



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ - ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Πρόγραμμα Μεταπτυχιακών Σπουδών
«Προηγμένα Συστήματα Πληροφορικής - Ανάπτυξη Λογισμικού και
Τεχνητής Νοημοσύνης»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Αναγνώριση συναισθήματος σε πραγματικό χρόνο και analytics, σε εφαρμογή mobile learning Real-time emotion recognition and analytics in a mobile learning application
Όνοματεπώνυμο Φοιτητή	Αρβανίτης Χαράλαμπος
Πατρώνυμο	Παναγιώτης
Αριθμός Μητρώου	ΜΠΣΠ 20003
Επιβλέπων	Ευθύμιος Αλέπης, Αναπληρωτής Καθηγητής

Ημερομηνία Παράδοσης **Νοέμβριος 2022**

Τριμελής Εξεταστική Επιτροπή

Αλέπης Ευθύμιος
Αναπληρωτής Καθηγητής

Βίρβου Μαρία
Καθηγήτρια

Ευάγγελος Σακκόπουλος
Αναπληρωτής Καθηγητής

“Machine intelligence is the last invention that humanity will ever need to make.”

Nick Bostrom

Περίληψη

Η παρούσα διπλωματική εργασία εκπονήθηκε στα πλαίσια του Προγράμματος Μεταπτυχιακών Σπουδών «Προηγμένα Συστήματα Πληροφορικής - Ανάπτυξη Λογισμικού και Τεχνητής Νοημοσύνης», και έχει ως στόχο την υλοποίηση μίας android εφαρμογής mobile learning, μέσω της οποίας γίνεται η αναγνώριση του τρέχοντος συναισθήματος του χρήστη και συλλέγονται δεδομένα κατά τη διάρκεια της μάθησης. Τα δεδομένα χρησιμοποιούνται είτε αυτούσια, είτε επεξεργάζονται μέσω μοντέλων μηχανικής μάθησης για την εξαγωγή συμπερασμάτων. Σκοπός είναι η παροχή πληροφορίας μέσω analytics στον εκπαιδευτικό, ώστε να μπορεί να εκτιμήσει την απόδοση του κάθε μαθητή, να αξιολογήσει το εκπαιδευτικό περιεχόμενο που έχει δημιουργήσει, καθώς και την μέθοδο διδασκαλίας του. Παρουσιάζονται αρκετά στατιστικά στοιχεία ως παραδείγματα της πληροφορίας που μπορεί να εξαχθεί ατομικά για κάποιον χρήστη ή μαζικά για πολλούς. Τα Analytics διατίθενται μέσω δύο jupyter notebook που δημιουργήσαμε στο Google Colab. Με αυτό τον τρόπο παρέχεται στον εκπαιδευτικό η πρόσβαση στην πληροφορία μαζί με τη δυνατότητα για εύκολη επεξεργασία. Τα analytics που παρουσιάζουμε αποτελούν παραδείγματα από dummy data που εισήχθησαν ενδεικτικά και δεν αποτελούν πραγματικά δεδομένα. Εξάλλου σκοπός είναι να παρουσιαστούν οι διαφορετικές πληροφορίες που μπορούν να εξαχθούν και όχι να τις αξιοποιήσουμε για να βγάλουμε συμπεράσματα για την μάθηση μέσω κινητού. Τέλος πρέπει να αναφέρουμε έναν ακόμα στόχο της συγκεκριμένης διπλωματικής, ο οποίος είναι να δείξει τον τρόπο και την ευκολία με την οποία ένα κινητό τηλέφωνο μπορεί να συλλέξει δεδομένα και να τα αναλύσει σε πραγματικό χρόνο. Αυτό γίνεται εφικτό μέσω των πολλών διαφορετικών API που παρέχονται ελεύθερα στον προγραμματιστή από την Google.

Abstract

This master's thesis of the postgraduate program "Advanced Informatics and Computing Systems - Software Development and Artificial Intelligence" aims to implement a mobile learning application that could recognize the user's current emotion and collect various other data during the learning process. These data are either used as is or processed through machine learning models to draw conclusions. The main objective is to provide information through analytics to the instructor so that he can estimate the performance of each student, evaluate the educational content he has created, as well as his teaching method. Several statistics are presented as examples of the information that can be extracted for one user individually or for a group of students. The Analytics are provided through two jupyter notebooks that we created in Google Colab. This way the instructor is able to access and edit the information with convenience. The analytics presented in chapter 4, are based on dummy data as examples and not on actual data. After all, the purpose is to present the different information that can be extracted and not to use it in order to draw conclusions regarding mobile learning. Finally, we must mention another objective of this thesis, which is to show how easily a mobile phone can collect data and analyze it in real-time. This can be done through the various APIs, provided freely to the developer by Google.

Ευχαριστίες

Αρχικά, θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου, κ. Ευθύμιο Αλέπη, για την εμπιστοσύνη του και επειδή ήταν πάντα πρόθυμος να με βοηθήσει κάθε φορά που χρειαζόμουν την καθοδήγηση και τις συμβουλές του. Η συνεισφορά του ήταν καθοριστική όχι μόνο για την ολοκλήρωση της διπλωματικής μου διατριβής, αλλά και για τις γνώσεις που μου μετέδωσε κατά τη διάρκεια των μεταπτυχιακών μου σπουδών.

Ήταν μια πολύ έντονη περίοδος για μένα, όχι μόνο στον επιστημονικό τομέα, αλλά και σε προσωπικό επίπεδο. Έτσι, θα ήθελα να εκφράσω την ευγνωμοσύνη μου στους ανθρώπους που με στήριζαν και με βοήθησαν αυτήν την περίοδο.

Θα ήθελα να ευχαριστήσω την γυναίκα μου και τους φίλους μου που με στήριζαν και με ενθάρρυναν καθ' όλη τη διάρκεια αυτής της δύσκολης χρονιάς. Η συμβολή τους ήταν πολύτιμη και ουσιαστική.

Τέλος θέλω να εκφράσω ένα τεράστιο ευχαριστώ στους γονείς μου, που με παρακίνησαν και για την εμπιστοσύνη που μου έδειξαν κατά την διάρκεια των σπουδών μου. Το σημαντικότερο όμως είναι ότι συνέβαλαν ώστε να γίνω ένας σωστός άνθρωπος και αυτό είναι κάτι που δεν μαθαίνεται αλλά μεταδίδεται.

Περιεχόμενα

1	Εισαγωγή	6
1.1	Περιγραφή του Προβλήματος	6
1.2	Σκοπός - Συνεισφορά της Διπλωματικής Εργασίας	6
2	Θεωρητικό υπόβαθρο	8
2.1	Συναίσθημα	8
2.1.1	Διαχωρισμός συναισθημάτων	8
2.1.2	Εκφράσεις έναντι συναισθημάτων	10
2.2	Τεχνητή Νοημοσύνη	11
2.2.1	Τύποι Τεχνητής Νοημοσύνης με βάση την λειτουργία τους	12
2.2.2	Τύποι Τεχνητής Νοημοσύνης με βάση την μάθηση	15
2.2.3	Περιοχές της τεχνητής νοημοσύνης	15
2.2.4	Machine Learning	16
2.2.5	Deep Learning	18
2.2.6	Νευρωνικά Δίκτυα	18
2.2.7	Αναγνώριση Προτύπων	23
2.3	Εντοπισμός και Αναγνώριση Αντικειμένων	24
2.3.1	Εντοπισμός Αντικειμένων (Object Detection)	24
2.3.2	Αναγνώριση Αντικειμένων (Object Recognition)	24
2.3.3	Τεχνικές αναγνώρισης αντικειμένων	25
2.3.4	Αναγνώριση αντικειμένων με χρήση τεχνικών μηχανικής μάθησης	25
2.3.5	Αναγνώριση αντικειμένων με χρήση τεχνικών βαθιάς μάθησης	25
2.4	Ανίχνευση Προσώπου	26
2.4.1	Single Shot Detection	26
2.4.2	BlazeFace	27
2.5	Αναγνώριση Προσώπου	28
2.5.1	FaceNet	29
2.5.2	MobileNets	29
2.5.3	MobileFaceNet	29
2.6	Μάθηση μέσω Κινητών Συσκευών	30
2.6.1	Η εξέλιξη της κινητής μάθησης	30
2.7	API	31
2.7.1	Google Machine Learning Kit	31
2.7.2	Google Activity Recognition	31
2.7.3	Google Fused Location Provider	32
2.7.4	Firebase	32
2.7.5	TensorFlow lite	33
2.7.6	Youtube Data API	33
2.7.7	CameraX	33
3	Μεθοδολογία	35
3.1	Αναγνώριση συναισθήματος	35
3.1.1	Αλγόριθμος αναγνώρισης συναισθήματος	35
3.1.2	Εντοπισμός προσώπου	36
3.1.3	Αναγνώριση συναισθήματος	39
3.2	Αναγνώριση σκηνής	43
3.2.1	Αλγόριθμος αναγνώρισης σκηνής	43

3.2.2 Μοντέλο αναγνώρισης σκηνής	44
3.3 Αναγνώριση δραστηριότητας	45
3.4 Εντοπισμός τοποθεσίας	45
3.5 Επισήμανση Στοιχείων Εικόνας και Εντοπισμός Αντικειμένων	46
3.5.1 Επισήμανση στοιχείων εικόνας (Image Labeling)	46
3.5.2 Αλγόριθμος επισήμανσης εικόνων	47
3.5.3 Εντοπισμός αντικειμένων (Object Detection)	47
3.5.4 Αλγόριθμος εντοπισμού αντικειμένων	48
3.6 Αλγόριθμος Συλλογής Δεδομένων	48
3.6.1 Αισθητήρες Android	49
3.6.2 Εξαγωγή δεδομένων από Android Camera	50
3.6.3 Φιλτράρισμα και επεξεργασία δεδομένων	51
3.7 Ανάπτυξη Εφαρμογών	52
3.8 Analytics και Metrics	53
3.8.1 User Analytics	54
3.8.2 Mass Analytics	54
3.9 Διαδικασία Ανάπτυξης Λογισμικού	54
4 Υλοποίηση	56
4.1 Εφαρμογή Android	56
4.1.1 Οθόνη Splash	56
4.1.2 Οθόνη LogIn	57
4.1.3 Οθόνη Register	58
4.1.4 Κεντρική οθόνη Emoricon	59
4.1.5 Οθόνη Προβολής Μαθήματος	60
4.1.6 Οθόνη Live Capture	61
4.1.7 Οθόνη Test Models	62
4.1.8 Εισαγωγή Μαθημάτων	62
4.1.9 Menu Διαχείρισης	65
4.1.10 Οθόνη Ρυθμίσεων	65
4.1.11 Οθόνη Logger	66
4.2 Εφαρμογή διαχείρισης Analytics	67
4.2.1 UserAnalytics.ipynb	67
4.2.2 MassAnalytics.ipynb	71
5 Συζήτηση	74
5.1 Προσωπικά δεδομένα	74
6 Συμπεράσματα	75
A' Πίνακας ορολογίας	77
B' Ακρωνύμια και συντομογραφίες	78

Κατάλογος Σχημάτων

2.1	Ο τροχός των συναισθημάτων	9
2.2	AI-ML-DL [1]	12
2.3	Κλάδοι τεχνητής νοημοσύνης [2]	17
2.4	Κατηγορίες μηχανικής μάθησης [3]	18
2.5	Βιολογικός Νευρώνας	19
2.6	Δομή και λειτουργία τεχνητού νευρώνα	20
2.7	Σιγμοειδής συνάρτηση	20
2.8	Αρχιτεκτονική νευρωνικού δικτύου ενός επιπέδου	21
2.9	Αρχιτεκτονική πολυεπίπεδου νευρωνικού δικτύου	22
2.10	Απεικόνιση της λειτουργίας συνέλιξης 2D [4]	23
2.11	Ανίχνευση ακμών με συνέλιξη [4]	23
2.12	Σύγκριση καθυστέρησης μεταξύ MobileNetV2 και BlazeFace	28
2.13	Απόκριση MobileNetV2 σε διαφορετικές κινητές συσκευές	28
3.1	Αλγόριθμος εντοπισμού συναισθήματος	36
3.2	Δυνατότητες μοντέλου αναγνώρισης προσώπου ML Kit	37
3.3	Παράδειγμα bounding box και face contours κατά τον εντοπισμό προσώπου	38
3.4	Επιλογές διαμόρφωσης για την ανίχνευση προσώπου σε πραγματικό χρόνο	38
3.5	Ετικέτες εκπαίδευσης μοντέλου συναισθήματος	39
3.6	Κατανομή συναισθημάτων στο σύνολο δεδομένων [5]	40
3.7	Δομή νευρωνικού δικτύου [5]	41
3.8	Παράμετροι εκπαίδευσης νευρωνικού δικτύου [5]	42
3.9	Κανονικοποιημένος πίνακας σύγχυσης [5]	42
3.10	Metrics του εκπαιδευμένου μοντέλου [5]	42
3.11	Ετικέτες εκπαίδευσης μοντέλου αναγνώρισης σκηνής [6]	44
3.12	Κατανομή δεδομένων εκπαίδευσης μοντέλου αναγνώρισης σκηνής [6]	44
3.13	Κατανομή δεδομένων εκπαίδευσης μοντέλου αναγνώρισης σκηνής [6]	45
3.14	Παράδειγμα αναγνωρισμένου αντικειμένου [7]	48
3.15	Αλγόριθμος συλλογής δεδομένων	49
3.16	Διάγραμμα ακολουθίας του αλγορίθμου συλλογής δεδομένων	51
4.1	Οθόνη Splash	57
4.2	Οθόνη Splash	58
4.3	Οθόνες δημιουργίας λογαριασμού	59
4.4	Κεντρική οθόνη εφαρμογής	60
4.5	Οθόνη Προβολής Μαθήματος	61
4.6	Οθόνη Live Capture	61
4.7	Οθόνη δοκιμαστικής λειτουργίας μοντέλων	62
4.8	Εισαγωγή μαθημάτων από Youtube	63
4.9	Εισαγωγή Youtube ιδεο από Chrome Web Browser	63
4.10	Manifest file για την δραστηριότητα προσθήκης βίντεο	64
4.11	Youtube Embedded URL	64
4.12	addVideoToDatabase method	64
4.13	addVideoToDatabase method	65
4.14	Constants class	66

4.15	Οθόνη Logger	67
4.16	Παράδειγμα από pandas dataframe αφού έχει εισαχθεί το json από την firebase	68
4.17	Πίτα διαφορετικών συναισθημάτων που ανιχνεύθηκαν κατά τη διαδικασία μάθησης ενός συγκεκριμένου χρήστη	68
4.18	Πίτες που εκφράζουν σε τι ποσοστό χαμογελούσε ο χρήστης κατά τη διάρκεια της μάθησης και σε τι ποσοστό κοιμόταν αντίστοιχα	69
4.19	Παράδειγμα εκτέλεσης κώδικα για την δημιουργία του παρακάτω διαγράμματος	69
4.20	Διαφορετικά αντικείμενα που αναγνωρίστηκαν από το Image Labeling	70
4.21	Διαφορετικά αντικείμενα που ανιχνεύθηκαν από το Object Detection	70
4.22	Dataframe διαφορετικών μαθημάτων	71
4.23	Εικόνα βασικών στατιστικών	71
4.24	Dataframe διαφορετικών μαθημάτων ανά χρήστη	71
4.25	Στατιστικά μαθημάτων ανά χρήστη	71
4.26	Στατιστικά μαθημάτων ανά χρήστη	72
4.27	Στατιστικά μαθημάτων ανά χρήστη	72
4.28	Επικρατέστερο συναίσθημα ανά χρήστη	73
4.29	Ποσοστιαία κατανομή συναισθημάτων εφαρμογής emoricon	73

1 Εισαγωγή

Η ραγδαία ανάπτυξη της τεχνολογίας και των τεχνολογικών συσκευών, κατά τις τελευταίες δεκαετίες, έχει καθορίσει τον τρόπο με τον οποίο ζούμε, εργαζόμαστε και μαθαίνουμε. Το φαινόμενο αυτό είναι ακόμη πιο εμφανές στους νέους που ορίζονται ως ψηφιακά ιθαγενείς, καθώς γεννήθηκαν στην εποχή της τεχνολογικής εξέλιξης και μαθαίνουν πώς να χρησιμοποιούν την τεχνολογία από τα πρώτα κιόλας χρόνια της ζωής τους [8].

Η τεχνολογία είναι πολύ σημαντική στο να αυξήσει τα κίνητρα των μαθητών κατά την εκπαιδευτική διαδικασία. Μπορεί να κρατήσει τους μαθητές πιο αφοσιωμένους και να τους βοηθήσει να συγκρατήσουν πιο εύκολα τις πληροφορίες, προωθώντας την αλληλεπίδραση σε ένα ουσιαστικότερο πλαίσιο.

Η πανδημία που αιφνίδια εισέβαλε στις ζωές μας το 2019, βοήθησε την τεχνολογία να εισχωρήσει εκδυσιαστικά ακόμα και στις πιο απόμακρες τεχνολογικά δομές. Παράλληλα μια μεγάλη ποικιλία εφαρμογών ανέλαβε τα ηνία για να βοηθήσει στην επικοινωνία, στη διδασκαλία και στην περάτωση εργασιών που απαιτούσαν τη φυσική παρουσία των ανθρώπων.

1.1 Περιγραφή του Προβλήματος

Είναι δεδομένο, ότι ειδικά τώρα που έγιναν εμφανή σε όλους τα πλεονεκτήματα που προσφέρει η τεχνολογική ανάπτυξη και η χρήση της τεχνολογίας, η επιστροφή στην προηγούμενη κατάσταση είναι σχεδόν αδύνατη. Το πόσο όμως αποδοτική είναι η νέα πραγματικότητα και το πόσο αξιόπιστα επιτυγχάνεται η απομακρυσμένη μάθηση με τη χρήση υπολογιστών και κινητών συσκευών, είναι ερωτήματα που δεν έχουν ξεκάθαρη απάντηση και με τα οποία οι εκπαιδευτικοί θα πρέπει να πορευθούν από εδώ και στο εξής.

Δυστυχώς τα αποτελέσματα μίας τόσο ουσιαστικής αλλαγής ενδέχεται να εμφανιστούν χρόνια αργότερα δεδομένου ότι πρόκειται για μη ποσοτικοποιημένες μεταβλητές.

