρασίας και υγρασίας από τον αισθητήρα HS3003 και τα εμφανίζει σε μια οθόνη LCD. Ανάλογα με τα αποτελέσματα της ταξινόμησης, μπορεί να ελέγχει μια λυχνία LED και να παρέχει ανατροφοδότηση μέσω της οθόνης LCD.

```

/* Defines-----*/
#define LED_pin 2
#define EI_CLASSIFIER_SLICES_PER_MODEL_WINDOW 2

/* Includes ----- */
#include <PDM.h>
#include <commander6_inferencing.h> // προσθήκη βιβλιοθηκών ήχου, μοντέλου
#include <Wire.h> // οθόνης και αισθητήρων
#include <LiquidCrystal_I2C.h>
#include <Arduino_HS300x.h>

/** Audio buffers, pointers and selectors */
typedef struct {
    signed short *buffers[2];
    unsigned char buf_select; //προσθήκη global variables
    unsigned char buf_ready;
    unsigned int buf_count;
    unsigned int n_samples;
} inference_t;

static inference_t inference;
static bool record_ready = false;
static signed short *sampleBuffer;
static bool debug_nn = false; // Set this to true to see e.g. features generated from the raw signal
static int print_results = -(EI_CLASSIFIER_SLICES_PER_MODEL_WINDOW);

LiquidCrystal_I2C lcd(0x27, 16, 2);

float old_temp = 0;
float old_hum = 0;

int detectedClassValue = 0;
int detectedClass = 0;

int anapse;
int kleise;
int thermokrasia;
int background;
int ugrasia;

/**
 * @brief Arduino setup function
 */
void setup() //λειτουργία set up
{
    lcd.init(); //αρχικοποίηση της οθόνης LCD
    //lcd.backlight();

    if (!HS300x.begin()) { //αρχικοποίηση του αισθητήρα HS300x
        while (1);
    }

    Serial.begin(115200); //αχρή λειτουργίας του Serial monitor
    // while (!Serial);
    Serial.println("Edge Impulse Inferencing Demo");

    // summary of inferencing settings (from model_metadata.h)
    ei_printf("Inferencing settings:\n");
    ei_printf("\tInterval: %.2f ms.\n", (float)EI_CLASSIFIER_INTERVAL_MS); //ρύθμιση της σειριακής επικοινωνίας για αποσφμάτωση
    ei_printf("\tFrame size: %d\n", EI_CLASSIFIER_DSP_INPUT_FRAME_SIZE);
    ei_printf("\tSample length: %d ms.\n", EI_CLASSIFIER_RAW_SAMPLE_COUNT / 16);
    ei_printf("\tNo. of classes: %d\n", sizeof(ei_classifier_inferencing_categories) /
        sizeof(ei_classifier_inferencing_categories[0]));
}

```



```

run_classifier_init();
if (microphone_inference_start(EI_CLASSIFIER_SLICE_SIZE) == false) { //Αρχειοποίηση του ταξινομητή Edge Impulse και προετοιμασία μικροφώνου
    ei_printf("ERR: Could not allocate audio buffer (size %d), this could be due to the window length of your model\r\n", EI_CLASSIFIER_RAW_SAMPLE_COUNT);
    return;
}

pinMode(LED_pin, OUTPUT); //ορισμός pin ως output
}

/**
 * @brief Arduino main function. Runs the inferencing loop.
 */
void loop() //λειτουργία loop
{
    float temperature = HS300x.readTemperature(); //Ανάγνωση δεδομένων θερμοκρασίας
    float humidity = HS300x.readHumidity(); //Ανάγνωση δεδομένων υγρασίας

    bool m = microphone_inference_record(); //Ξεκινά την εγγραφή ήχου
    if (!m) {
        ei_printf("ERR: Failed to record audio...\n");
        return;
    }

    signal_t signal;
    signal.total_length = EI_CLASSIFIER_SLICE_SIZE;
    signal.get_data = &microphone_audio_signal_get_data; //Εξαγωγή συμπερασμάτων από τον ταξινομητή
    ei_impulse_result_t result = {0};

    EI_IMPULSE_ERROR r = run_classifier_continuous(&signal, &result, debug_nn);
    if (r != EI_IMPULSE_OK) {
        ei_printf("ERR: Failed to run classifier (%d)\n", r);
        return;
    }

    anapse = result.classification[0].value*100;
    background = result.classification[1].value*100; //Απόδίδει σε κάθε εντολή μία τιμή
    kleise = result.classification[2].value*100; //και δίνει το ποσοστό στα 100 για την προβλεπόμενη
    thermokrasia = result.classification[3].value*100;
    ugrasia = result.classification[4].value*100;

    //Εκτέλεση ενεργειών ανάλογα με την κάθε εντολή

    if (anapse > 95) { //πρόβλεψη εντολής άναψε με ποσοστό 95 και μεγαλύτερο
        digitalWrite(LED_pin, HIGH); //Ανάβει το led
        delay(1000); //καθυστέρηση 1 δευτερόλεπτο
    }

    if (kleise > 95) { //πρόβλεψη τιμής κλείσε με ποσοστό 95 και μεγαλύτερο
        digitalWrite(LED_pin, LOW); //Σβήνει το led
        delay(1000); //καθυστέρηση 1 δευτερόλεπτο
    }
}