1.2 Σκοπός - Συνεισφορά της Διπλωματικής Εργασίας

Στην παρούσα διπλωματική εργασία γίνεται μία προσπάθεια να δοθούν τα εργαλεία στον εκπαιδευτικό ώστε να φροντίσει να γίνει η διδασκαλία πιο ουσιαστική. Κατασκευάσαμε μία Android εφαρμογή mobile learning, την emoricon, η οποία βασίζεται στη συγκομιδή δεδομένων.

Η emoricon δίνει τη δυνατότητα στον μαθητή (χρήστη) να παρακολουθήσει τα μαθήματά του σύγχρονα ή ασύγχρονα και να κρατήσει σημειώσεις για αυτά. Παράλληλα και κατά τη διάρκεια της μάθησης, χρησιμοποιείται η κάμερα του κινητού, και πολλοί άλλοι αισθητήρες και API για τη λήψη δεδομένων. Είναι προφανές ότι ως άνθρωποι αδυνατούμε να διαχειριστούμε έναν τόσο μεγάλο όγκο δεδομένων, για αυτό στη συγκεκριμένη εφαρμογή, είτε τα φιλτράρουμε και τα χρησιμοποιούμε αυτούσια, είτε τα εισάγουμε σε μοντέλα που έχουν εκπαιδευτεί με τη χρήση αλγορίθμων μηχανικής μάθησης, με σκοπό την εξαγωγή συμπερασμάτων. Η σωστή ανάλυση και διερμηνεία αυτών των δεδομένων, είναι που δημιουργεί την πληροφορία που θα μετατραπεί σε γνώση.

Τελικά επιδιώκουμε να χρησιμοποιηθεί αυτή η γνώση από τον εκπαιδευτικό, για να μπορεί να κριθεί η αποτελεσματικότητα της μάθησης ξεχωριστά για κάθε

μαθητή, αλλά και για το σύνολο της τάξης. Κάποιες από τις ατομικές πληροφορίες που παρέχουμε, είναι το συναίσθημα του μαθητή, το πλαίσιο στο οποίο λαμβάνει χώρα η εκμάθηση, η δραστηριότητά του, τα αντικείμενα που υπάρχουν στο χώρο, η τοποθεσία του και άλλα δεδομένα από τους αισθητήρες της συσκευής του.

Για να είναι εύκολη η διαχείριση των αναλυτικών στοιχείων και η επεξεργασία τους από τον εκπαιδευτικό, δημιουργήσαμε δύο online jupyter notebooks μέσω του Google Colab ώστε να του παρέχουμε πρόσβαση σε αυτές τις πληροφορίες. Αυτές μπορεί να τις αξιοποιήσει είτε για να δώσει εξατομικευμένες συμβουλές, είτε για να αξιολογήσει το εκπαιδευτικό περιεχόμενο που έχει δημιουργήσει καθώς και τον τρόπο διδασκαλίας του.

2 Θεωρητικό υπόβαθρο

Σε αυτό το κεφάλαιο παρουσιάζονται εν συντομία μερικές θεωρητικές έννοιες που είναι απαραίτητες για την κατανόηση του συγκεκριμένου project. Είναι απαραίτητο να κατανοήσουμε τις βασικές έννοιες, καθώς ως δομικές μονάδες θα βοηθήσουν μετέπειτα, στην δημιουργία ενός γενικότερου πλαισίου το οποίο είναι αναγκαίο για την κατανόηση της μεθοδολογίας που ακολουθήσαμε.

Αρχικά θα αναπτυχθούν οι ευρύτερες έννοιες και σταδιακά θα εμβαθύνουμε σε πιο συγκεκριμένα θέματα και σχετικές έννοιες. Μια ενότητα που θα καλύπτει το σχετικό ερευνητικό έργο θα συμπεριληφθεί επίσης προς το τέλος του κεφαλαίου

2.1 Συναίσθημα

Τα συναισθήματα καθορίζουν την ποιότητα της ζωής μας και αποτελούν αναπόσπαστο μέρος της, αλλά αν γίνουν πολύ έντονα ή παράλογα, μπορεί να προκαλέσουν προβλήματα. Εμφανίζονται σε κάθε σχέση που μας ενδιαφέρει, στο χώρο εργασίας, στις φιλίες μας, στις σχέσεις με τα μέλη της οικογένειας και στις πιο στενές μας σχέσεις. Μπορεί να μας οδηγήσουν να ενεργήσουμε με τρόπους που θεωρούμε ρεαλιστικούς και κατάλληλους, αλλά μπορούν επίσης και να μας οδηγήσουν να ενεργήσουμε με τρόπους για τους οποίους μετανιώνουμε τρομερά εκ των υστέρων. Όπως είπε ο Paul Ekman «Τα συναισθήματα μπορούν, και συχνά ξεκινούν πολύ γρήγορα, τόσο γρήγορα, που στην πραγματικότητα, ο συνειδητός εαυτός μας δεν συμμετέχει ή δεν είναι καν μάρτυρας του τι στο μυαλό μας πυροδοτεί ένα συναίσθημα σε κάθε συγκεκριμένη στιγμή» [9]. Ίσως είναι μια απλή αλλά κεντρική ιδέα αλλά τα συναισθήματα εξελίχθηκαν για να μας προετοιμάσουν να αντιμετωπίζουμε γρήγορα, γεγονότα ζωτικής σημασίας.

Κάθε συναίσθημα συνδέεται με μοναδικά σήματα, με τα πιο αναγνωρίσιμα να εντοπίζονται στο πρόσωπο και τη φωνή. Κάθε άνθρωπος μπορεί να αντιδρά διαφορετικά απέναντι στο ίδιο γεγονός. Όμως υπάρχουν γεγονότα που όλοι οι άνθρωποι αντιδρούν με παρόμοιο τρόπο. Για παράδειγμα, ο φόβος μας κυριεύει όλους στιγμιαία πριν από ένα αυτοκινητιστικό ατύχημα, χωρίς να προλάβουμε να συνειδητοποιήσουμε την σκέψη μας ή να ελέγξουμε την αντίδρασή μας. Εν κατακλείδι, μπορούμε να πούμε ότι πρώτον, τα συναισθήματα είναι αντιδράσεις σε θέματα που φαίνονται πολύ σημαντικά για την ευημερία μας, και δεύτερον, ότι τα συναισθήματα συχνά ξεκινούν τόσο γρήγορα που δεν έχουμε επίγνωση των διεργασιών που συμβαίνουν στο μυαλό μας και τα πυροδοτούν.

2.1.1 Διαχωρισμός συναισθημάτων

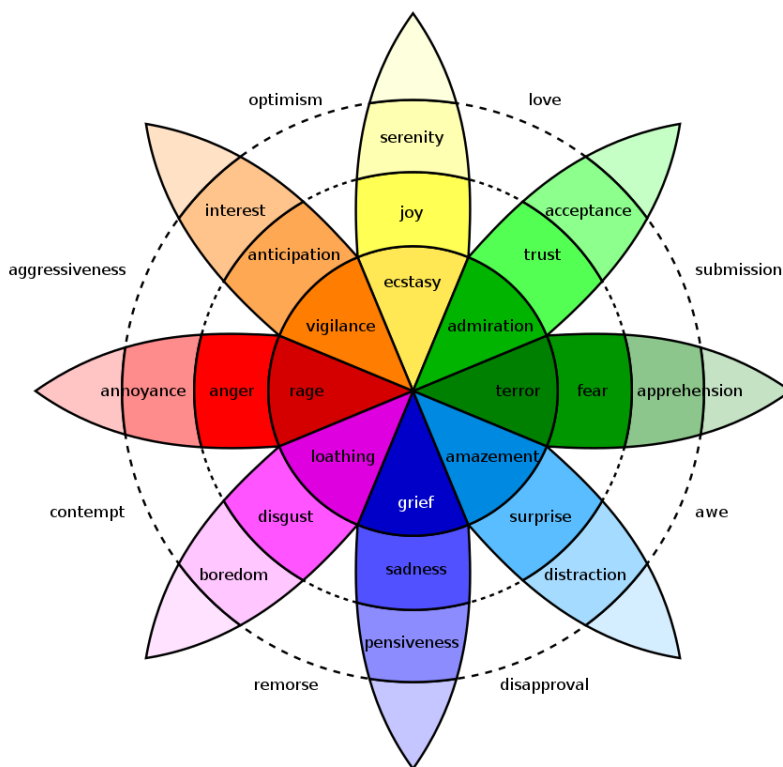
Ήδη από τον 4ο αιώνα π.Χ., ο Αριστοτέλης προσπάθησε να προσδιορίσει τον ακριβή αριθμό των βασικών συναισθημάτων του ανθρώπου. Στην «Λίστα των Συναισθημάτων», ο φιλόσοφος πρότεινε δεκατέσσερις διακριτές συναισθηματικές εκφράσεις: φόβος, εμπιστοσύνη, θυμός, φιλία, ηρεμία, εχθρότητα, ντροπή, αιδώ, οίκτος, καλοσύνη, φθόνος, αγανάκτηση, μίμηση και περιφρόνηση. [10]

Στη δημοσίευσή του «Η έκφραση των συναισθημάτων στον άνθρωπο και στα ζώα» (1872), ο Κάρολος Δαρβίνος διατύπωσε τη θεωρία ότι τα συναισθήματα είναι έμφυτα, έχουν εξελιχθεί και έχουν λειτουργικό σκοπό. Αν και ο Δαρβίνος δεν όρισε ρητά αυτά τα «βασικά συναισθήματα», πιστεύεται ότι οραματιζόταν έναν μικρότερο κα-

τάλογο βασικών συναισθημάτων, όπως ο φόβος, ο θυμός, η θλίψη, η ευτυχία και η αγάπη. [11]

Ο Τροχός των Συναισθημάτων

Μια από τις πιο γνωστές θεωρίες του 20ού αιώνα είναι ο τροχός των συναισθημάτων του Robert Plutchik. Σε αυτόν, ο Plutchik πρότεινε οκτώ βασικά συναισθήματα: χαρά, θλίψη, εμπιστοσύνη, αηδία, φόβο, θυμό, έκπληξη και προσμονή, τα οποία πίστευε ότι επικαλύπτονται και διαχέονται μεταξύ τους όπως οι αποχρώσεις σε έναν χρωματικό κύκλο. [12] Ο Plutchick εξήγησε περαιτέρω ότι τα βασικά συναισθηματικά «χρώματα» μπορούν να συνδυαστούν για να σχηματίσουν τα δευτερεύοντα και συμπληρωματικά συναισθηματικά «χρώματα». Για παράδειγμα, η προσδοκία συν τη χαρά μπορεί να συνδυαστούν για να σχηματίσουν την αισιοδοξία, ενώ ο φόβος και η έκπληξη μπορούν μαζί να περιγράψουν το δέος.



Σχήμα 2.1: Ο τροχός των συναισθημάτων

Πολλοί ερευνητές σε πρόσφατες έρευνες, όπως η [11] του 2012, αμφισβήτησαν το μοντέλο του Plutchik και υποστήριξαν ότι τα δευτερεύοντα και συμπληρωματικά συναισθήματα μπορεί συχνά να διαφέρουν ανάλογα με τον πολιτισμό ή την κοινωνία. Επιμένουν ότι, για να θεωρηθεί ένα συναίσθημα θεμελιώδες, πρέπει να βιώνεται καθολικά σε όλους τους πολιτισμούς.

Αξίζει να αναφερθεί ότι σε παλιότερες έρευνες όπως η συγκεκριμένη εν έτη 2000, είχε εντοπιστεί μια τεράστια ποικιλία άλλων συναισθημάτων που μπορούν να εμφανιστούν μέσω του προσώπου, επιβεβαιώνοντας ουσιαστικά τον τροχό των συναισθημάτων του Robert Plutchik. Ένα παράδειγμα αυτών των ερευνών είναι το [13], όπου

διακρίνονται 136 συναισθηματικές καταστάσεις και ταξινομούνται σε μια δενδρική δομή όπου κάποια πιο βασικά συναισθήματα βρίσκονται στην κορυφή (αγάπη, χαρά, έκπληξη, θυμός, θλίψη, φόβος). Αυτά τα πρωτογενή συναισθήματα χωρίζονται επιπλέον σε δευτερογενή αλλά και σε τριτογενή συναισθήματα.

Το μοντέλο των 6 συναισθημάτων

Σύμφωνα με τον Dr. Paul Ekman, ένα ανθρώπινο πρόσωπο μπορεί να πάρει περισσότερες από 10 χιλιάδες εκφράσεις αλλά τα συναισθήματα που κρύβονται πίσω από αυτές παραμένουν τα ίδια. [9]

Κατά τη δεκαετία του 1970, ο ψυχολόγος Paul Ekman προσδιόρισε έξι βασικά συναισθήματα τα οποία, όπως υποστήριξε, βιώνονται καθολικά σε όλους τους ανθρώπινους πολιτισμούς. Τα συναισθήματα που προσδιόρισε ήταν η ευτυχία, η θλίψη, η αηδία, ο φόβος, η έκπληξη και ο θυμός. [14]

Αργότερα επέκτεινε τον κατάλογο των βασικών συναισθημάτων του για να συμπεριλάβει και την περιφρόνηση και δημιούργησε αυτό που ονόμασε σύστημα κωδικοποίησης της δράσης του προσώπου (FACS). Πρόκειται για ένα μοντέλο ταξινόμησης το οποίο μετρά και αξιολογεί τις κινήσεις των μυών του προσώπου καθώς και εκείνες των ματιών και του κεφαλιού.

2.1.2 Εκφράσεις έναντι συναισθημάτων

Σε συνέχεια του έργου του Ekman, μια ερευνητική ομάδα του Πανεπιστημίου της Γλασκώβης το 2016 είχε ως στόχο να εντοπίσει τα συναισθήματα με βάση τις εκφράσεις του προσώπου ανεξάρτητα από τις κοινωνικοπολιτισμικές επιρροές. [15]

Αυτό που διαπίστωσαν οι ερευνητές ήταν ότι ορισμένα συναισθήματα προκαλούσαν την ίδια αντίδραση του προσώπου. Ο φόβος και η έκπληξη, για παράδειγμα, απασχολούσαν τους ίδιους μύες του προσώπου και, αντί να αντιπροσωπεύουν δύο συναισθήματα, μπορούσαν να θεωρηθούν ένα. Το ίδιο θα μπορούσε να εφαρμοστεί στην αηδία και το θυμό ή στον ενθουσιασμό και το σοκ. [16]

Με βάση τα ευρήματά τους, οι επιστήμονες περιόρισαν τον αριθμό των μη αναγώγιμων συναισθημάτων σε μόλις τέσσερα: ευτυχία, θλίψη, θυμός και φόβος. Πέρα από αυτά, υποστήριξαν, οι πιο σύνθετες παραλλαγές των συναισθημάτων έχουν εξελιχθεί κατά τη διάρκεια των χιλιετιών κάτω από πολυάριθμες κοινωνικές και πολιτισμικές επιρροές. [15]

Η ομοιότητα μεταξύ των εκφράσεων του προσώπου, οφείλεται κυρίως σε βιολογικούς παράγοντες καθώς γεννιόμαστε με αυτούς. Ενώ η διάκριση μεταξύ λεπτών και σύνθετων συναισθηματικών εκφράσεων είναι κυρίως κοινωνιολογική καθώς την μαθαίνουμε και αναπτύσσουμε με την πάροδο του χρόνου πολιτισμικά.

Μία έρευνα του Dr. Paul Ekman το 2020, επιβεβαίωσε την ύπαρξη καθολικών εκφράσεων προσώπου και προχώρησε την ιδέα ακόμη παραπέρα, υποδεικνύοντας ότι μπορεί να μοιραζόμαστε τις παρακάτω δεκαέξι σύνθετες εκφράσεις [17]:

- Διασκέδαση
- Θυμός
- Δέος
- Συγκέντρωση

- Σύγχυση
- Περιφρόνηση
- Ικανοποίηση
- Επιθυμία
- Απογοήτευση
- Αμφιβολία
- Ενθουσιασμός
- Ενδιαφέρον
- Πόνος
- Θλίψη
- Έκπληξη
- Θρίαμβος

2.2 Τεχνητή Νοημοσύνη

Τεχνητή Νοημοσύνη είναι ο τομέας της επιστήμης των υπολογιστών ο οποίος ασχολείται με τη σχεδίαση και την υλοποίηση προγραμμάτων που μπορούν να μιμηθούν τις ανθρώπινες γνωστικές ικανότητες. Εμφανίζουν έτσι χαρακτηριστικά που συνήθως αποδίδουμε σε ανθρώπινη συμπεριφορά, όπως η μάθηση, η επίλυση προβλημάτων, η προσαρμοστικότητα, η εξαγωγή συμπερασμάτων, η κατανόηση της φυσικής γλώσσας, η επίλυση προβλημάτων κα [18].

Ο John McCarthy μία από τις σημαντικότερες φυσιογνωμίες στη θεωρητική πληροφορική ορίζει τον τομέα αυτόν ως την επιστήμη και μεθοδολογία της δημιουργίας νοούντων μηχανών και ειδικότερα ευφυών προγραμμάτων υπολογιστών [19]. Σχετίζεται με το παρόμοιο έργο της χρήσης υπολογιστών για την κατανόηση της ανθρώπινης νοημοσύνης, αλλά η τεχνητή νοημοσύνη δεν χρειάζεται να περιορίζεται σε μεθόδους που είναι βιολογικά παρατηρήσιμες. Για τη «νοημοσύνη» στον όρο τεχνητή νοημοσύνη, αναφέρει ότι είναι το υπολογιστικό μέρος της ικανότητας επίτευξης στόχων ανά τον κόσμο. Διαφορετικά είδη και βαθμοί νοημοσύνης εμφανίζονται σε ανθρώπους, σε πολλά ζώα αλλά και σε ορισμένες μηχανές.

Οι επιστήμες οι οποίες συνεισφέρουν στην τεχνική νοημοσύνη είναι πολλές πέρα από την πληροφορική: όπως η ψυχολογία, η νευρολογία, η επιστήμη των μηχανών, η γλωσσολογία, ακόμα και η φιλοσοφία προκειμένου να γίνει εφικτή η σύνδεση της ευφυούς συμπεριφοράς με την εκμάθηση και την προσαρμογή στο εκάστοτε περιβάλλον των μηχανών και υπολογιστών.

Κλασική και υπολογιστική τεχνητή νοημοσύνη

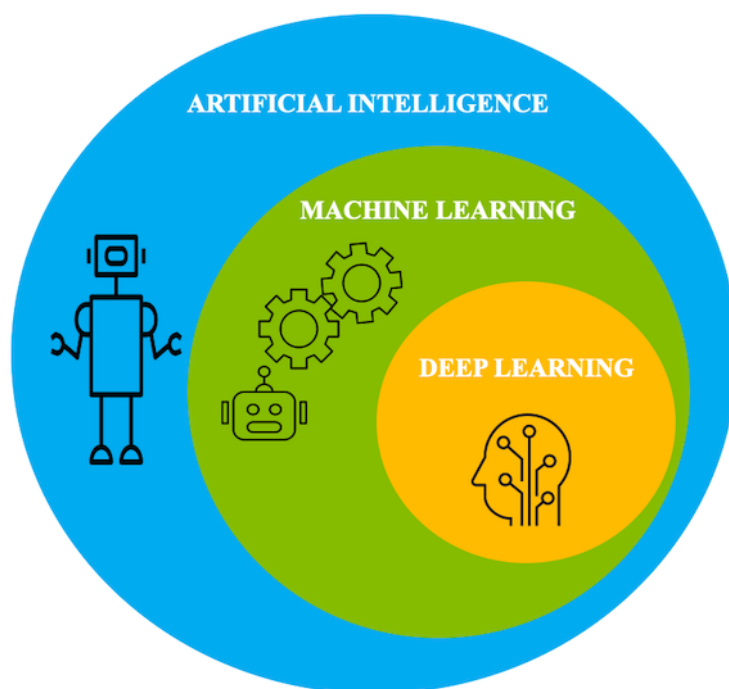
Διακρίνουμε δύο προσεγγίσεις για την τεχνητή νοημοσύνη. Αυτές είναι η κλασική (ή συμβολική) και η υπολογιστική (ή συνδετική) τεχνητή νοημοσύνη.

Η συμβολική τεχνητή νοημοσύνη βασίζεται στην κατανόηση των νοητικών διεργασιών και ασχολείται με τη προσομοίωση της ανθρώπινης νοημοσύνης προσεγγίζοντας την με αλγορίθμους και συστήματα κανόνων υψηλού επιπέδου. Σε αυτή συμπεριλαμβάνονται η αναπαράσταση γνώσης (knowledge representation), η συλλογιστική (reasoning) και η πρόβλεψη/προγραμματισμός (prediction / planning).

Η μη συμβολική τεχνητή νοημοσύνη, βασίζεται στη μίμηση της βιολογικής λειτουργίας του εγκεφάλου. Πρακτικά προσπαθεί να αναπαράγει την ανθρώπινη ευφυΐα χρησιμοποιώντας στοιχειώδη αριθμητικά μοντέλα που συνθέτουν επαγωγικά νοήμονες συμπεριφορές μέσω της διαδοχικής αυτοοργάνωσης απλούστερων δομικών συστατικών. Με αυτόν τον τρόπο επιτυγχάνεται η προσομοίωση πραγματικών βιολογικών διαδικασιών, όπως για παράδειγμα η λειτουργία του ανθρώπινου εγκεφάλου και η εξέλιξη των ειδών.

Η μηχανική μάθηση που θα δούμε σε επόμενο κεφάλαιο αποτελεί μία από τις υποκατηγορίες που ανήκουν στην μη συμβολική τεχνητή νοημοσύνη και ασχολείται με την βελτίωση της γνώσης και της απόδοσης της μηχανής μέσω δεδομένων αλληλεπιδράσεων με το περιβάλλον κ.α. Αποτελεί αν όχι το μεγαλύτερο, μία από τις μεγαλύτερες κατηγορίες της τεχνητής νοημοσύνης και αποτελείται από αλγόριθμους που συνεχώς βελτιώνουν την απόδοσή τους, καθώς με την πάροδο του χρόνου εκπαιδεύονται πάνω σε περισσότερα δεδομένα.

Επίσης ένα υποσύνολο της μηχανικής μάθησης είναι η βαθιά μάθηση στην οποία πολυεπίπεδα νευρωνικά δίκτυα μαθαίνουν από πολύ μεγάλες ποσότητες δεδομένων. Αν και η ανοδική πορεία της τεχνητής νοημοσύνης είναι αδιαμφισβήτητη, παρόλα αυτά εξακολουθεί να αποτελεί ένα ανοικτό και πρόσφορο πεδίο προς περαιτέρω διερεύνηση και έρευνα.



Σχήμα 2.2: AI-ML-DL [1]

2.2.1 Τύποι Τεχνητής Νοημοσύνης με βάση την λειτουργία τους

Με βάση τη λειτουργικότητα των συστημάτων που βασίζονται στην Τεχνητή Νοημοσύνη, η Τεχνητή Νοημοσύνη μπορεί να κατηγοριοποιηθεί στους ακόλουθους τύπους συστημάτων [18]:

1. Αντιδραστικά μηχανήματα : αντιδρούν λογικά
2. Μηχανές με περιορισμένη μνήμη : σκέφτονται λογικά
3. Μηχανές με θεωρία του νου : σκέφτονται σαν άνθρωποι
4. Μηχανές με αυτογνωσία : συμπεριφέρονται όπως οι άνθρωποι

Ο Άλαν Τούρινγκ το 1950, σε ένα άρθρο του για την πιθανότητα ανάπτυξης τεχνητής νοημοσύνης επινόησε ένα τεστ ειδικά για αυτόν το σκοπό. Έχοντας ασχοληθεί επί μακράν με το θέμα πρότεινε το ακόλουθο κριτήριο: αν ένας άνθρωπος μπορεί να συμμετέχει σε μια συζήτηση για πέντε λεπτά χωρίς να καταλάβει ότι μιλάει σε μια μηχανή, τότε ο υπολογιστής περνάει το τεστ. Με βάση το συγκεκριμένο τεστ, αν δεν μπορούμε να ξεχωρίσουμε τον άνθρωπο από τη μηχανή, τότε η μηχανή πρέπει να είναι ευφυής. Ο Τούρινγκ, είχε προβλέψει πως μέχρι το 2000 θα έχει αναπτυχθεί η τεχνητή νοημοσύνη τόσο, που θα μπορούσε έπειτα από πέντε λεπτά συζήτησης να ξεγελάσει το 30% των ερωτώντων.

Το πρώτο πρόγραμμα H/Y που πέρασε το συγκεκριμένο τεστ, θεωρείται πως είναι το ELIZA, που έφτιαξε το 1976 ο Αμερικανός προγραμματιστής Τζόσεφ Βάιζενμπουμ, και με το οποίο κατάφερε να πείσει τη γραμματέα του πως συνομιλούσε με εκείνον. Από τότε έχουν περάσει το τεστ και άλλα προγράμματα, από τα οποία συμπεραίνουμε πως ο έλεγχος του Τούρινγκ, παρόλο που μπορεί να αποδείξει πως οι άνθρωποι μπορούν να ξεγελαστούν από μηχανές, δεν απαντά στα σύγχρονα ερωτήματα για την τεχνητή νοημοσύνη, όπως για παράδειγμα την ποσοτικοποίηση της νοημοσύνης, τις συνέπειες της, ή το να προβλέπει το πότε θα επιτευχθεί.

Όμως οι άνθρωποι κάνουν λάθη, και προσπαθώντας να φτιάξουμε μια μηχανή να συμπεριφέρεται όπως ο άνθρωπος, θα μας ενδιέφερε αυτή να κάνει τα ίδια λάθη.

Αντιδραστικά μηχανήματα (Reactive Machines)

Ο πιο βασικός τύπος τεχνητής νοημοσύνης είναι η αντιδραστική τεχνητή νοημοσύνη (Reactive AI), η οποία είναι προγραμματισμένη να παρέχει μια προβλέψιμη έξοδο με βάση την είσοδο που λαμβάνει. Οι αντιδραστικές μηχανές ανταποκρίνονται πάντα σε πανομοιότυπες καταστάσεις με τον ίδιο ακριβώς τρόπο κάθε φορά και δεν είναι σε θέση να μαθαίνουν ενέργειες ή να αντιλαμβάνονται το παρελθόν ή το μέλλον. [20]

Η αντιδραστική τεχνητή νοημοσύνη ήταν ένα τεράστιο βήμα στην ιστορία της ανάπτυξης της τεχνητής νοημοσύνης, αλλά αυτού του είδους η τεχνητή νοημοσύνη δεν μπορεί να λειτουργήσει πέρα από τις εργασίες για τις οποίες σχεδιάστηκε αρχικά. Οι αντιδραστικές μηχανές (Reactive Machines) εκτελούν βασικές λειτουργίες και δεν υπάρχει μάθηση. Αυτό είναι το πρώτο στάδιο κάθε συστήματος τεχνητής νοημοσύνης. Μια μηχανή μάθησης που δέχεται ένα ανθρώπινο πρόσωπο ως είσοδο και εξάγει ένα πλαίσιο γύρω από το πρόσωπο για να το αναγνωρίσει ως πρόσωπο είναι μια απλή, αντιδραστική μηχανή. Το μοντέλο δεν αποθηκεύει καμία είσοδο και δεν πραγματοποιεί καμία μάθηση [21]. Αυτό τις καθιστά εγγενώς περιορισμένες και ώριμες για βελτίωση. Οι επιστήμονες ανέπτυξαν τον επόμενο τύπο τεχνητής νοημοσύνης από αυτή τη βάση.

Μηχανές με περιορισμένη μνήμη (Limited Memory Machines)

Οι μηχανές με περιορισμένη μνήμη (Limited Memory Machines) έχουν κάποιες δυνατότητες μνήμης αλλά και πάλι σε περιορισμένο βαθμό. Εξ ου και η ονομασία, δηλαδή περιορισμένη μνήμη. Με τέτοιες μηχανές τεχνητής νοημοσύνης, είμαστε σε θέση να πάρουμε σωστότερες αποφάσεις στο μέλλον, με βάση δεδομένα του παρελθόντος. Χρησιμοποιούν δηλαδή πληροφορίες που λαμβάνονται από μια βάση δεδομένων, καταγράφουν βασικές πληροφορίες για το περιβάλλον και επιπλέον μαθαίνουν από αυτές.

Το μεγαλύτερο παράδειγμα τέτοιων μοντέλων υπάρχει στο λογισμικό μοντελοποίησης αυτοκινήτων και στα συστήματα GPS που έχουν κατασκευαστεί με τεχνητή νοημοσύνη. Αυτά είναι σε θέση να παρέχουν χρήσιμες πληροφορίες στους χρήστες για την αποφυγή μπουτιλιαρίσματος, τη λήψη των καλύτερων δυνατών διαδρομών και ακόμη και την ελαχιστοποίηση των κινδύνων πιθανών ατυχημάτων [21].

Στις μηχανές με περιορισμένη μνήμη υπάρχουν τα παρακάτω τρία βασικά είδη μοντέλων μηχανικής μάθησης:

- Reinforcement learning
- Long Short Term Memory (LSTMs)
- Evolutionary Generative Adversarial Networks (E-GAN)

Μηχανές με θεωρία του νου (Theory of Mind)

Οι μηχανές με τη θεωρία του νου, είναι ένας τύπος τεχνητής νοημοσύνης που βρίσκεται ακόμη σε εξέλιξη. Αναμένεται ότι στο μέλλον ορισμένες μηχανές θα είναι σε θέση να κατανοήσουν τόσο την ανθρώπινη σκέψη όσο και τα συναισθήματα και να λάβουν αποφάσεις από αυτήν, κάτι που συνεπάγεται κοινωνική αλληλεπίδραση.

Αυτοί είναι οι τύποι ρομπότ που είναι σε θέση να χρησιμοποιούν την κοινή λογική τους για να ερμηνεύουν δεδομένα και να λαμβάνουν αποφάσεις σε πραγματικό χρόνο. Γίνονται πολλές έρευνες για να αναπτυχθούν μηχανές που να αναπτύσσουν μια έμφυτη κοινή λογική που τους επιτρέπει όχι μόνο να αναλύουν δεδομένα αλλά και να τα αξιολογούν προς μεγαλύτερο όφελος [22].

Προς το παρόν, τα πλησιέστερα δυνατά ρομπότ είναι σε θέση να ερμηνεύουν σκάλες και άλλα ρομπότ που έρχονται από την αντίθετη κατεύθυνση. Με βάση την ερμηνεία των εμποδίων, το ρομπότ είναι σε θέση να προχωρήσουν σε μια ασφαλή καθοδική διάβαση. Ένα πολύ καλό παράδειγμα αυτού του τύπου τεχνητής νοημοσύνης, είναι το ρομπότ Sophia, που δημιουργήθηκε το 2016.

Μηχανές με αυτογνωσία (Self-Aware)

Οι μηχανές με αυτογνωσία (Self-Aware) θα μπορούσαν να είναι ικανές να έχουν αντιληπτές αντιλήψεις, σκέψεις και συμπεριφορές. Πρόκειται δηλαδή για μηχανές ικανές να αντιληφθούν, να δημιουργήσουν λόγο και να ενεργήσουν σαν ανθρώπινα όντα. Η δημιουργία αυτού του τύπου τεχνητής νοημοσύνης, η οποία απέχει δεκαετίες, αν όχι αιώνες από την υλοποίησή της, είναι και θα είναι πάντα ο απώτερος στόχος κάθε έρευνας για την τεχνητή νοημοσύνη.

2.2.2 Τύποι Τεχνητής Νοημοσύνης με βάση την μάθηση

Η τεχνική νοημοσύνη με βάση τη μάθηση θα μπορούσε να χωριστεί στις παρακάτω κατηγορίες:

1. Τεχνητή Στενή Ευφυΐα – Artificial Narrow Intelligence (ANI)
2. Τεχνητή Γενική Ευφυΐα – Artificial General Intelligence (AGI)
3. Τεχνητή Υπερευφυΐα - Artificial Super Intelligence (ASI)

Τεχνητή Στενή Ευφυΐα – Artificial Narrow Intelligence (ANI)

Η Τεχνητή Στενή Ευφυΐα, γνωστή και ως αδύναμη τεχνητή νοημοσύνη ANI, είναι το στάδιο της τεχνητής νοημοσύνης που περιλαμβάνει μηχανές που μπορούν να εκτελούν μόνο ένα στενά καθορισμένο σύνολο συγκεκριμένων εργασιών. Σε αυτό το στάδιο, η μηχανή δεν διαθέτει καμία ικανότητα σκέψης, απλώς εκτελεί ένα σύνολο προκαθορισμένων λειτουργιών.

Παραδείγματα αδύναμης τεχνητής νοημοσύνης είναι η Siri, η Alexa, τα αυτοκινούμενα αυτοκίνητα, το Alpha-Go, το ανθρωποειδές Sophia και ούτω καθεξής. Σχεδόν όλα τα συστήματα που είναι βασισμένα στην τεχνητή νοημοσύνη και έχουν κατασκευαστεί μέχρι σήμερα, εμπίπτουν στην κατηγορία της αδύναμης τεχνητής νοημοσύνης.

Τεχνητή Γενική Ευφυΐα – Artificial General Intelligence (AGI)

Η Τεχνητή Γενική Ευφυΐα, γνωστή και ως Ισχυρή Τεχνητή Νοημοσύνη, η AGI είναι το στάδιο της εξέλιξης της Τεχνητής Νοημοσύνης όπου οι μηχανές θα έχουν την ικανότητα να σκέφτονται και να λαμβάνουν αποφάσεις όπως οι άνθρωποι.

Προς το παρόν δεν υπάρχουν παραδείγματα Ισχυρής Τεχνητής Νοημοσύνης, ωστόσο πιστεύεται ότι σύντομα θα είμαστε σε θέση να δημιουργήσουμε μηχανές που θα είναι εξίσου έξυπνες με τους ανθρώπους. Η ισχυρή τεχνητή νοημοσύνη θεωρείται απειλή για την ανθρώπινη ύπαρξη από πολλούς επιστήμονες, συμπεριλαμβανομένου και του Στίβεν Χόκινγκ που δήλωσε ότι: «Η ανάπτυξη πλήρους τεχνητής νοημοσύνης θα μπορούσε να σημάνει το τέλος της ανθρώπινης φυλής.... Θα απογειωθεί από μόνη της και θα επανασχεδιάζει τον εαυτό της με συνεχώς αυξανόμενο ρυθμό. Οι άνθρωποι, οι οποίοι περιορίζονται από την αργή βιολογική εξέλιξη, δεν θα μπορέσουν να την ανταγωνιστούν και θα αντικατασταθούν».

Τεχνητή Υπερευφυΐα - Artificial Super Intelligence (ASI)

Η Τεχνητή Υπερνοημοσύνη είναι το στάδιο της Τεχνητής Νοημοσύνης κατά το οποίο οι ικανότητες των υπολογιστών θα ξεπεράσουν τον άνθρωπο. Η ASI είναι επί του παρόντος μια υποθετική κατάσταση όπως απεικονίζεται σε ταινίες και βιβλία επιστημονικής φαντασίας, στην οποία οι μηχανές έχουν κατακτήσει τον κόσμο.