```

```

//Εμφάνιση της τιμής της θερμοκρασίας σε lcd οθόνη, μέσω του αντίστοιχου αισθητήρα
if (thermokrasia > 95) { //προβλεψη εντολής θερμοκρασίας με ποσοστό 95 και μεγαλύτερο
    if (temperature >= 28) { //ανάγνωση τιμής θερμοκρασίας // αν η τιμή της είναι 28και πάνω
        lcd.setCursor(0, 1); //θέσε τον κέρσορα την πρώτη θέση της 2ης γραμμής της οθόνης
        lcd.print("Hot!"); //εκτύπωσε"Hot!"
    }
    if ((temperature > 18) && (temperature < 28)) { // αν η τιμή της είναι απο 18 μεχρι 28
        lcd.setCursor(0, 1); //θέσε τον κέρσορα την πρώτη θέση της 2ης γραμμής της οθόνης
        lcd.print("Normal!"); //εκτύπωσε"Normal!"
    }
    if (temperature <= 18) { // αν η τιμή της είναι κάτω από 18
        lcd.setCursor(0, 1); //θέσε τον κέρσορα την πρώτη θέση της 2ης γραμμής της οθόνης
        lcd.print("Cold!"); //εκτύπωσε"Cold!"
    }
    lcd.backlight(); //ενεργοποίησε την οθόνη
    lcd.setCursor(0, 0); //θέσε τον κέρσορα την πρώτη θέση της 1ης γραμμής της οθόνης
    lcd.print("Temp = "); //εκτύπωσε "Temp = "
    lcd.print(temperature, 1); //εκτύπωσε τη θερμοκρασία δίνοντας 1 δεκαδικό ψηφίο
    lcd.print(" "); //εκτύπωσε 1 κενό
    lcd.print((char)223); //εκτύπωσε "°"
    lcd.print("C"); //εκτύπωσε "C"
    delay(4000); //καθηστέρησε 4 δευτερόλεπτα
    lcd.clear(); //καθάρισε την οθόνη
    lcd.noBacklight(); //σβήσε την οθόνη
}

//Εμφάνιση της τιμής της υγρασίας σε lcd οθόνη, μέσω του αντίστοιχου αισθητήρα
if (ugrasia > 95) { //προβλεψη εντολής υγρασίας με ποσοστό 95 και μεγαλύτερο
    if (humidity >= 55) { //ανάγνωση τιμής θερμοκρασίας // αν η τιμή της είναι 55και πάνω
        lcd.setCursor(0, 1); //θέσε τον κέρσορα την πρώτη θέση της 2ης γραμμής της οθόνης
        lcd.print("High!"); //εκτύπωσε"High!"
    }
}

if ((humidity > 35) && (humidity < 55)) { // αν η τιμή της είναι απο 35 μεχρι 55
    lcd.setCursor(0, 1); //θέσε τον κέρσορα την πρώτη θέση της 2ης γραμμής της οθόνης
    lcd.print("Normal!"); //εκτύπωσε"Normal!"
}
if (humidity <= 35) { // αν η τιμή της είναι κάτω από 35
    lcd.setCursor(0, 1); //θέσε τον κέρσορα την πρώτη θέση της 2ης γραμμής της οθόνης
    lcd.print("Low!"); //εκτύπωσε"Low!"
}
lcd.backlight(); //ενεργοποίησε την οθόνη
lcd.setCursor(0, 0); //θέσε τον κέρσορα την πρώτη θέση της 1ης γραμμής της οθόνης
lcd.print("Hum = "); //εκτύπωσε "Hum = "
lcd.print(humidity, 1); //εκτύπωσε την υγρασία δίνοντας 1 δεκαδικό ψηφίο
lcd.print("%"); //εκτύπωσε"%"
delay(4000); //καθηστέρησε 4 δευτερόλεπτα
lcd.clear(); //καθάρισε την οθόνη
lcd.noBacklight(); //σβήσε την οθόνη
}

//Εκτύπωση των αποτελεσμάτων της ταξινόμησης στην σειριακή οθόνη
if (++print_results >= (EI_CLASSIFIER_SLICES_PER_MODEL_WINDOW)) {
    // print the predictions
    ei_printf("Predictions ");
    ei_printf("(DSP: %d ms., Classification: %d ms., Anomaly: %d ms.)",
        result.timing.dsp, result.timing.classification, result.timing.anomaly);
    ei_printf("\n");
    for (size_t ix = 0; ix < EI_CLASSIFIER_LABEL_COUNT; ix++) {
        ei_printf(" %s: %.5f\n", result.classification[ix].label,
            result.classification[ix].value);
    }
}