2.2.3 Περιοχές της τεχνητής νοημοσύνης

Η τεχνητή νοημοσύνη είναι μια σύνθεση μοντέλων, μεθοδολογιών, εργαλείων και συστημάτων που τα δανείζεται, από τις παρακάτω τέσσερις διαφορετικές επιστημονικές περιοχές, αλλά ταυτόχρονα τα ενοποιεί και τα εξειδικεύει [23]:

- τις Γνωστικές Επιστήμες (Cognitive Sciences), όπως φιλοσοφία, λογική, ψυχολογία, κυβερνητική,
- την μαθηματική επιστήμη και ειδικότερα την μαθηματική λογική, την θεωρία μοντέλων και αποδείξεων, επιχειρησιακή Έρευνα,
- την ευρύτερη επιστημονική περιοχή της πληροφορικής καθώς το κυριότερο προϊόν της τεχνητής νοημοσύνης είναι τα έξυπνα πληροφοριακά συστήματα, τα έμπειρα συστήματα, οι αποδείξεις θεωρημάτων, κ.λ.π.
- την αυτοματική και τη θεωρία ελέγχου σε συνεργασία με την βιοτεχνολογία, που προτείνουν την ανάπτυξη ευφυών συστημάτων σε επίπεδο υλικού (hardware, νευρωνικά δίκτυα)

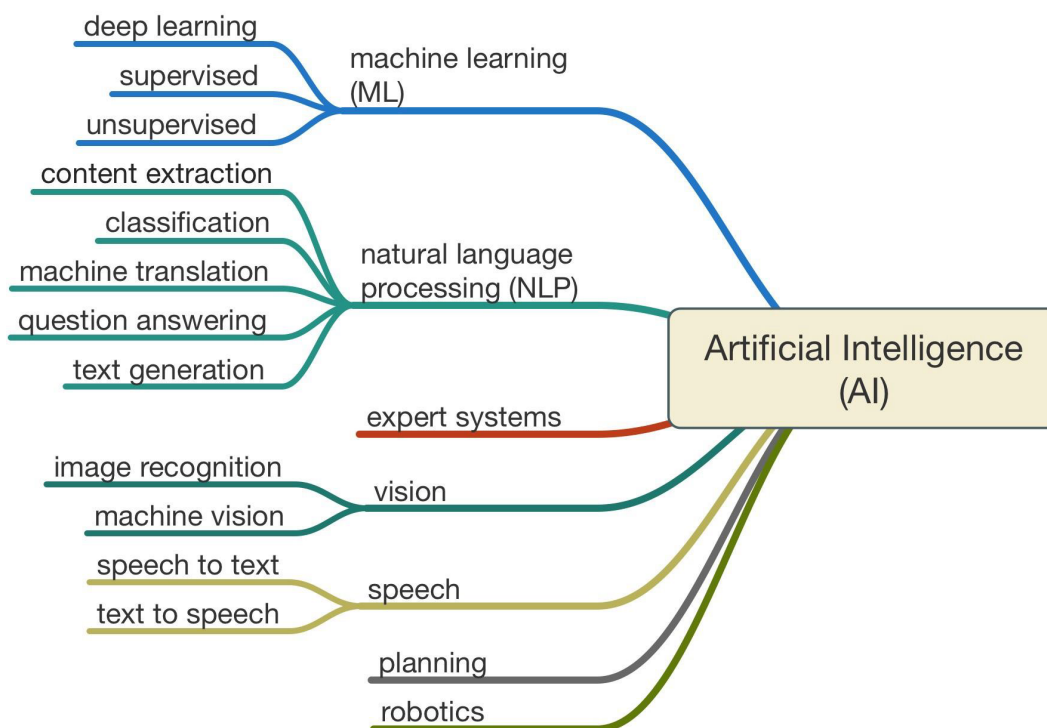
Η τεχνητή νοημοσύνη μπορεί να χρησιμοποιηθεί για την επίλυση προβλημάτων του πραγματικού κόσμου με την εφαρμογή των ακόλουθων διαδικασιών/τεχνικών:

- Μηχανική μάθηση
- Αναγνώριση Προτύπων
- Εξόρυξη και διαχείριση γνώσης
- Έμπειρα Συστήματα
- Συστήματα Γνώσης
- Ευφυείς πράκτορες
- Ρομποτική
 - επεξεργασία φυσικής γλώσσας
 - τεχνητή όραση
 - τεχνητή όσφρηση,
 - αυτόνομα συστήματα,
 - αναπαράσταση κίνησης

2.2.4 Machine Learning

Μηχανική μάθηση ονομάζουμε τη δημιουργία μοντέλων ή προτύπων από ένα σύνολο δεδομένων που προέρχονται από ένα υπολογιστικό σύστημα. Ανάλογα με το είδος του προβλήματος έχουν αναπτυχθεί διάφορες τεχνικές μηχανικής μάθησης. [18]

Η εισαγωγή του κλάδου της μηχανικής μάθησης στην επιστήμη των υπολογιστών, επέτρεψε στους υπολογιστές να μπορούν να αντιμετωπίσουν προβλήματα αντίληψης για τον πραγματικό κόσμο, όσο και να παίρνουν υποκειμενικές αποφάσεις. Αποτελεί ένα πολύ σημαντικό κομμάτι της τεχνητής νοημοσύνης και αφορά, την ανάπτυξη κατάλληλων αλγορίθμων με μοναδικό σκοπό την εκμάθηση στους υπολογιστές. Το λογισμικό που αναπτύσσεται είναι πλέον σε θέση να προσαρμόζεται μόνο του και να είναι αρκετά ευέλικτο βάση των δεδομένων που του δίνουμε, σε αντίθεση με την κλασική προσέγγιση που αφορούσε την αλγοριθμική ανάπτυξη με βάση τη διαίσθησή του μηχανικού. Οι αλγόριθμοι ML, επιτρέπουν σε συστήματα AI να προσαρμόζονται εύκολα σε καινούργια προβλήματα, απαιτώντας ελάχιστη παρέμβαση από τον άνθρωπο. Επιτρέπουν στις μηχανές να καταλαβαίνουν καταστάσεις και βασισμένα σε αυτές, να παίρνουν αποφάσεις. Ο αλγόριθμος μαθαίνει από τα πειραματικά δεδο-



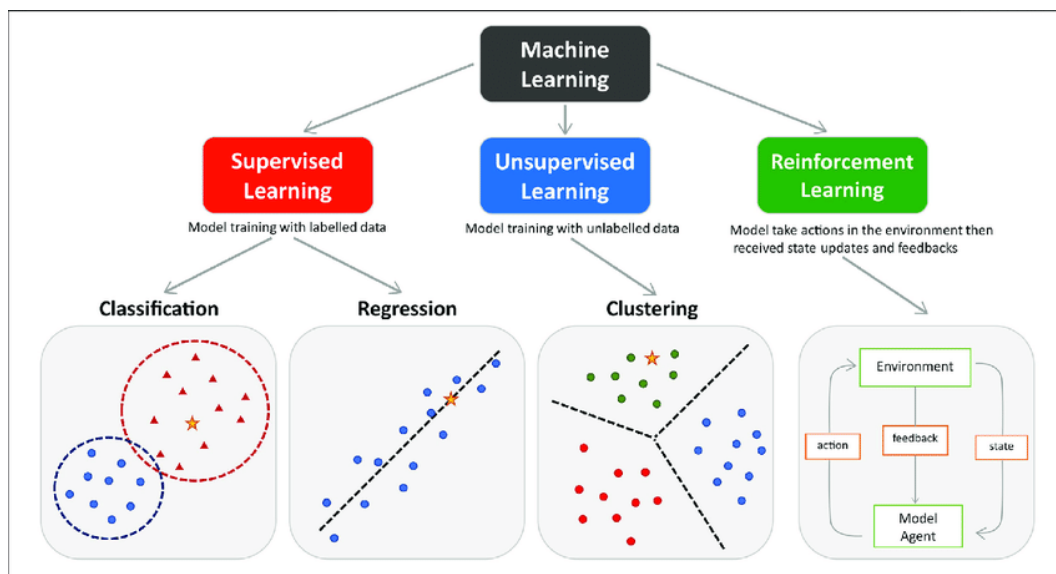
Σχήμα 2.3: Κλάδοι τεχνητής νοημοσύνης [2]

μένα και με αυτόν τον τρόπο βελτιώνει την επίδοσή του. Από την εκπαίδευση αυτή προκύπτει το μοντέλο, το οποίο μπορεί να κάνει προβλέψεις ή να πάρει αποφάσεις για τα νέα δεδομένα που του δίνονται ως προς το πρόβλημα που του έχει ανατεθεί [24].

Μπορούμε να διακρίνουμε τρεις βασικές κατηγορίες μηχανικής μάθησης όπως φαίνεται και στο σχήμα 2.4

- Τη μάθηση με επίβλεψη (Supervised Learning): Στο υπολογιστικό σύστημα καλείται να μάθει από ένα σύνολο δεδομένων. Του δίνονται παραδείγματα εισόδου και επιθυμητής εξόδου (στα δεδομένα έχουν προηγουμένως ανατεθεί ετικέτες λαβελς) και στόχος είναι να εξαχθεί ένας γενικός κανόνας αντιστοίχισης της εισόδου με την επιθυμητή έξοδο.
- Τη μάθηση χωρίς επίβλεψη - (Unsupervised Learning): Το σύστημα πρέπει να μάθει από τη δημιουργία προτύπων. Τα δεδομένα που του δίνονται δεν έχουν ετικέτες (labels) και ο αλγόριθμος ML καλείται να βρει από μόνος του δομές στα δεδομένα εισόδου.
- Την εκμάθηση δια ανταμοιβής - (Reinforcement Learning): Σκοπός του συστήματος είναι η μεγιστοποίηση της αριθμητικής τιμής της ανταμοιβής. Ο πράκτορας αλληλεπιδρά με ένα δυναμικό περιβάλλον στο οποίο πρέπει να εκτελέσει ένα συγκεκριμένο στόχο, χωρίς την ύπαρξη ενός "δασκάλου" που να ορίζει ρητά αν έχει φθάσει κοντά στο στόχο ενώ δεν υπάρχει αρχική γνώση για το πως θα επιτευχθεί ο στόχος.

Αξίζει να αναφερθεί ότι κάποια προβλήματα είναι υβριδικά, και αποτελούν συνδυασμό των πιο πάνω κατηγοριών, όπως για παράδειγμα η μάθηση με ημιεπίβλεψη που αποτελεί ένα συνδυασμό των δύο πρώτων κατηγοριών.



Σχήμα 2.4: Κατηγορίες μηχανικής μάθησης [3]

2.2.5 Deep Learning

Η βαθιά μηχανική μάθηση καταφέρνει να μαθαίνει να αναπαριστά τον κόσμο ως μία ένθετη ιεραρχία εννοιών, όπου η κάθε έννοια ορίζεται σε σχέση με άλλες πιο απλές έννοιες, και πιο αφηρημένες μορφές αναπαραστάσεων σε σχέση με λιγότερο αφηρημένες.

Σε μια προσπάθεια να δημιουργηθούν συστήματα που μαθαίνουν παρόμοια με τον τρόπο που μαθαίνουν οι άνθρωποι, η υποκείμενη αρχιτεκτονική της βαθιάς μάθησης εμπνεύστηκε από τη δομή του ανθρώπινου εγκεφάλου. Για το λόγο αυτό, αρκετές θεμελιώδεις ορολογίες στο πλαίσιο της βαθιάς μάθησης μπορούν να αντιστοιχηθούν με τη νευρολογία.

Παρόμοια με τον τρόπο με τον οποίο οι νευρώνες αποτελούν τα θεμελιώδη δομικά στοιχεία του εγκεφάλου, έτσι και η αρχιτεκτονική της βαθιάς μάθησης περιέχει μια υπολογιστική μονάδα που επιτρέπει τη μοντελοποίηση μη γραμμικών συναρτήσεων και ονομάζεται perceptron.

2.2.6 Νευρωνικά Δίκτυα

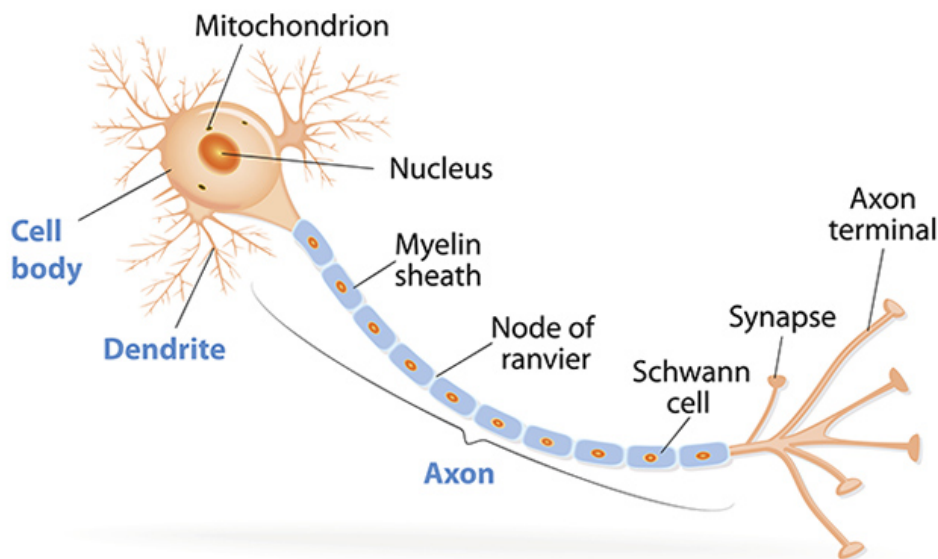
Τα Τεχνητά Νευρωνικά Δίκτυα είναι εμπνευσμένα από το βιολογικό νευρικό σύστημα του ανθρώπου. Η βασική επεξεργαστική μονάδα του εγκεφάλου είναι ο νευρώνας (σχήμα 2.5), με το ανθρώπινο νευρικό σύστημα να αποτελείται από περίπου 86 εκατομμύρια νευρώνες και περίπου

$$[10^{14}, 10^{15}]$$

διασυνδέσεις.

Ένας βιολογικός νευρώνας αποτελείται από τον πυρήνα, τον άξονά του, διάφορους δενδρίτες και από συνάψεις ανάμεσα στους δενδρίτες και στους άξονες άλλων νευρώνων ή ανάμεσα στον άξονά του και τους δενδρίτες άλλων νευρώνων.

Τα κύρια μέρη ενός νευρώνα είναι τα εξής



Σχήμα 2.5: Βιολογικός Νευρώνας

- Δενδρίτης (Dendrites): Είναι πύλες μέσω των οποίων εισέρχονται ηλεκτρικά σήματα/παλμοί στον νευρώνα.
- Πυρήνας (kernel): Επεξεργάζεται τους εισερχόμενους παλμούς
- Σώμα του κυττάρου (Cell body): Εξάγει συμπεράσματα, με βάση τις εισόδους.
- Άξονας (Axon): Είναι πύλη μέσω του οποίου διοχετεύονται ηλεκτρικοί παλμοί στους δενδρίτες άλλων νευρώνων. Πρακτικά συνδέει την έξοδο που λαμβάνεται από το σώμα του κυττάρου με τις νευραξονικές απολήξεις.
- Νευραξονικές απολήξεις ή συνάψεις: Είναι σύνδεσμοι ανάμεσα σε άξονες και δενδρίτες που επιτρέπουν τη διέλευση ηλεκτρικών παλμών. Συνδέουν τον άξονα του νευρώνα με τις συνάψεις, από όπου και μεταφέρεται η πληροφορία στην είσοδο άλλων νευρώνων.

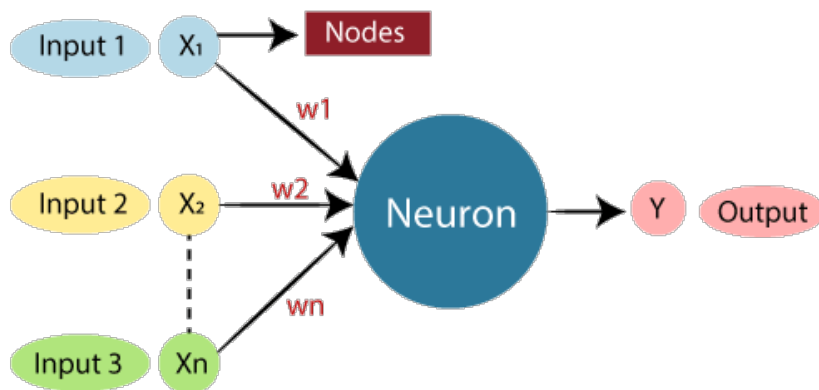
Η λειτουργία των βιολογικών νευρώνων μοντελοποιείται με τον τεχνητό νευρώνα που αναπαρίσταται στο σχήμα 2.7 και όπως ο «νευρώνας» στον ανθρώπινο εγκέφαλο μεταδίδει ηλεκτρικούς παλμούς σε ολόκληρο το νευρικό μας σύστημα, έτσι και το perceptron λαμβάνει έναν κατάλογο σημάτων εισόδου και τα μετατρέπει σε σήματα εξόδου.

Ένας τεχνητός νευρώνας με M δενδρίτες δέχεται μέσω M συνάψεων με αντίστοιχα συναπτικά βάρη w_1, w_2, \dots, w_M , τα σήματα εισόδου $U_1(t), U_2(t), \dots, U_M(t)$. Ο πυρήνας του νευρώνα αθροίζει τα σήματα εισόδου και στη συνέχεια εφαρμόζει σε αυτό το άθροισμα ένα μετασχηματισμό που ονομάζεται συνάρτηση ενεργοποίησης (στο σχήμα 2.7 συμβολίζεται με f).

Επειδή ένας νευρώνας μπορεί να είναι πολωμένος (βιασεδ - β), το μαθηματικό μοντέλο που τον περιγράφει πλήρως παίρνει την μορφή:

$$y(t) = f(W_1 U_1(t) + W_2 U_2(t) + \dots + W_M U_M(t)) - \theta$$

όπου η $y(t)$ είναι το σήμα εξόδου που τροφοδοτείται προς άλλους νευρώνες και θ

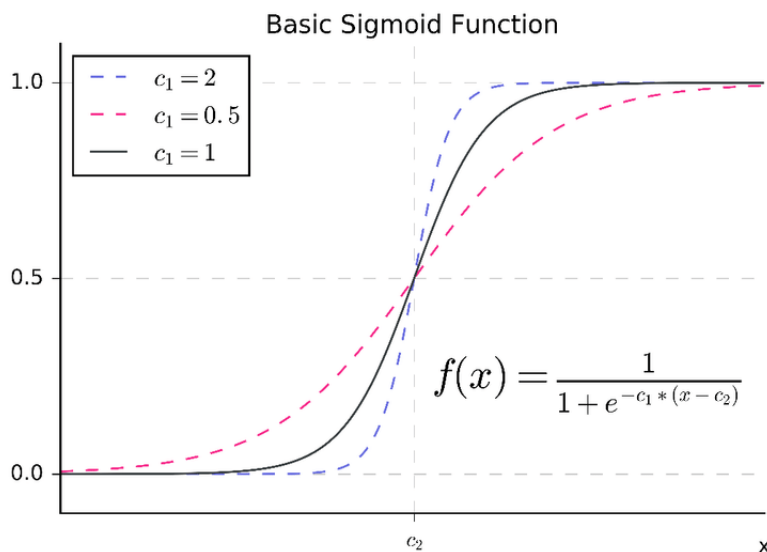


Σχήμα 2.6: Δομή και λειτουργία τεχνητού νευρώνα

ένα κατώφλι ενεργοποίησης του νευρώνα.