```

```

#if EI_CLASSIFIER_HAS_ANOMALY == 1
    ei_printf("    anomaly score: %.3f\n", result.anomaly);
#endif

    print_results = 0;
}
}

/**
 * @brief      PDM buffer full callback
 *            Get data and call audio thread callback
 */

//Χειρισμός της ανάκλησης δεδομένων ήχου PDM κατά τη διάρκεια της εγγραφής ήχου
static void pdm_data_ready_inference_callback(void)
{
    int bytesAvailable = PDM.available();

    // read into the sample buffer
    int bytesRead = PDM.read((char *)&sampleBuffer[0], bytesAvailable);

    if (record_ready == true) {
        for (int i = 0; i < bytesRead; i++) {
            inference.buffers[inference.buf_select][inference.buf_count++] = sampleBuffer[i];

            if (inference.buf_count >= inference.n_samples) {
                inference.buf_select ^= 1;
                inference.buf_count = 0;
                inference.buf_ready = 1;
            }
        }
    }
}

//Αρχικοποιεί και διαμορφώνει το μικρόφωνο για εγγραφή ήχου.
static bool microphone_inference_start(uint32_t n_samples)
{
    inference.buffers[0] = (signed short *)malloc(n_samples * sizeof(signed short));

    if (inference.buffers[0] == NULL) {
        return false;
    }

    inference.buffers[1] = (signed short *)malloc(n_samples * sizeof(signed short));

    if (inference.buffers[1] == NULL) {
        free(inference.buffers[0]);
        return false;
    }

    sampleBuffer = (signed short *)malloc((n_samples >> 1) * sizeof(signed short));

    if (sampleBuffer == NULL) {
        free(inference.buffers[0]);
        free(inference.buffers[1]);
        return false;
    }

    inference.buf_select = 0;
    inference.buf_count = 0;
    inference.n_samples = n_samples;
    inference.buf_ready = 0;

    // configure the data receive callback
    PDM.onReceive(&pdm_data_ready_inference_callback);

    PDM.setBufferSize((n_samples >> 1) * sizeof(int16_t));
}

```

```

// initialize PDM with:
// - one channel (mono mode)
// - a 16 kHz sample rate
if (!PDM.begin(1, EI_CLASSIFIER_FREQUENCY)) {
    ei_printf("Failed to start PDM!");
}

// set the gain, defaults to 20
PDM.setGain(127);

record_ready = true;

return true;
}

/**
 * @brief      Wait on new data
 *
 * @return     True when finished
 */

//Περιμένει να γίνουν διαθέσιμα νέα δεδομένα ήχου για εξαγωγή συμπερασμάτων.
static bool microphone_inference_record(void)
{
    bool ret = true;

    if (inference.buf_ready == 1) {
        ei_printf(
            "Error sample buffer overrun. Decrease the number of slices per model window "
            "(EI_CLASSIFIER_SLICES_PER_MODEL_WINDOW)\n");
        ret = false;
    }

    while (inference.buf_ready == 0) {
        delay(1);
    }

    inference.buf_ready = 0;

    return ret;
}

/**
 * Get raw audio signal data
 */

//Ανάκτηση ακατέργαστων δεδομένων ηχητικού σήματος για εξαγωγή συμπερασμάτων.
static int microphone_audio_signal_get_data(size_t offset, size_t length, float *out_ptr)
{
    numpy::int16_to_float(&inference.buffer[inference.buf_select ^ 1][offset], out_ptr, length);

    return 0;
}

/**
 * @brief      Stop PDM and release buffers
 */

//Σταματάει το μικρόφωνο και απελευθερώνει την εκχωρημένη μνήμη.
static void microphone_inference_end(void)
{
    PDM.end();
    free(inference.buffer[0]);
    free(inference.buffer[1]);
    free(sampleBuffer);
}

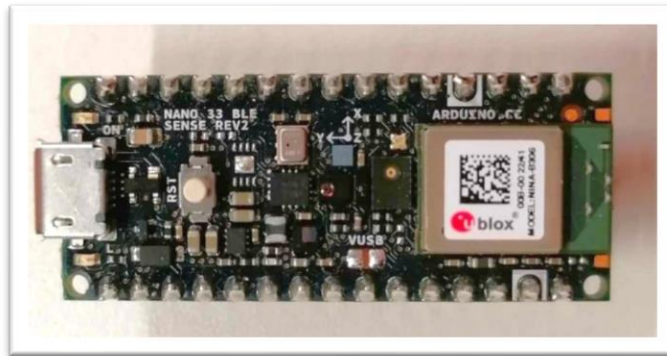
#if !defined(EI_CLASSIFIER_SENSOR) || EI_CLASSIFIER_SENSOR != EI_CLASSIFIER_SENSOR_MICROPHONE
#error "Invalid model for current sensor."
#endif

```

Εικόνα 33: Λογισμικό συσκευής

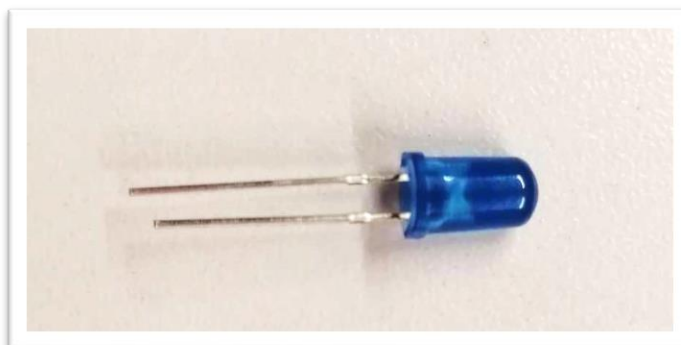
4.2 Υλικό συσκευής – Hardware

Για την σχεδίαση του υλικού (Hardware) του συστήματος, χρειάστηκε να προστεθούν τα απαραίτητα εξαρτήματα, ώστε να αλληλοεπιδρούν με τις φωνητικές εντολές και το περιβάλλον.



Εικόνα 34: Arduino Nano BLE Sense Rev 2

Στην πλακέτα Arduino Nano 33 BLE Sense, η οποία αποτελεί την βάση του συστήματος, συνδέθηκε αρχικά ένα LED (χρώματος μπλε). Το LED αυτό, λειτουργεί με την αναγνώριση των φωνητικών εντολών «άναψε» και «κλείσε», με τις οποίες ανάβει και σβήνει αντίστοιχα.



Εικόνα 35: Δίοδος εκπομπής φωτός LED

Στην άνοδο του LED συνδέθηκε μία αντίσταση ($R = 220\Omega$), ενώ το άκρο του, στην συνέχεια, τοποθετήθηκε στον ακροδέκτη (D2) της πλακέτας. Η κάθοδος του LED οδηγήθηκε στον ακροδέκτη της γείωσης της πλακέτας.