Ο νευρώνας μπορεί να παράγει ένα ισχυρό σήμα (παλμό) αν το άθροισμα των σημάτων εισόδου ξεπερνάει το κατώφλι ενεργοποίησης θ . Αν το κατώφλι ενεργοποίησης δεν ξεπερνιέται τότε δεν παράγεται ισχυρό σήμα στον άξονα του νευρώνα.

Επομένως για τη μοντελοποίηση της συνάρτησης ενεργοποίησης μπορούμε να χρησιμοποιήσουμε μια σιγμοειδή συνάρτηση (sigmoid function), όπως για παράδειγμα η λογιστική συνάρτηση (logistic function).



Σχήμα 2.7: Σιγμοειδής συνάρτηση

Εναλλακτικά, η σιγμοειδής συνάρτηση ενεργοποίησης μπορεί να εκφραστεί σε διακριτή μορφή ως

$$f(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases} \tag{2.1}$$

Η επιλογή κατάλληλης συνάρτησης ενεργοποίησης των νευρώνων δεν είναι τυχαία, αφού επηρεάζει την απόδοση του αλγορίθμου Backpropagation, ο οποίος χρησιμοποιείται για την εκπαίδευση των νευρωνικών δικτύων.

Η παραπάνω λογιστική και άλλες σιγμοειδείς συναρτήσεις περιγράφουν με αρ-

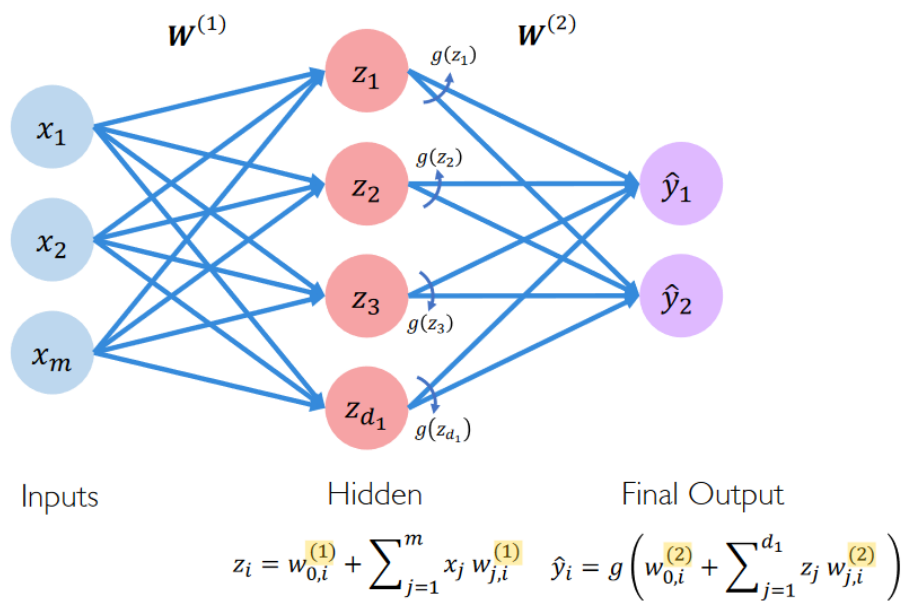
κετή ακρίβεια τη λειτουργία (στοιχειώδη επεξεργασία) του πυρήνα ενός νευρώνα και χρησιμοποιούνται στη σύνθεση τεχνητών νευρωνικών δικτύων.

Όπως ένα βιολογικό νευρωνικό δίκτυο αποτελείται από μεγάλο αριθμό διασυνδεδεμένων βιολογικών νευρώνων, έτσι και ένα τεχνητό νευρωνικό δίκτυο αποτελείται από πολλούς διασυνδεδεμένους τεχνητούς νευρώνες.

Σκοπός του τεχνητού νευρωνικού δικτύου είναι η κατανόηση της αναπαράστασης δεδομένων και αυτό επιτυγχάνεται με τη στοίβαξη πολλών επιπέδων. Κάθε επίπεδο είναι υπεύθυνο για την κατανόηση κάποιου μέρους της εισόδου.

Η έξοδος από το εκάστοτε επίπεδο μπορεί να εκφραστεί ως

$$A_{i+1} = f_i(A_i \bullet W_i + B_i)$$



Σχήμα 2.8: Αρχιτεκτονική νευρωνικού δικτύου ενός επιπέδου

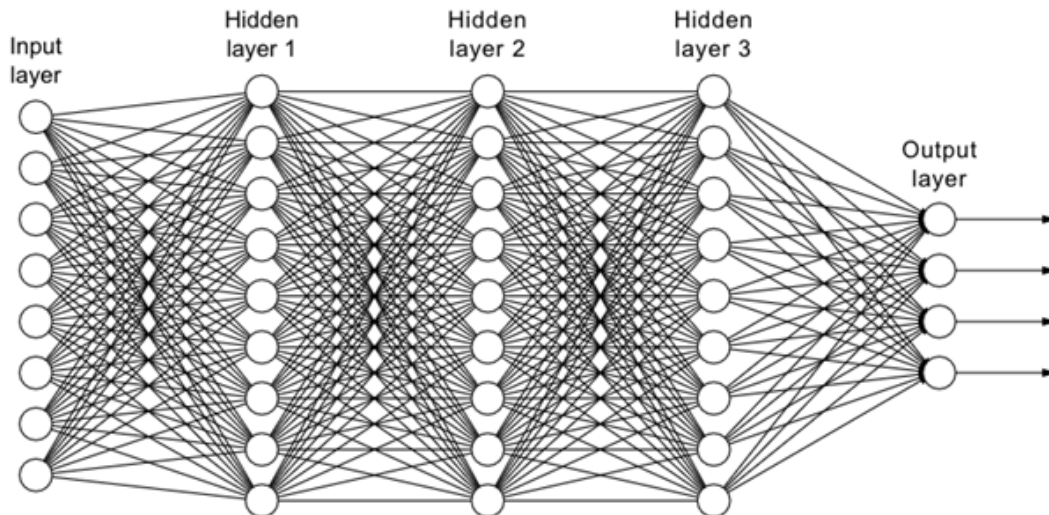
Τα πολυστρωματικά τεχνητά νευρωνικά δίκτυα αποτελούνται από ένα επίπεδο εισόδου, ένα επίπεδο εξόδου και τουλάχιστον ένα κρυφό επίπεδο (hidden layer). Σε κάθε επίπεδο του δικτύου περιέχεται ένας αριθμός τεχνητών νευρώνων. Ο αριθμός των νευρώνων εξόδου αντιστοιχεί στα απαιτούμενα σήματα εξόδου και οι νευρώνες εισόδου είναι τόσοι όσα τα σήματα που επιθυμούμε να επεξεργαστεί το δίκτυο. Από επίπεδο σε επίπεδο, τα σήματα όλων των νευρώνων τροφοδοτούνται προς όλους τους νευρώνες του επόμενου επιπέδου μέσω συνάψεων. Τέλος οι νευρώνες του επιπέδου εξόδου δεν διαθέτουν συνάρτηση ενεργοποίησης.

Το επόμενο διάγραμμα είναι ενδεικτικό της πολυπλοκότητας ενός τεχνητού νευρωνικού δικτύου πολλών επιπέδων.

Συνέλιξη

Η συνέλιξη είναι μια μαθηματική πράξη που εφαρμόζεται σε δύο συναρτήσεις και το αποτέλεσμα της είναι μία τρίτη συνάρτηση

$$(f * g)$$



Σχήμα 2.9: Αρχιτεκτονική πολυεπίπεδου νευρωνικού δικτύου

η οποία εκφράζει το πώς το σχήμα της μιας τροποποιείται από την άλλη. Ο όρος συνέλιξη αναφέρεται τόσο στη συνάρτηση που προκύπτει όσο και στη διαδικασία υπολογισμού της. Ορίζεται ως το ολοκλήρωμα του γινομένου των δύο συναρτήσεων αφού η μία αντιστραφεί και μετατοπιστεί. Το ολοκλήρωμα αξιολογείται για όλες τις τιμές της μετατόπισης, παράγοντας τη συνάρτηση συνέλιξης [25]. Ορίζονται δύο συναρτήσεις, η συνέλιξη διακριτών συναρτήσεων και η συνέλιξη συνεχών συναρτήσεων.

Έστω οι διακριτές συναρτήσεις ϕ , γ . Ορίζεται ως συνέλιξη των ϕ , γ και συμβολίζεται με

$$(\phi * \gamma)$$

η συνάρτηση:

$$(\phi * \gamma)(x) = \sum_{\kappa=-\infty}^{+\infty} \phi(\kappa) \cdot \gamma(x - \kappa) \quad (2.2)$$

Έστω οι συνεχείς συναρτήσεις ϕ , γ . Ορίζεται ως συνέλιξη των ϕ , γ και συμβολίζεται με

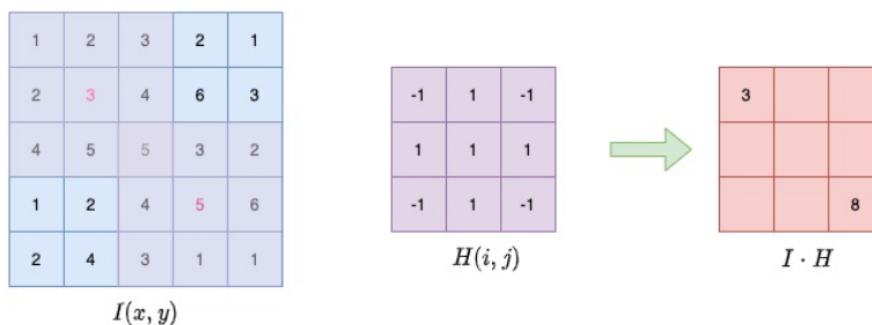
$$\phi * \gamma$$

η συνάρτηση:

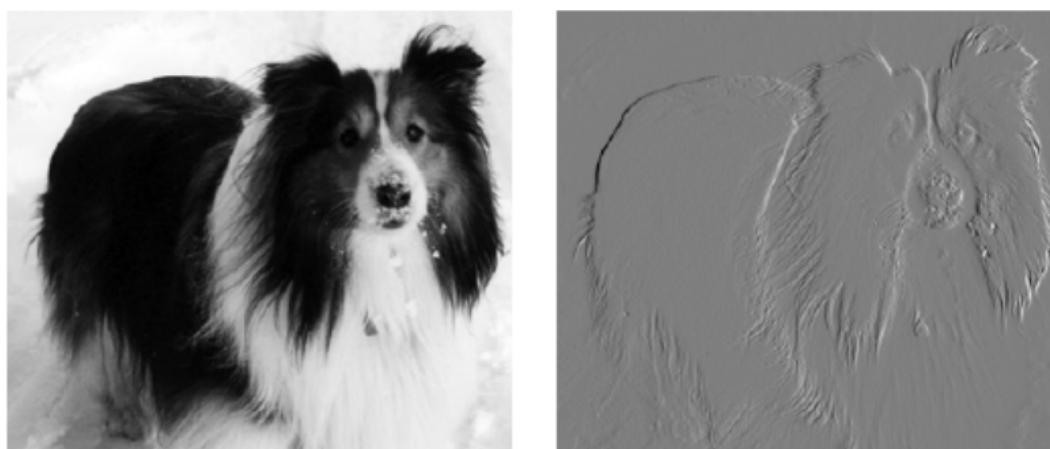
$$(\phi * \gamma)(x) = \int_{-\infty}^{+\infty} \phi(\kappa) \cdot \gamma(x - \kappa) d\kappa \quad (2.3)$$

Η συνέλιξη έχει τις ίδιες ιδιότητες με τον πολλαπλασιασμό. Είναι αντιμεταθετική, προσεταιριστική, επιμεριστική ως προς την πρόσθεση και υπάρχει ένα ουδέτερο στοιχείο η γενικευμένη συνάρτηση $\delta(x)$ (συνάρτηση του Ντιράκ), η οποία μηδενίζεται για κάθε x , εκτός από το σημείο μηδέν όπου τείνει στο θετικό άπειρο.

Έχει χρησιμοποιηθεί αποτελεσματικά στη 2D συνέλιξη, όπου οι δύο συναρτήσεις μπορούν να θεωρηθούν ως η εικόνα και το φίλτρο. Μετά την συνέλιξη έχουμε ως αποτέλεσμα μια άλλη εικόνα που μπορεί να περιέχει ορισμένα χαρακτηριστικά της αρχικής εικόνας με βάση το φίλτρο που χρησιμοποιείται.



Σχήμα 2.10: Απεικόνιση της λειτουργίας συνέλιξης 2D [4]



Σχήμα 2.11: Ανίχνευση ακμών με συνέλιξη [4]

2.2.7 Αναγνώριση Προτύπων

Συνελικτικά Νευρωνικά Δίκτυα

Τα νευρωνικά δίκτυα συνέλιξης (CNN) ανήκουν στην ευρύτερη οικογένεια των συστημάτων μηχανικής μάθησης και αποτελούν μια ειδική κατηγορία νευρωνικών δικτύων. Πρόκειται για τεχνητά νευρωνικά δίκτυα που χρησιμοποιούνται κυρίως στα προβλήματα της αναγνώρισης και εντοπισμού αντικειμένων σε εικόνες. Αυτή τους η δυνατότητα, οφείλεται αρχικά στην δομή τους, η οποία επιτρέπει την εισαγωγή δεδομένων τριών διαστάσεων και τελικά στην εκπαίδευσή τους, η οποία ξεφεύγει από την κλασική σύνδεση των νευρώνων. Πιο συγκεκριμένα δεν συνδέει κάθε έξοδο με κάθε είσοδο όπως στα απλά νευρωνικά δίκτυα, αλλά δημιουργεί τοπικούς τομείς ανάλυσης, στους οποίους οφείλεται και η ανίχνευση των ιδιαίτερων χαρακτηριστικών των εικόνων.

Κάθε νευρώνας δέχεται ένα σήμα εισόδου, εφαρμόζει μία πράξη εσωτερικού γινομένου σε αυτό, και προαιρετικά εφαρμόζει στο αποτέλεσμα μία μη γραμμική συνάρτηση. Το τελευταίο επίπεδο των CNN είναι πλήρες συνδεδεμένο και έχει μία συνάρτηση σφάλματος. Τα νευρωνικά δίκτυα συνέλιξης μοντελοποιούν μικρά τμήματα πληροφορίας, τα οποία στην συνέχεια ενώνονται για να δημιουργήσουν υψηλότερου επιπέδου πληροφορία. Για παράδειγμα σε ένα μοντέλο CNN το πρώτο επίπεδο προσπαθεί να εντοπίσει ακμές, το δεύτερο επίπεδο παίρνει την πληροφορία των ακμών και προσπαθεί να εντοπίσει περιγράμματα, κτλ.

Τα CNN είναι επίσης γνωστά ως τεχνητά νευρωνικά δίκτυα με αναλλοίωτη μετατόπιση ή αναλλοίωτο χώρο (SIANN). Βασίζονται στην αρχιτεκτονική κοινού-βάρους (shared-weight architecture) των συνελκτικών πυρήνων ή φίλτρων, που ολισθαίνουν κατά μήκος των χαρακτηριστικών εισόδου και παρέχουν αντίστοιχες μεταγλωττισμένες απαντήσεις γνωστές και ως χάρτες χαρακτηριστικών (feature maps). [26]

Τα CNN χρησιμοποιούν σχετικά λίγη προεπεξεργασία σε σύγκριση με άλλους αλγορίθμους ταξινόμησης εικόνων. Αυτό σημαίνει ότι το δίκτυο μαθαίνει να βελτιστοποιεί τα φίλτρα (ή πυρήνες) μέσω αυτοματοποιημένης μάθησης, ενώ στους παραδοσιακούς αλγορίθμους αυτά τα φίλτρα σχεδιάζονται με το χέρι. Αυτή η ανεξαρτησία από την προηγούμενη γνώση και την ανθρώπινη παρέμβαση στην εξαγωγή χαρακτηριστικών είναι ένα σημαντικό πλεονέκτημα. [27]

2.3 Εντοπισμός και Αναγνώριση Αντικειμένων

2.3.1 Εντοπισμός Αντικειμένων (Object Detection)

Ο εντοπισμός ή ανίχνευση αντικειμένων, η οποία αποτελεί υποσύνολο της αναγνώρισης αντικειμένων, είναι η διαδικασία εύρεσης περιπτώσεων αντικειμένων μιας συγκεκριμένης κατηγορίας σε εικόνες, όπως ζώα, άνθρωποι ή αυτοκίνητα. Έχει πολλές εφαρμογές σε διάφορους τομείς της υπολογιστικής όρασης, όπως η ανίχνευση προσώπων (face detection), τα βίντεο παρακολούθησης (video surveillance), ανάκτηση εικόνων (image retrieval) και η παρακολούθηση της κίνησης αντικειμένων (tracking movement of objects). Η βασική της ιδέα βασίζεται στο γεγονός ότι κάθε κλάση αντικειμένων έχει τα δικά της μοναδικά χαρακτηριστικά που βοηθούν στην ταξινόμηση της συγκεκριμένης κατηγορίας. Για παράδειγμα, όλοι οι κύκλοι είναι στρογγυλοί, πράγμα που σημαίνει ότι κατά την αναζήτηση κύκλων, τα αντικείμενα που έχουν σταθερή απόσταση από ένα σταθερό σημείο (κέντρο), αναζητούνται σε μια εικόνα. Με τον ίδιο τρόπο, τα τετράγωνα έχουν κάθετες γωνίες με ίσες πλευρές σε μήκος. Επομένως, όταν επιχειρείται η ανίχνευση τετράγωνων αντικειμένων, αναζητούνται τέτοια χαρακτηριστικά.

2.3.2 Αναγνώριση Αντικειμένων (Object Recognition)

Η αναγνώριση αντικειμένων είναι ένας τομέας της τεχνολογίας της όρασης υπολογιστών που επικεντρώνεται στην εύρεση και αναγνώριση αντικειμένων σε μια εικόνα ή σε ένα βίντεο. Οι άνθρωποι κοιτάζοντας μια συγκεκριμένη εικόνα ή βίντεο, μπορούμε εύκολα να εντοπίσουμε τους ανθρώπους, τα αντικείμενα, να καταλάβουμε το σκηνικό και να προσέξουμε άλλες οπτικές λεπτομέρειες. Ο στόχος της αναγνώρισης αντικειμένων είναι να μάθουμε στους υπολογιστές να κάνουν αυτό που σε εμάς τους ανθρώπους είναι φυσικό, δηλαδή το να κατανοήσουν (σε ένα βαθμό πάντα), το τι περιέχει μία εικόνα.