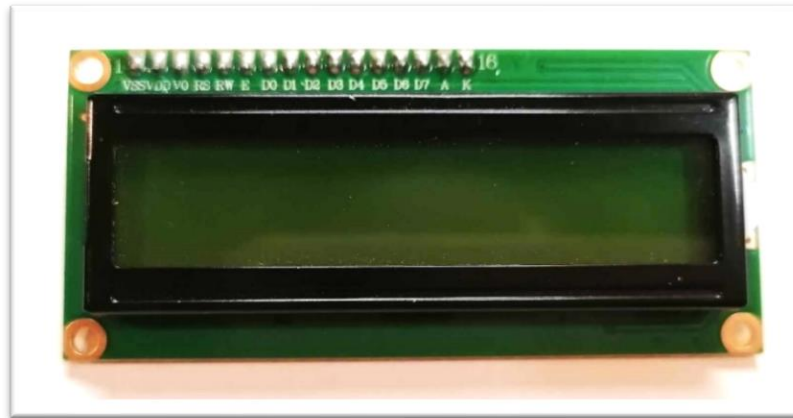


Εικόνα 36: Αντίσταση

Τέλος, σε όλο το σύστημα προστέθηκε και μία LCD (16 στήλες x 2 γραμμές) οθόνη, μέσω της οποίας αλληλοεπιδρούν οι άλλες δύο εντολές.

Με την αναγνώριση της εντολής «θερμοκρασία» η LCD ενεργοποιείται και, λαμβάνοντας την τιμή της θερμοκρασίας του χώρου, από τον κατάλληλο αισθητήρα της πλακέτας, αναγράφει τις ζητούμενες πληροφορίες στην οθόνη, σε βαθμούς Κελσίου.

Επιπλέον με την εντολή «υγρασία», με παρόμοιο τρόπο με αυτόν που προαναφέρθηκε, εμφανίζεται το ποσοστό της υγρασίας του χώρου, σε πραγματικό χρόνο.



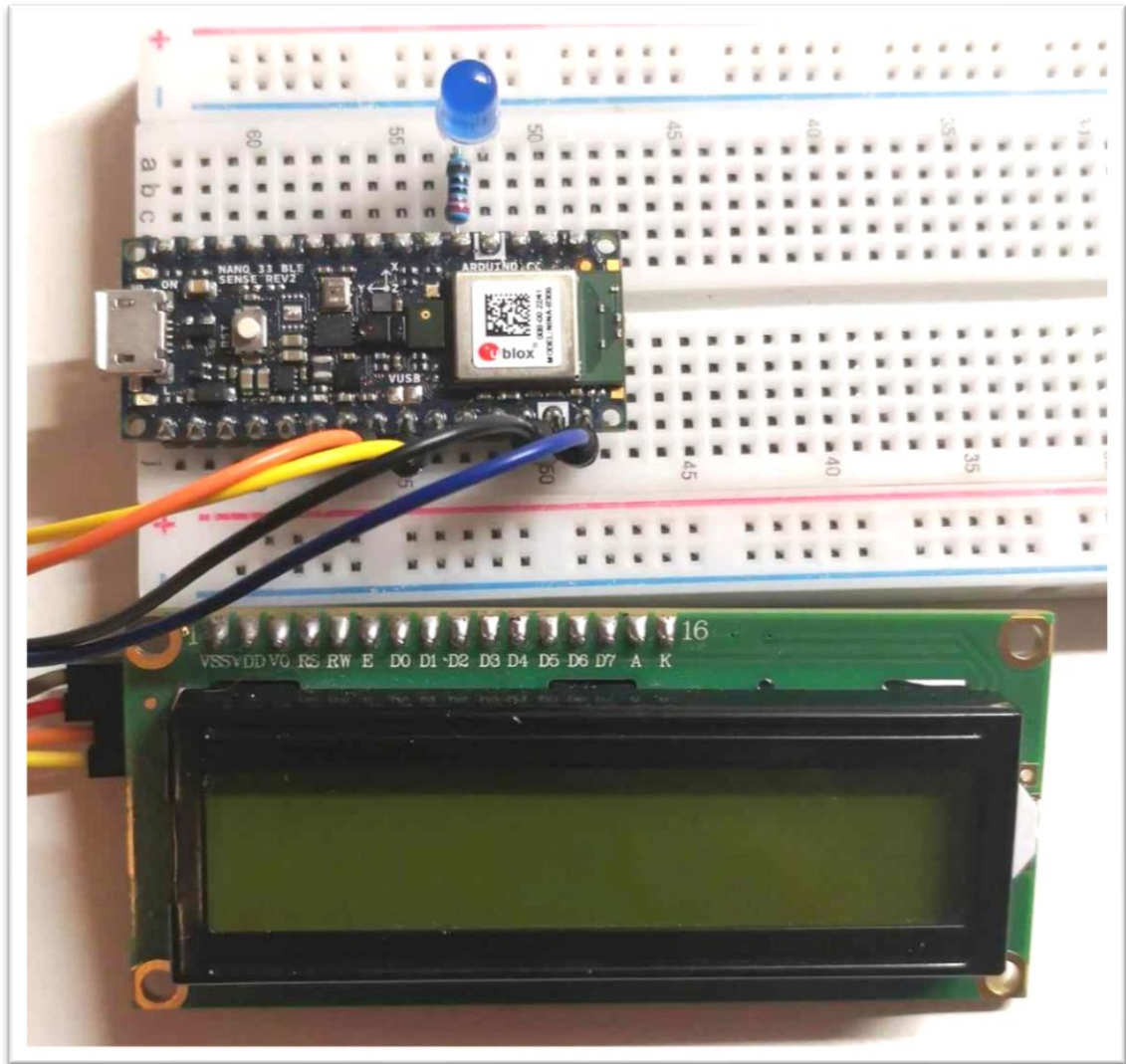
Εικόνα 37: Οθόνη LCD μπροστά όψη

Όσον αφορά την συνδεσμολογία της LCD οθόνης, η συγκεκριμένη περιλαμβάνει τέσσερις ακροδέκτες στο πίσω της μέρος. Αυτοί είναι, ο GND (Ground), ο VCC (Voltage Common Collector), ο SDA (Serial Data) και ο SCL (Serial Clock) και συνδέονται με την πλακέτα στους ακροδέκτες, GND, Vin, A4 και A5 αντίστοιχα.



Εικόνα 38: Οθόνη LCD πίσω όψη

Ολοκληρώνοντας, η τελική μορφή του συστήματος αναγνώρισης φωνητικών εντολών δίνεται παρακάτω:



Εικόνα 39: Ολοκληρωμένο σύστημα αναγνώρισης εντολών

Επίλογος

Ολοκληρώνοντας την εξερεύνηση της "Ενσωματωμένης μηχανικής μάθησης", εμβαθύνουμε στις μετασχηματιστικές δυνατότητες της τεχνητής νοημοσύνης στα ενσωματωμένα συστήματα. Από τη θεωρία στην πράξη, είδαμε πώς η τεχνητή νοημοσύνη ενισχύει τις δυνατότητες των συσκευών σε αλληλεπιδράσεις σε πραγματικό χρόνο.

Καθώς η τεχνολογία και η ΤΝ συγκλίνουν, οι συσκευές μας γίνονται πιο έξυπνες και πιο ευέλικτες. Αυτή η διπλωματική εργασία έδειξε ότι οι δυνατότητες είναι απεριόριστες, από την αναγνώριση φωνής έως τις αυτόνομες περιβαλλοντικές αντιδράσεις.

Η ενσωματωμένη μηχανική μάθηση δεν είναι απλώς ένα τεχνικό κατόρθωμα, είναι μια αλλαγή παραδείγματος προς μια πιο διαισθητική και αποτελεσματική τεχνολογία. Καθώς προχωράμε μπροστά, προβλέπουμε ένα μέλλον όπου η ενσωματωμένη μηχανική μάθηση θα συνεχίσει να διαμορφώνει την επόμενη γενιά έξυπνων συσκευών, βελτιώνοντας τις καθημερινές μας εμπειρίες.

Η παρούσα διατριβή χρησιμεύει στο να ανακαλύψουμε τις απεριόριστες ευκαιρίες που βρίσκονται μπροστά μας σε αυτόν τον δυναμικό τομέα.