Αυτή η έννοια, η οποία υφίσταται για περισσότερο από μισή δεκαετία, έχει γίνει πλέον ένα από τα πιο συναρπαστικά πεδία της όρασης υπολογιστών και της τεχνητής νοημοσύνης. Η ανάπτυξη των αρχιτεκτονικών των νευρωνικών δικτύων συνελκτικού τύπου (CNN), που υποστηρίζονται από μεγάλα δεδομένα εκπαίδευσης (dataset) και η προηγμένη υπολογιστική τεχνολογία, έχουν δώσει τη δυνατότητα στους υπολογιστές να αναγνωρίζουν αμέσως όλα τα αντικείμενα σε μια σκηνή. Αυτή η δυνατότητα έχει πάρα πολλές εφαρμογές, από την ανίχνευση καρκίνου και την ανίχνευση προσώπων μέχρι τα αυτοματοποιημένα οχήματα κ.λ.π. Δεν επικεντρώνεται μόνο στην

αναγνώριση αντικειμένων σε εικόνες, αλλά και στον ακριβή εντοπισμό τους, δηλαδή στην εύρεση της ακριβούς θέσης τους στην εικόνα ή στο καρέ του βίντεο. Αυτό επιτρέπει την αναγνώριση και τον εντοπισμό πολλαπλών αντικειμένων στην ίδια εικόνα.

2.3.3 Τεχνικές αναγνώρισης αντικειμένων

Οι περισσότερες μέθοδοι που χρησιμοποιούνται στον τομέα της αναγνώρισης και της ανίχνευσης αντικειμένων βασίζονται είτε στη μηχανική μάθηση, είτε στη βαθιά μάθηση. Όπως είναι λογικό και στις δύο τεχνικές ο αλγόριθμος μαθαίνει να αναγνωρίζει αντικείμενα σε εικόνες, ωστόσο, διαφέρουν ως προς τον τρόπο εκτέλεσής τους. Συνοπτικά θα μπορούσαμε να πούμε ότι στην μηχανική μάθηση εφαρμόζονται μέθοδοι ταξινόμησης (classification) προκειμένου να καθοριστούν ειδικά χαρακτηριστικά των αντικειμένων, ενώ στη βαθιά μάθηση περιλαμβάνονται τεχνικές που συνήθως βασίζονται σε συνελκτικά νευρωνικά δίκτυα (CNN) και είναι σε θέση να ανιχνεύσουν αντικείμενα end-to-end, ενώ καθορίζουν μόνες τους τα ειδικά χαρακτηριστικά.

2.3.4 Αναγνώριση αντικειμένων με χρήση τεχνικών μηχανικής μάθησης

Οι τεχνικές μηχανικής μάθησης για την αναγνώριση αντικειμένων προσφέρουν διαφορετικές προσεγγίσεις από τη βαθιά μάθηση. Συγκεκριμένα, για την εκτέλεση της αναγνώρισης αντικειμένων με μηχανική μάθηση, είναι σημαντικό να υπάρχει ένα σύνολο δεδομένων (dataset) με εικόνες ή βίντεο και όλα τα σχετικά χαρακτηριστικά τους.

Στην συνέχεια, θα μπορούσε για παράδειγμα να τρέξει ένας αλγόριθμος εξαγωγής χαρακτηριστικών ώστε να εξάγει χαρακτηριστικά ακμών ή γωνιών που θα μπορούσαν να χρησιμοποιηθούν για την διάκριση των αντικειμένων μεταξύ των κλάσεων που δίνονται. Αυτά τα εξαγόμενα χαρακτηριστικά προστίθενται στο μοντέλο μηχανικής μάθησης, το οποίο πλέον θα μπορεί να μάθει ταξινομεί τα αντικείμενα σε κατηγορίες, χρησιμοποιώντας αυτά τα χαρακτηριστικά με τη χρήση κατάλληλων αλγορίθμων. Τέτοιοι αλγόριθμοι είναι οι supported vector machines (SVM), k-nearest neighbors, ο Naive Bayes κ.α.

Οι πληροφορίες αυτές θα χρησιμοποιηθούν κατά την ανάλυση και την ταξινόμηση νέων αντικειμένων. Υπάρχει μια ποικιλία αλγορίθμων μηχανικής μάθησης και μεθόδων εξαγωγής χαρακτηριστικών, οι οποίες προσφέρουν πολλούς διαφορετικούς συνδυασμούς με σκοπό τη δημιουργία ενός ακριβούς μοντέλου αναγνώρισης αντικειμένων, το οποίο θα μπορεί να επιτύχει ακριβή αποτελέσματα με ελάχιστα δεδομένα. Αυτό κάνει τις μεθόδους μηχανικής μάθησης αρκετά ευέλικτες ώστε να μπορεί να επιλέγεται ο καλύτερος συνδυασμός χαρακτηριστικών και ταξινομιτών για να επιτευχθεί η μάθηση.

2.3.5 Αναγνώριση αντικειμένων με χρήση τεχνικών βαθιάς μάθησης

Οι τεχνικές βαθιάς μάθησης, όπως τα CNN, χρησιμοποιούνται για την αυτόματη εξαγωγή χαρακτηριστικών προκειμένου να αναγνωρίσουν ένα αντικείμενο. Για παράδειγμα, ένα CNN μπορεί να μάθει να αναγνωρίζει τις διαφορές μεταξύ γατών και σκύλων αναλύοντας χιλιάδες εικόνες εκπαίδευσης και μαθαίνοντας τα χαρακτηριστικά που διακρίνουν τις γάτες από τους σκύλους.

Προκειμένου να πραγματοποιηθεί η αναγνώριση αντικειμένων με τη χρήση ενός μοντέλου βαθιάς μάθησης, είναι υποχρεωτική η εκπαίδευση αυτού του μοντέλου σε ένα σύνολο δεδομένων. Το μοντέλο μπορεί είτε να εκπαιδευτεί από το μηδέν, είτε μπορεί να χρησιμοποιηθεί ένα προ-εκπαιδευμένο μοντέλο βαθιάς μάθησης. Στην πρώτη περίπτωση, ένα πολύ μεγάλο σύνολο δεδομένων πρέπει να συγκεντρωθεί και να σχεδιαστεί η αρχιτεκτονική του μοντέλου. Πιο συγκεκριμένα, θα πρέπει να εξεταστεί ένα πολύ ευρύ σύνολο επισημασμένων δεδομένων, προκειμένου να συμπεριληφθεί ένα εύρος εικόνων με διαφορετικά μεγέθη, στάσεις και τύπους αντικειμένων γιατί το δείγμα θα πρέπει να είναι επαρκές και ικανοποιητικό. Στην συνέχεια θα πρέπει να οριστεί η αρχιτεκτονική δικτύου, όπως τα βάρη και τα στρώματα. Μία τέτοια προσέγγιση μπορεί να παράγει εντυπωσιακά αποτελέσματα, αλλά απαιτεί μεγάλο όγκο δεδομένων.

Η επιλογή μεταξύ των προσεγγίσεων που βασίζονται στη μηχανική μάθηση και τη βαθιά μάθηση εξαρτάται σε μεγάλο βαθμό από το είδος του προβλήματος που πρέπει να επιλυθεί και από τη διαθέσιμη υπολογιστική ισχύ. Σε πολλές περιπτώσεις, η μηχανική μάθηση μπορεί να είναι αρκετά αποτελεσματική, και ιδίως όταν τα χαρακτηριστικά της εικόνας είναι αρκετά για τη διάκριση αντικειμένων διαφορετικών κλάσεων. Από την άλλη για την επιλογή της βαθιάς μάθησης θα πρέπει να υπάρχει αρκετός αριθμός από επισημασμένα χαρακτηριστικά και αρκετή υπολογιστική ισχύς, δεδομένου του ότι συμβάλλει σε σημαντική μείωση του χρόνου εκπαίδευσης.

2.4 Ανίχνευση Προσώπου

Η ανίχνευση προσώπου (Face Detection) είναι η διαδικασία μέσω της οποίας ένας αλγόριθμος αναλύει ψηφιακές εικόνες και ανιχνεύει τις περιοχές που περιέχουν ανθρώπινα πρόσωπα, συμπεριλαμβάνοντας το μέγεθος και τη θέση κάθε προσώπου. Πρόκειται για ένα από τα πιο ερευνημένα προβλήματα όρασης υπολογιστών, με τεράστιο πλήθος εγγράφων.

Πολλές προσπάθειες έχουν γίνει για την επίλυση αυτού του προβλήματος κατά τη διάρκεια της ιστορίας [28] [29] [30] [31] [32]. Μερικές από αυτές θα συζητηθούν συνοπτικά στις παρακάτω ενότητες.

Η ανίχνευση προσώπου συνεπάγεται τον προσδιορισμό της παρουσίας ενός ανθρώπινου προσώπου μεταξύ άλλων αντικειμένων σε μια δεδομένη εικόνα ή σε ένα καρέ βίντεο. Αυτό το βήμα είναι συνήθως βήμα προεπεξεργασίας πριν από την λειτουργία της αναγνώρισης προσώπου.

2.4.1 Single Shot Detection

Οι αρχικές προσπάθειες επίλυσης του προβλήματος ανίχνευσης με τη χρήση CNN δεν ήταν πολύ αποτελεσματικές [30]. Οι περισσότερες από αυτές ήταν «Νευρωνικά δίκτυα με βάση την περιοχή» (R-CNN) [33]. Κάτι τέτοιο απαιτούσε την εφαρμογή ενός αλγορίθμου, σε μια εικόνα εισόδου, πολλές φορές και σε πολλές διαφορετικές θέσεις. Συνοπτικά να αναφέρουμε ότι χρησιμοποιείτο ένας αλγόριθμος με σκοπό να βρει και να προβλέψει τις περιοχές ενώ τα πλαίσια υποθέτονταν. Στη συνέχεια αντιστοιχίζονταν, ένας χάρτης χαρακτηριστικών σε κάθε πλαίσιο και ακολουθούσε μια κατάταξη ώστε να ανατίθεται η κατηγορία του αντικειμένου στο πλαίσιο. Αυτή η προσέγγιση απαιτούσε μια μεγάλη σειρά εργασιών (pipeline) και επομένως ήταν

πολύ χρονοβόρα. Για παράδειγμα, σύμφωνα με τον W. Liu στο [30], ακόμη και ο αλγόριθμος Faster R-CNN [34], που ήταν η βελτιωμένη έκδοση του R-CNN, παρουσιάζει επτά fps σε GPU και επομένως δεν μπορεί να χρησιμοποιηθεί για την ανίχνευση αντικειμένων σε πραγματικό χρόνο.

Για τους παραπάνω λόγους δοκιμάστηκε μία νέα προσέγγιση που ονομάστηκε "Single Shot Detection"(SSD) [30], όπου μια εικόνα εξετάζεται μόνο μία φορά από τον αλγόριθμο. Για παράδειγμα, το "You only look once"(YOLO) [31] είναι μια από τις μεθόδους στις οποίες έχει χρησιμοποιηθεί αυτή η τεχνική. Σύμφωνα με τον Redmon, η YOLO επιτυγχάνει 45 καρέ ανά δευτερόλεπτο, ενώ υπερτερεί έναντι της μεθόδου R-CNN [33] και άρα και όλων των προηγούμενων.

Ο "Single Shot Multibox Detector" [30] είναι μια άλλη γνωστή εφαρμογή που χρησιμοποιεί αυτόν τον αλγόριθμο όπου μόνο το τελευταίο μέρος του δικτύου διαφέρει από την YOLO. Σύμφωνα με τον Wei Liu, όπως αναφέρεται στο [30], ο SSD αποδίδει ακόμη καλύτερα από άποψη ακρίβειας σε σύγκριση με τον YOLO, και έχει επιτύχει αποδεδειγμένα αποτελέσματα για ακόμη μικρότερα μεγέθη εικόνας στην είσοδο. Ο αλγόριθμος SSD αποδεικνύεται ως ζωτικής σημασίας και για αυτό οι ερευνητές έχουν καταβάλει μεγάλη προσπάθεια για την περαιτέρω βελτίωση του.

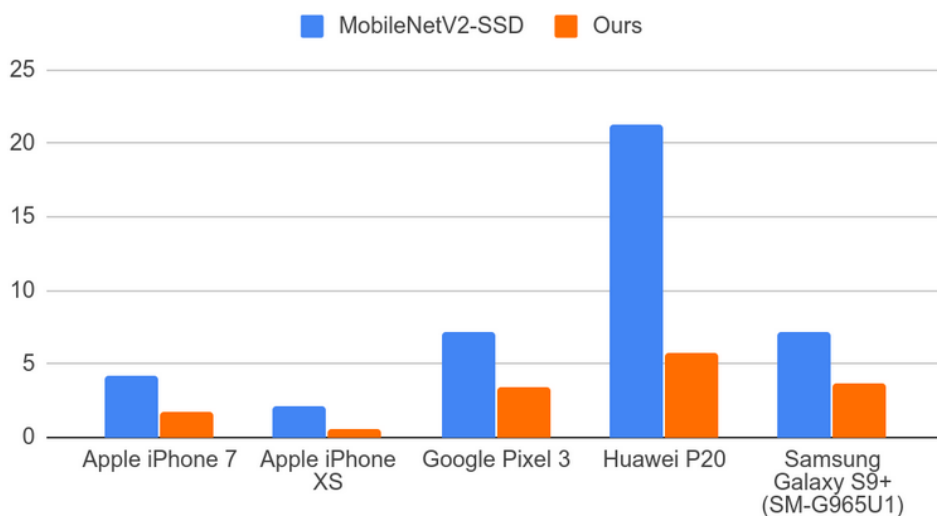
Ένα παράδειγμα αλγορίθμου "Single Shot Multibox Detector" είναι ο MobileNetV2-SSD ο οποίος χρησιμοποιεί σαν εξαγωγέα χαρακτηριστικών το μοντέλο MobileNetV2 [35].

2.4.2 BlazeFace

Ο BlazeFace [32] είναι ένας άλλος αλγόριθμος που αναπτύχθηκε από μια ερευνητική ομάδα της Google και σχεδιάστηκε ειδικά για να χειρίζεται την ανίχνευση προσώπων σε βίντεο σε πραγματικό χρόνο χρησιμοποιώντας GPU κινητών. Έχει αναπτυχθεί πάνω στο μοντέλο SSD, αλλά έχουν γίνει ορισμένες αξιοσημείωτες βελτιώσεις, όπως το αυξημένο μέγεθος πυρήνα και έναν καλύτερο εξαγωγέα χαρακτηριστικών με μια συγκεκριμένη μονάδα συνέλιξης που ονομάζεται Blazeblock.

Όπως φαίνεται στις παρακάτω εικόνες ο BlazeFace αποδίδει πολύ καλύτερα από τον αλγόριθμο MobileNetV2-SSD και έτσι είναι ακόμα πιο κατάλληλος για ανίχνευση προσώπου σε πραγματικό χρόνο.

Latency: MobileNetV2-SSD vs BlazeFace, ms



Σχήμα 2.12: Σύγκριση καθυστέρησης μεταξύ MobileNetV2 και BlazeFace

Device	MobileNetV2-SSD, ms	Ours, ms
Apple iPhone 7	4.2	1.8
Apple iPhone XS	2.1	0.6
Google Pixel 3	7.2	3.4
Huawei P20	21.3	5.8
Samsung Galaxy S9+ (SM-G965U1)	7.2	3.7

Σχήμα 2.13: Απόκριση MobileNetV2 σε διαφορετικές κινητές συσκευές

2.5 Αναγνώριση Προσώπου

Οι εφαρμογές αναγνώρισης προσώπου (Face recognition) χρησιμοποιούνται αρκετά συχνά σήμερα και είναι πάρα πολύ σημαντικές. Πολλά νέα προϊόντα λογισμικού σχεδιάζονται με τη χρήση εφαρμογών αναγνώρισης προσώπου και κάτι τέτοιο είναι απαραίτητο για τη γενικότερη ασφάλεια αλλά και για την ταυτοποίηση του χρήστη μίας εφαρμογής. Η αναγνώριση προσώπου είναι πολύ διαδεδομένη και είναι ένας τομέας που έχει μελετηθεί αρκετά έχοντας ωστόσο αρκετά περιθώρια εξέλιξης, δεδομένου των δυσκολιών που παρουσιάζονται λόγω των μεταβαλλομένων συνθηκών. Τέτοιες συνθήκες είναι η ποσότητα του φωτός, η προοπτική, ο εξοπλισμός που μπορεί να χρησιμοποιείται (γυαλιά καπέλο κασκόλ κλπ), ο χρόνος, η γήρανση, τα συναισθήματα και κατ'επέκταση οι πολλαπλές διαφορετικές εκφράσεις.

Σε ένα επιτυχημένο μοντέλο αναγνώρισης προσώπου παρά την πολυδιάστατη και πολύπλοκη δομή του προσώπου, θα πρέπει να καταφέρνουμε να το αναγνωρίζουμε με σαφήνεια και να το διακρίνουμε μεταξύ των αντικειμένων σε κάθε σε κάθε εικόνα που εξετάζουμε. Η πρόβλεψη θα πρέπει να είναι ακριβής ακόμα και αν αλλάζουν οι

υπό μεταβολή συνθήκες.

Κατά την αναγνώριση προσώπου ένα διάνυσμα χαρακτηριστικών που παράγεται από την εικόνα προσώπου συγκρίνεται με ένα σύνολο διανυσμάτων χαρακτηριστικών που είναι αποθηκευμένα σε μια βάση δεδομένων και επιλέγεται η εγγραφή με τη μεγαλύτερη αντιστοιχία. Χρησιμοποιούνται διάφοροι αλγόριθμοι για τη δημιουργία αυτού του διανύσματος χαρακτηριστικών που περιγράφει το πρόσωπο, όπως επίσης, χρησιμοποιούνται και διάφορες στρατηγικές για την εύρεση της εγγραφής που ταιριάζει.

Ανίχνευση Εκφράσεων Προσώπου

Παρακάτω παρουσιάζονται συνοπτικά διάφορα μοντέλα που χρησιμοποιούνται για τον εντοπισμό προσώπου:

2.5.1 FaceNet

Το FaceNet(FN) [36] είναι ένα project που υλοποιήθηκε από μια ερευνητική ομάδα της Google και παρέχει ένα ολοκληρωμένο framework για την ανάπτυξη εφαρμογών όπως η αναγνώριση προσώπου και η επαλήθευση προσώπων. Απλώς αντιστοιχίζει μια εικόνα προσώπου σε ένα διάνυσμα χαρακτηριστικών, γνωστό και ως ενσωμάτωση (embedding) στον ευκλείδειο χώρο. Η απόσταση μεταξύ δύο ενσωματώσεων αντιπροσωπεύει άμεσα την ομοιότητα μεταξύ δύο προσώπων.

Κατά την εκπαίδευση του δικτύου χρησιμοποίησαν μια συνάρτηση απώλειας που ονομάζεται "Triplet Loss" [36], για να ελαχιστοποιήσουν την απόσταση μεταξύ εικόνων του ίδιου προσώπου και να μεγιστοποιήσουν την απόσταση μεταξύ εικόνων διαφορετικών προσώπων. Σύμφωνα με τους συγγραφείς, αυτός ο αλγόριθμος έχει δείξει εξαιρετικά ποσοστά ακρίβειας με γνωστά dataset προσώπων.

2.5.2 MobileNets

Τα MobileNets [37] είναι lite νευρωνικά δίκτυα ειδικά σχεδιασμένα για κινητές και ενσωματωμένες συσκευές με εφαρμογές όρασης. Πρόκειται για ένα από τα πρωτοποριακά έργα που πραγματοποιήθηκαν στον τομέα αυτό από μια ερευνητική ομάδα της Google. Το MobileNet χρησιμοποιεί τη διαχωρίσιμη κατά βάθος συνέλιξη [37] αντί της τυπικής συνέλιξης για τη δημιουργία της αρχιτεκτονικής του. Αυτό μειώνει δραστικά το κόστος υπολογισμού καθώς και το μέγεθος του μοντέλου και παρέχει φιλικότητα προς τα κινητά. Παρέχει επίσης δύο υπερπαραμέτρους ως «πολλαπλασιαστή πλάτους» και «πολλαπλασιαστή ανάλυσης» για καλύτερη ακρίβεια και μικρότερη καθυστέρηση του μοντέλου. Όπως αναφέρουν οι συγγραφείς, τα μοντέλα αποδίδουν αξιοπρεπή ακρίβεια για γνωστά dataset. Το MobileNet εκπαιδεύτηκε για εργασίες αναγνώρισης προσώπου χρησιμοποιώντας knowledge distillation από το FaceNet [36] ως βασικό μοντέλο. Οι συγγραφείς δήλωσαν ότι η ακρίβεια είναι ελαφρώς πιο χαμηλή σε σύγκριση με το μοντέλο FaceNet.

2.5.3 MobileFaceNet

Όπως υποδηλώνει το όνομά του, το MobileFaceNet(MFN) [38] έχει σχεδιαστεί με σκοπό την επίτευξη υψηλότερης απόδοσης και ακρίβειας για την επαλήθευση προσώπου σε κινητές και ενσωματωμένες συσκευές. Όπως περιγράφεται στο [38], μια

κινεζική ερευνητική ομάδα πρότεινε αυτό το μοντέλο το οποίο σύμφωνα με τον Shen Chen, διορθώνει κοινά προβλήματα που συναντώνται σε παρόμοια μοντέλα όπως το FaceNet [38] και το MobileNetV2 [37] και έχει σχεδιαστεί ειδικά για επαληθεύσεις προσώπων σε πραγματικό χρόνο. Αυτό το μοντέλο δεν δίνει ίδια πιθανότητα μεταξύ των περιφερειακών και των κεντρικών pixel στην εικόνα εισόδου, δεδομένου ότι δεν συμβάλλουν το ίδιο στην λειτουργία της αναγνώρισης. Το μέγεθος των διανυσμάτων εξόδου έχει επίσης μειωθεί κατά 50% σε σύγκριση με τα άλλα μοντέλα. Όλοι αυτοί οι παράγοντες έκαναν αυτό το μοντέλο μοναδικό.

2.6 Μάθηση μέσω Κινητών Συσκευών

Η μάθηση μέσω κινητών συσκευών (Mobile Learning ή πολλές φορές και mLearning) μπορεί να οριστεί απλά, ως κάθε μορφή μάθησης που αξιοποιεί τις δυνατότητες που προσφέρουν οι κινητές και ασύρματες τεχνολογίες και συσκευές [39].

Πρόκειται για έναν τρόπο πρόσβασης στο εκπαιδευτικό περιεχόμενο μέσω κινητών συσκευών, ενισχύοντας τη μάθηση και επιτρέποντας στους χρήστες να έχουν πρόσβαση στο εκπαιδευτικό περιεχόμενο όπου και όποτε το επιθυμούν.

Αυτό είναι και το πιο σημαντικό χαρακτηριστικό της κινητής μάθησης, δηλαδή το ότι ο εκπαιδευόμενος έχει τη δυνατότητα να κινείται και με αυτόν τον τρόπο να μπορεί να επιλέξει πότε και πού θέλει να έχει πρόσβαση στο περιεχόμενο της μάθησης. Αυτό συνεπάγεται ότι μπορεί να ορίσει τον δικό του ρυθμό, αυξάνοντας έτσι τη δέσμευση στο αποτέλεσμα και βελτιώνοντας κατά αυτόν τον τρόπο τη διατήρηση της γνώσης.

2.6.1 Η εξέλιξη της κινητής μάθησης

Η εξέλιξη της κινητής μάθησης μπορεί να διαχωριστεί στις παρακάτω αλληλοκαλυπτόμενες φάσεις [40]:

- Εστίαση στις συσκευές
- Εστίαση στην μάθηση εκτός τάξης
- Εστίαση στην κινητικότητα του χρήστη

Οι κινητές συσκευές μέσω της εξέλιξης της τεχνολογίας συντελούν στην προαγωγή της γνώσης, των ικανοτήτων και των προοπτικών όσον αφορά τη μάθηση στον εικοστό πρώτο αιώνα [41].

Υπάρχουν πολλαπλά προγράμματα και σεμινάρια m-learning με σκοπό την προαγωγή γνώσης των μαθητών, την ανάπτυξη της κριτικής ικανότητας και την καλλιέργεια συνεργατικού και ομαδικού πνεύματος τόσο εντός όσο και εκτός της τάξης. Είναι δεδομένο ότι πλέον η πλειοψηφία των μαθητών διαθέτει κινητό τηλέφωνο σύγχρονης τεχνολογίας και αξιοποιεί τις παιδαγωγικές δυνατότητες που τους προσφέρει. Επί προσθέτως σε πολλές χώρες χρηματοδοτήθηκαν προγράμματα με στόχο την αξιοποίηση της μάθησης μέσω κινητού τηλεφώνου στα εκπαιδευτικά ιδρύματα.

Παρά το γεγονός ότι στο εξωτερικό έχει δοθεί ιδιαίτερη έμφαση στην εκπαίδευση μέσω elearning και m-learning, στην Ελλάδα δεν υπήρξε αντίστοιχη εξέλιξη. Αυτό μπορεί να οφείλεται είτε στους ελλειπείς πόρους και υποδομές, είτε στην δυσπιστία των εκπαιδευτικών οργανισμών. Πάντως με αφορμή την επέκταση της πανδημίας και

η Ελλάδα έκανε ένα μεγάλο βήμα προς τον εκσυγχρονισμό της εκπαίδευσης, εισάγοντας την τηλεεκπαίδευση, την κινητή μάθηση με σύγχρονο ή ασύγχρονο τρόπο.

2.7 API

2.7.1 Google Machine Learning Kit

Το machine learning kit είναι ένα mobile sdk που έχει κατασκευάσει η Google και δίνει την δυνατότητα στους προγραμματιστές να χρησιμοποιήσουν μηχανική μάθηση σε συσκευές κινητών με android και ios με μεγάλη ευκολία [42]. Έτσι μονάχα με μερικές γραμμές κώδικα κάποιος μπορεί να εκμεταλλευτεί αυτές τις δυνατότητες ακόμα και αν γνωρίζει ελάχιστα τα σχετικά με το μοντέλο βελτιστοποίησης και τα νευρωνικά δίκτυα. Το ml kit παρέχει ένα API το οποίο επιτρέπει τη χρήση μοντέλων tensor flow light έτσι ώστε να μπορεί να συνδυαστεί με τεχνικές μηχανικής μάθησης χρησιμοποιώντας την προηγούμενη βιβλιοθήκη σε ένα sdk.

Το ML Kit διευκολύνει την εφαρμογή τεχνικών ML στις εφαρμογές, συγκεντρώνοντας τις τεχνολογίες ML της Google, όπως το Google Cloud Vision API, το TensorFlow Lite και το Android Neural Networks API σε ένα ενιαίο SDK.

Μπορεί να λειτουργήσει είτε κατευθείαν χρησιμοποιώντας πόρους και κάνοντας την επεξεργασία στο Cloud της Google, είτε μέσω κάποιου έτοιμου μοντέλου κατάλληλα βελτιστοποιημένου για κινητά τηλέφωνα παρέχοντας έτσι real-time αποτελέσματα, είτε με την ευελιξία των προσαρμοσμένων μοντέλων TensorFlow Lite.

Διάφορες επιλογές που δίνει το machine-learning kit της Google είναι οι παρακάτω:

- Αναγνώριση κειμένου
- Αναγνώριση προσώπου
- Pose Detection
- Barcode scannin
- Image labeling
- Αναγνώριση αντικειμένων και εντοπισμός
- Αναγνώριση γλώσσας
- Μετάφραση
- Έξυπνη απάντηση
- Entity extraction
- Cloud AutoML Vision Edge
- Custom Models

2.7.2 Google Activity Recognition

Οι κινητές συσκευές έχουν γίνει για πολλούς ένα πανταχού παρόν μέρος της καθημερινής ζωής. Οι χρήστες έχουν τα κινητά τηλέφωνα μαζί τους καθ' όλη τη διάρκεια της ημέρας καθώς οδηγούν, περπατούν, ασκούνται, εργάζονται ή παίζουν. Η κατανόηση του τι κάνουν οι χρήστες στον φυσικό κόσμο επιτρέπει σε μία εφαρμογή να είναι πιο έξυπνη ως προς τον τρόπο αλληλεπίδρασης μαζί τους. Για παράδειγμα,

μια εφαρμογή μπορεί να αρχίσει να παρακολουθεί τους καρδιακούς παλμούς ενός χρήστη όταν αρχίζει να τρέχει, μια άλλη εφαρμογή μπορεί να μεταβεί σε λειτουργία αυτοκινήτου όταν ανιχνεύσει ότι ο χρήστης έχει αρχίσει να οδηγεί.

Το API αναγνώρισης δραστηριοτήτων [43] βασίζεται στους αισθητήρες που είναι διαθέσιμοι σε μια συσκευή. Οι αισθητήρες της συσκευής παρέχουν πληροφορίες σχετικά με το τι κάνουν αυτή τη στιγμή οι χρήστες. Ωστόσο, με δεκάδες σήματα από πολλαπλούς αισθητήρες και μικρές παραλλαγές στον τρόπο που οι άνθρωποι κάνουν πράγματα, η ανίχνευση του τι κάνουν οι χρήστες δεν είναι εύκολη.

Το API αναγνώρισης δραστηριοτήτων ανιχνεύει αυτόματα τις δραστηριότητες διαβάζοντας περιοδικά σύντομες εκρήξεις (short bursts) δεδομένων αισθητήρων και τα επεξεργάζονται με τη χρήση μοντέλων μηχανικής μάθησης. Για τη βελτιστοποίηση των πόρων, το API μπορεί να σταματήσει να αναφέρει την δραστηριότητα αν η συσκευή παραμείνει ακίνητη για αρκετό διάστημα. Τότε θα αρχίσει να χρησιμοποιεί αισθητήρες χαμηλής κατανάλωσης ενέργειας και μόλις ανιχνεύσει κίνηση θα συνεχίσει να αναφέρει την δραστηριότητα.

2.7.3 Google Fused Location Provider

Οι εφαρμογές μπορούν να επωφεληθούν από τα σήματα που παρέχουν οι πολλαπλοί αισθητήρες της συσκευής για να προσδιορίσουν τη θέση της συσκευής. Ωστόσο, η επιλογή του σωστού συνδυασμού σημάτων για μια συγκεκριμένη εργασία σε διαφορετικές συνθήκες δεν είναι απλή. Η εύρεση μιας λύσης που να είναι επίσης αποδοτική ως προς την μπαταρία είναι ακόμη πιο περίπλοκη.

Ο fused location provider είναι ένα API τοποθεσίας στις υπηρεσίες του Google Play που συνδυάζει έξυπνα διαφορετικά σήματα για να παρέχει τις πληροφορίες τοποθεσίας που χρειάζεται η εφαρμογή σας. [44]

Το fused location provider διαχειρίζεται τις υποκείμενες τεχνολογίες τοποθεσίας, όπως το GPS και το Wi-Fi, και παρέχει ένα απλό API που μπορείτε να χρησιμοποιήσετε για να καθορίσετε την απαιτούμενη ποιότητα υπηρεσίας. Για παράδειγμα, μπορείτε να ζητήσετε τα πιο ακριβή διαθέσιμα δεδομένα ή την καλύτερη δυνατή ακρίβεια χωρίς πρόσθετη κατανάλωση ενέργειας.

2.7.4 Firebase

Η Firebase είναι ένα σύνολο εργαλείων που βοηθάει στην ανάπτυξη εφαρμογών τα οποία καλύπτουν ένα μεγάλο μέρος υπηρεσιών και λειτουργιών που οι προγραμματιστές θα έπρεπε κανονικά να κατασκευάσουν οι ίδιοι. Η Google τα παρέχει out of the box ώστε να μπορούν οι προγραμματιστές να επικεντρωθούν στην ανάπτυξη της εφαρμογής. [45] Τέτοιες λειτουργίες είναι τα analytics, το authentication, οι βάσεις δεδομένων, το configuration, η αποθήκευση αρχείων, η αποστολή μηνυμάτων ειδοποίησης (push messaging) και πολλά άλλα. Όλες αυτές οι υπηρεσίες φιλοξενούνται στο google cloud και παρέχονται στον προγραμματιστή ο οποίος δεν χρειάζεται να καταβάλει σχεδόν καθόλου προσπάθεια για να τις αξιοποιήσει.

Όλες αυτές οι υπηρεσίες συντηρούνται και λειτουργούν πλήρως από την Google. Τα client SDKs που παρέχονται από την Firebase αλληλεπιδρούν απευθείας με αυτές τις backend υπηρεσίες, χωρίς να απαιτείται ενδιάμεσο λογισμικό μεταξύ της εφαρμογής και της υπηρεσίας. Αυτό διαφέρει από την παραδοσιακή ανάπτυξη εφαρμογών, η οποία συνήθως περιλαμβάνει τόσο frontend όσο και backend λογισμικό. Η

Google έχει δημιουργήσει μία ειδικά διαμορφωμένη πλατφόρμα το Firebase Console [46] που παρέχει εύκολη administrative access σε αυτές τις υπηρεσίες.

2.7.5 TensorFlow lite

Το Tensorflow είναι μία από τις πιο διαδεδομένες και ευρέως χρησιμοποιούμενες βιβλιοθήκες στον τομέα της βαθιάς μάθησης η οποία έχει δημιουργηθεί από την Google εδώ και αρκετά χρόνια με πρώτη εμφάνιση της το Νοέμβριο του 2015, δίνοντας τη δυνατότητα για ανάπτυξη εφαρμογών που υποστηρίζουν τεχνητή νοημοσύνη με Deep learning.

Το TensorFlow είναι μία open source deep learning library και χάρη στην ευέλικτη δομή του μας επιτρέπει να αναπτύξουμε εφαρμογές και να κάνουμε υπολογισμούς ανεξαρτήτου πλατφόρμας και λογισμικού χρησιμοποιώντας ένα ή περισσότερα CPUs και GPUs με ένα και μοναδικό API βασισμένο σε python. Το framework υποστηρίζει πολλές διαφορετικές γλώσσες προγραμματισμού όπως οι C++, Java, C#, JavaScript, R και επίσης python.

Τον Μάιο του 2017, η Google ανακοίνωσε ένα νέο software stack, το TensorFlow Lite. Πρόκειται για ένα σύνολο εργαλείων που επιτρέπει τη μηχανική μάθηση on-device, βοηθώντας τους προγραμματιστές να εκτελούν τα μοντέλα τους σε κινητά τηλέφωνα, embedded, και edge συσκευές. [47]

Τα μοντέλα TensorFlow Lite έχουν ένα ειδικό αποδοτικό και φορητό format γνωστό ως FlatBuffers. Το FlatBuffers έχει επέκταση αρχείου .tflite και παρέχει διάφορα πλεονεκτήματα σε σχέση με το πρωτόκολλο που χρησιμοποιεί το TensorFlow, όπως μειωμένο μέγεθος (μικρό αποτύπωμα κώδικα) και ταχύτερη εξαγωγή συμπερασμάτων γιατί τα δεδομένα προσπελάζονται απευθείας, χωρίς το επιπλέον βήμα ανάλυσης/αποσυμπίεσης. Αυτό επιτρέπει στο TensorFlow Lite να εκτελείται αποτελεσματικά σε συσκευές με περιορισμένους υπολογιστικούς πόρους και με χαμηλή μνήμη.

Τα μοντέλα TensorFlow Lite μπορούν προαιρετικά να περιέχουν metadata με μία περιγραφή του μοντέλου αναγνώσιμη από τον άνθρωπο και metadata με δεδομένα αναγνώσιμα από τη μηχανή για την αυτόματη δημιουργία σωληνώσεων πριν και μετά την επεξεργασία κατά την εξαγωγή συμπερασμάτων.

2.7.6 Youtube Data API

Το Youtube Data API επιτρέπει στους προγραμματιστές να έχουν πρόσβαση σε στατιστικά στοιχεία βίντεο και σε δεδομένα καναλιών YouTube μέσω δύο τύπων κλήσεων, REST και XML-RPC. Η Google περιγράφει το Youtube API ως «API και εργαλεία που σας επιτρέπουν να μεταφέρετε την εμπειρία του YouTube στην ιστοσελίδα, την εφαρμογή ή τη συσκευή σας» [48].