Βιβλιογραφία

1. «Machine learning | artificial intelligence». *Encyclopedia Britannica*.
2. Phil Simon (18 Μαρτίου 2013). [Too Big to Ignore: The Business Case for Big Data](#). Wiley. σελ. 89. [ISBN 978-1-118-63817-0](#).
3. Ron Kohavi; Foster Provost (1998). [«Glossary of terms»](#). *Machine Learning* **30**: 271–274.
4. Machine learning and pattern recognition "can be viewed as two facets of the same field."
5. Wernick, Yang, Brankov, Yourganov and Strother, Machine Learning in Medical Imaging, *IEEE Signal Processing Magazine*, vol. 27, no. 4, July 2010, pp. 25-38
6. Mannila, Heikki (1996). «Data mining: machine learning, statistics, and databases». Int'l Conf. Scientific and Statistical Database Management. IEEE Computer Society.
7. Friedman, Jerome H. (1998). «Data Mining and Statistics: What's the connection?». *Computing Science and Statistics* **29** (1): 3–9.
8. Mitchell, T. (1997). *Machine Learning*, McGraw Hill, *Machine Learning*, McGraw Hill, p.2
9. Harnad, Stevan (2008), [«The Annotation Game: On Turing \(1950\) on Computing, Machinery, and Intelligence»](#), στο: Epstein, Robert; Peters, Grace, επιμ., *The Turing Test Sourcebook: Philosophical and Methodological Issues in the Quest for the Thinking Computer*, Kluwer
10. Mnih, Volodymyr; Kavukcuoglu, Koray; Silver, David; Graves, Alex; Antonoglou, Ioannis; Wierstra, Daan; Riedmiller, Martin (2013-12-19). [«Playing Atari with Deep Reinforcement Learning»](#). *arXiv:1312.5602 [cs.LG]*.
11. Vaswani, Ashish; Shazeer, Noam; Parmar, Niki; Uszkoreit, Jakob; Jones, Llion; Gomez, Aidan N.; Kaiser, Lukasz; Polosukhin, Illia (2017-12-05). [«Attention Is All You Need»](#). *arXiv:1706.03762 [cs]*.

12. [Τεχνητή Νοημοσύνη - Δ' Έκδοση, 2020, Εκδόσεις Πανεπιστημίου Μακεδονίας, Βλαχάβας, Κεφαλάς, Βασιλειάδης, Κόκκορας, Σακελλαρίου, ISBN 978-618-5196-44-8](#)
13. [Andreas Kaplan](#), 2022, [Artificial Intelligence, Business and Civilization - Our Fate Made in Machines](#), Routledge
14. West, Jeremy; Ventura, Dan; Warnick, Sean (2007). "[Spring Research Presentation: A Theoretical Foundation for Inductive Transfer](#)". Brigham Young University, College of Physical and Mathematical Sciences. Archived from [the original](#) on 2007-08-01. Retrieved 2007-08-05.
15. George Karimpanal, Thommen; Bouffanais, Roland (2019). "Self-organizing maps for storage and transfer of knowledge in reinforcement learning". *Adaptive Behavior*. **27** (2): 111–126. [arXiv:1811.08318](#). [doi:10.1177/1059712318818568](#). [ISSN 1059-7123](#). [S2CID 53774629](#).
16. Stevo. Bozinovski and Ante Fulgosi (1976). "The influence of pattern similarity and transfer learning upon the training of a base perceptron B2." (original in Croatian) Proceedings of Symposium Informatica 3-121-5, Bled.
17. Stevo Bozinovski (2020) "[Reminder of the first paper on transfer learning in neural networks, 1976](#)". Informatica 44: 291–302.
18. S. Bozinovski (1981). "Teaching space: A representation concept for adaptive pattern classification." COINS Technical Report, the University of Massachusetts at Amherst, No 81-28 [available online: UM-CS-1981-028.pdf]
19. Arnold, Ken. *Embedded controller hardware design*, Newnes, 2001. [ISBN 1-878707-52-3](#).
20. Gadre, Dhananjay V. [Programming and customizing the AVR microcontroller](#), McGraw-Hill, 2000. [ISBN 0-07-134666-X](#).
21. "[Arduino FAQ – With David Cuartielles](#)". [Malmö University](#). April 5, 2013. Archived from [the original](#) on 2017-09-06. Retrieved 2014-03-24.
22. "[Business Entity Summary for Arduino LLC](#)". *Mass.gov. State of Massachusetts*.

23. Allan, Alasdair (6 March 2015). "[Arduino Wars: Group Splits, Competing Products Revealed?](#)". *makezine.com*. Maker Media, Inc. Retrieved 21 April 2015.
24. Banzi, Massimo (19 March 2015). "[Massimo Banzi: Fighting for Arduino](#)". *makezine.com*. Maker Media, Inc. Retrieved 21 April 2015.
25. <https://docs.arduino.cc/hardware/nano-33-ble-sense?queryID=undefined>
26. <https://edgeimpulse.com/>
27. <https://docs.edgeimpulse.com/docs/>
28. <https://www.sap.com/greece/products/artificial-intelligence/what-is-machine-learning.html>
29. <http://www.britannica.com/EBchecked/topic/1116194/machine-learning>
30. [«Machine Learning: What it is and why it matters»](#)