Όπως και όλα τα API της Google, γλιτώνει τον προγραμματιστή από πολλές γραμμές κώδικα δίνοντάς του εύκολη πρόσβαση στην τεράστια αυτή βάση δεδομένων του Youtube.

2.7.7 CameraX

Η CameraX είναι μια νέα βιβλιοθήκη Jetpack που επιτρέπει στους προγραμματιστές να ελέγχουν την κάμερα μιας συσκευής και επικεντρώνεται στη συμβατότητα

μεταξύ των συσκευών. Παρέχει ένα συνεπές και εύχρηστο API που λειτουργεί στη συντριπτική πλειοψηφία των συσκευών Android, με συμβατότητα προς τα πίσω μέχρι το Android 5.0 (επίπεδο API 21).

Το να διατηρηθεί η ίδια συμπεριφορά της κάμερας σε διαφορετικές συσκευές είναι κάτι πολύ δύσκολο. Πρέπει να λάβουμε υπόψιν μας την αναλογία διαστάσεων, τον προσανατολισμό, την περιστροφή, το μέγεθος προεπισκόπησης και το μέγεθος της εικόνας. Με την CameraX, απαλλασσόμαστε από αυτά τα προβλήματα μιας και όλες αυτές οι βασικές συμπεριφορές απλά λειτουργούν. [49]

Η CameraX δίνει έμφαση σε συγκεκριμένες εφαρμογές (use cases), ώστε να μπορούμε ως προγραμματιστές να επικεντρωθούμε στην εργασία που πρέπει να κάνουμε αντί στο πώς θα διαχειριζόμαστε την διαφορετικότητα μεταξύ των συσκευών.

Παρακάτω παρουσιάζονται οι use cases που υποστηρίζονται, οι οποίες είναι και οι πιο κοινές εφαρμογές μιας κάμερας:

- **Preview:** Προβολή μιας εικόνας στην οθόνη. Δέχεται ένα αντικείμενο Surface [50] για να προβάλλει ένα preview, όπως για παράδειγμα το PreviewView.
- **Image Analysis:** Μας παρέχονται CPU-accessible buffers ώστε να μπορούμε να τους χρησιμοποιήσουμε στους αλγορίθμους μας για ανάλυση όπως για παράδειγμα στη μηχανική μάθηση μέσω του ML Kit.
- **Image Capture:** Αποθήκευση εικόνων.
- **Video Capture:** Αποθήκευση βίντεο και ήχου.

3 Μεθοδολογία

3.1 Αναγνώριση συναισθήματος

Όπως αναφέραμε και στο κεφάλαιο 2, οι άνθρωποι χρησιμοποιούμε τις εκφράσεις του προσώπου μας για να δείξουμε το πώς αισθανόμαστε, όπως για παράδειγμα χαμογελώντας δείχνουμε ότι είμαστε χαρούμενοι, ενώ συνοφρυώνοντας δείχνουμε ότι είμαστε θυμωμένοι. Ιστορικά, η έρευνα για την όραση των υπολογιστών έχει επικεντρωθεί στην ανάλυση και στην εκμάθηση αυτών των χαρακτηριστικών του προσώπου για την αναγνώριση συναισθημάτων. Ωστόσο, αυτά τα χαρακτηριστικά του προσώπου δεν είναι καθολικά και ποικίλλουν σε μεγάλο βαθμό μεταξύ πολιτισμών και καταστάσεων γεγονός που κάνει την αναζήτηση ικανοποιητικού dataset ακόμα πιο δύσκολη όπως βέβαια και την εκπαίδευση του νευρωνικού δικτύου.

3.1.1 Αλγόριθμος αναγνώρισης συναισθήματος

Στην συγκεκριμένο project ο αλγόριθμος που ακολουθήθηκε για τον εντοπισμό συναισθήματος φαίνεται στην εικόνα 3.1 και εξηγείται αναλυτικά στα επόμενα κεφάλαια.

Συνοπτικά θα μπορούσαμε να πούμε ότι προκειμένου να εντοπίσουμε το συναίσθημα χρησιμοποιούμε την κάμερα του κινητού, η οποία λειτουργεί live, και σε κάθε frame παίρνουμε μία εικόνα. Την εικόνα αυτή την δίνουμε σαν παράμετρο στο FaceDetector του mlkit της Google και μας επιστρέφεται η πληροφορία με το πρόσωπο ή τα πρόσωπα της εικόνας καθώς και τα χαρακτηριστικά τους. Το τελευταίο αντικείμενο με το πρόσωπο (Face Object) το μετατρέπουμε σε εικόνα συγκεκριμένων προδιαγραφών, την οποία εισάγουμε σε ένα ήδη εκπαιδευμένο μοντέλο TensorFlow Lite, για να προβλέψουμε το συναίσθημα.

Emotion Detection Process



Σχήμα 3.1: Αλγόριθμος εντοπισμού συναισθήματος

Παρακάτω εμφανίζονται τα βήματα εκτέλεσης του αλγορίθμου σε high level:

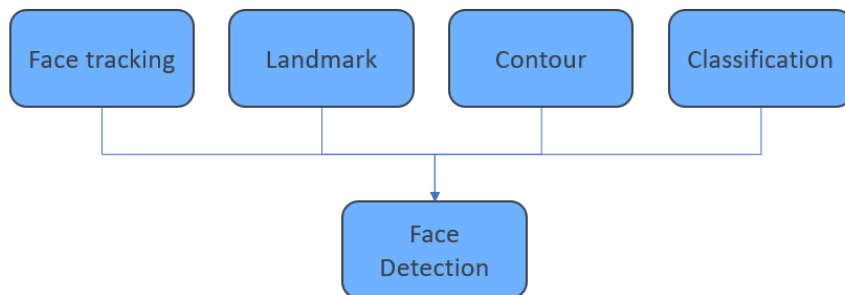
1. Δημιουργία αντικειμένου εικόνας σε κάθε frame του ζωντανού βίντεο που λαμβάνει η κάμερα
2. Μετατροπή της εικόνας σε διαστάσεις 480x360 pixels
3. Εισαγωγή της εικόνας στον ανιχνευτή προσώπου (FaceDetector)
4. Ανίχνευση προσώπου από τον ανιχνευτή και εξαγωγή λίστας με αντικείμενο Face
5. Για κάθε πρόσωπο της λίστας face
 - (α') Αντιγραφή αντικειμένου Face σε τοπικό αντίγραφο
 - (β') Δημιουργία αντικειμένου εικόνας από το αντικείμενο face (που περιέχει μόνο το πρόσωπο) με διαστάσεις 200x200 pixels
 - (γ') Εισαγωγή της εικόνας του προσώπου στο νευρωνικό δίκτυο
 - (δ') Ανίχνευση συναισθήματος για το συγκεκριμένο πρόσωπο
 - (ε') Δημιουργία αντικειμένου Emotion και αποθήκευσή του ως τρέχον συναίσθημα

3.1.2 Εντοπισμός προσώπου

Όπως αναφέραμε και προηγουμένως, για τον εντοπισμό του προσώπου χρησιμοποιήσαμε το ML kit της Google και συγκεκριμένα το Face Detection module. Όπως αναφέρει και το όνομά του το Face Detection Module εντοπίζει ανθρώπινα πρόσωπα

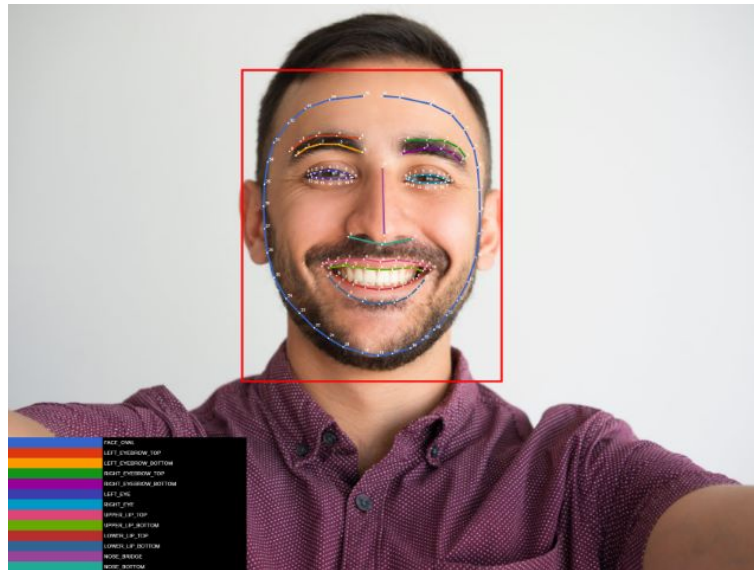
σε οπτικά μέσα, όπως ψηφιακές εικόνες ή βίντεο αλλά μας δίνει και άλλες, πολύ σημαντικές δυνατότητες, που περιγράφουμε παρακάτω σε high level: [51]

1. **Landmark Detection:** Εντοπίζει τα χαρακτηριστικά του προσώπου (μάτια, αυτιά, μάγουλα, μύτη και στόμα)
2. **Face Contour:** Εντοπίζει ένα σύνολο σημείων που ακολουθούν το σχήμα ενός χαρακτηριστικού προσώπου καθώς και του περιγράμματος ενός προσώπου.
3. **Classification:** Εντοπίζει τις εκφράσεις του προσώπου και πιο συγκεκριμένα την πιθανότητα να χαμογελάει ο χρήστης ή να έχει κλειστά τα μάτια του
4. **Face Tracking:** Παρακολουθεί τα πρόσωπα σε ένα βίντεο δίνοντάς σε καθένα από αυτά ένα μοναδικό αναγνωριστικό id έτσι ώστε να μπορούμε να προχωρήσουμε σε λειτουργίες που απαιτούν αναγνώριση προσώπου. Από την στιγμή που θα αναγνωριστεί ένα πρόσωπο, τότε μπορεί να αναγνωριστεί πάλι σε όλες τις εικόνες ή τα βίντεο. (βλ. κεφάλαιο 6)



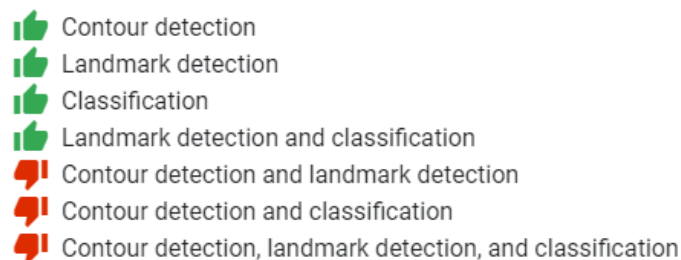
Σχήμα 3.2: Δυνατότητες μοντέλου αναγνώρισης προσώπου ML Kit

Μόλις εντοπιστούν τα πρόσωπα, επιστρέφεται πίσω μία λίστα με αντικείμενα Face, και για κάθε ένα από αυτά μπορούμε να πάρουμε τις συντεταγμένες του περιγράμματός του (bounding box), την εικόνα του καθώς και οποιαδήποτε άλλη πληροφορία από αυτές που αναφέραμε προηγουμένως, εφόσον βέβαια έχουν ζητηθεί ως επιλογές κατά τον ορισμό του αντικειμένου FaceDetector.



Σχήμα 3.3: Παράδειγμα bounding box και face contours κατά τον εντοπισμό προσώπου

Τέλος πρέπει να αναφερθεί ότι το Face Detection module μας δίνει την δυνατότητα να εκτελεσθεί real-time αρκεί να ακολουθηθούν οι κατευθυντήριες γραμμές που δίνονται. [52] Όπως αναγράφεται, για να μπορεί να γίνει real-time ο εντοπισμός προσώπου θα πρέπει να ρυθμιστεί ο ανιχνευτής προσώπου ώστε να χρησιμοποιεί είτε ανίχνευση περιγράμματος προσώπου (face contour detection), είτε ταξινόμηση (classification) και ανίχνευση χαρακτηριστικών (landmark detection), αλλά όχι και τα δύο.



Σχήμα 3.4: Επιλογές διαμόρφωσης για την ανίχνευση προσώπου σε πραγματικό χρόνο

Στο συγκεκριμένο project η διαδικασία εντοπισμού προσώπου γίνεται realtime, λαμβάνοντας εικόνες σε κάθε καρέ της κάμερας και γι αυτό το λόγο ο FaceDetector έχει παραμετροποιηθεί ώστε να κάνει μόνο classification. Οι προδιαγραφές αναφέρουν ότι η εικόνα που χρησιμοποιηθεί στον ανιχνευτή προσώπου πρέπει να έχει διαστάσεις τουλάχιστον 480x360 pixels και ειδικά όταν θέλουμε η ανίχνευση να γίνεται σε πραγματικό χρόνο, τότε η λήψη καρέ σε αυτή την ελάχιστη ανάλυση είναι σχεδόν απαραίτητη για να μειώσει την καθυστέρηση. Γι αυτό το λόγο πριν εισάγουμε την εικόνα στο μοντέλο του FaceDetector. Ο σκοπός μας είναι να χρησιμοποιούνται τα χαρακτηριστικά που εξάγονται ώστε να βγουν κάποια γρήγορα συμπεράσματα (βλ. Analytics).

3.1.3 Αναγνώριση συναισθήματος

Για τον εντοπισμό συναισθήματος χρησιμοποιούμε το αντικείμενο Face που έχει εξάγει ο ανιχνευτής προσώπων από το οποίο παίρνουμε την εικόνα του και την επεξεργαζόμαστε κατάλληλα ώστε να μπορέσουμε να την εισάγουμε στο μοντέλο μας. Το μοντέλο που χρησιμοποιούμε απαιτεί η εικόνα να είναι διαστάσεων 200x200 pixels και συνεπώς κάνουμε μία κατάλληλη προεπεξεργασία πριν το χρησιμοποιήσει το CNN. Το νευρωνικό δίκτυο επιστρέφει ένα διάνυσμα πιθανοτήτων συσχετισμένο με τις ετικέτες των κλάσεων εκπαίδευσης (βλ σχήμα 3.5) το οποίο πρακτικά αποτελεί έναν πίνακα με τα score για τα έξι συναισθήματα που έχουμε ορίσει. Στη συνέχεια επιλέγουμε το συναίσθημα με το μεγαλύτερο score ως το επικρατέστερο χρησιμοποιώντας την υλοποιηθείσα μέθοδο ArgMax η οποία βρίσκει από μία λίστα τον δείκτη που περιέχει την μεγαλύτερη τιμή. Τέλος μέσω μεθόδου, της κλάσης *Emotion*, αντιστοιχίζουμε αυτόν τον δείκτη με την κλάση μας και δημιουργούμε ένα αντικείμενο *Emotion*.

id	Emotion
0	anger
1	disgust
2	fear
3	happy
4	neutral
5	sadness
default	surprised

Σχήμα 3.5: Ετικέτες εκπαίδευσης μοντέλου συναισθήματος

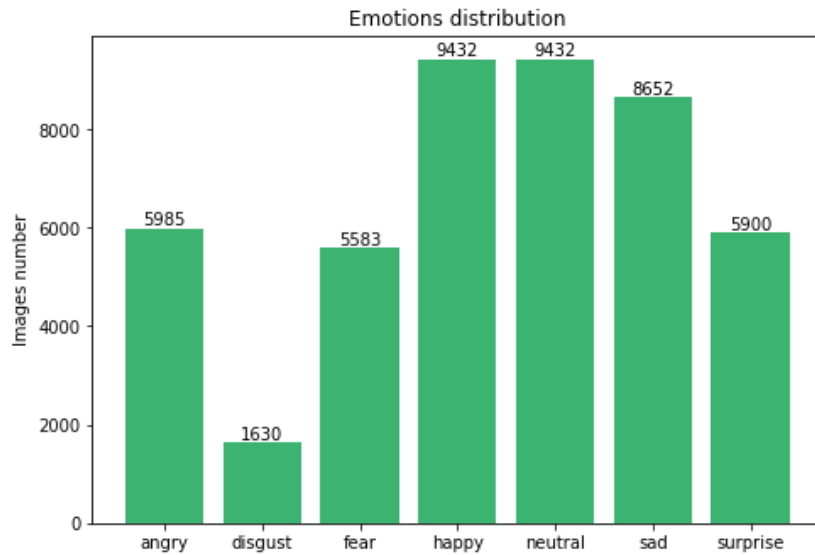
Μοντέλο αναγνώρισης συναισθήματος

Το μοντέλο που χρησιμοποιήσαμε για την αναγνώριση συναισθήματος [5] παρέχεται υπό την άδεια MIT [53] και βασίζεται αρχιτεκτονικά στο MobileNetV2 το οποίο όμως έχει τροποποιηθεί κατάλληλα ώστε να μπορεί να λειτουργήσει σε συσκευές χαμηλών προδιαγραφών.

Τα σύνολα δεδομένων των εικόνων που χρησιμοποιήθηκαν για την εκπαίδευση του CNN είναι τα :

- JAFFE [54]
- CK [55]
- FER2013 [56]
- RAF-DB [57]

Έτσι δημιουργήθηκε ένα υβριδικό dataset με 46614 εικόνες που έχει την παρακάτω κατανομή δεδομένων :



Σχήμα 3.6: Κατανομή συναισθημάτων στο σύνολο δεδομένων [5]

Για την ταξινόμηση των συναισθημάτων του προσώπου η εφαρμογή χρησιμοποιεί εκπαιδευμένο βαθύ νευρωνικό δίκτυο συνελκτικού τύπου deep CNN (`simple_classifier.tflite`). Κάθε pixel του μετατρέπεται από ακέραιο αριθμό που ανήκει στο διάστημα $[0, 255]$ σε δεκαδικό στο διάστημα $[0, 1]$.

Το νευρωνικό δίκτυο έχει την ακόλουθη δομή:

Layer (type)	Output Shape	Param #
conv2d_178 (Conv2D)	(None, 48, 48, 32)	320
conv2d_179 (Conv2D)	(None, 48, 48, 32)	9248
conv2d_180 (Conv2D)	(None, 48, 48, 32)	9248
conv2d_181 (Conv2D)	(None, 48, 48, 32)	9248
max_pooling2d_70 (MaxPooling)	(None, 23, 23, 32)	0
dropout_90 (Dropout)	(None, 23, 23, 32)	0
conv2d_182 (Conv2D)	(None, 23, 23, 64)	18496
conv2d_183 (Conv2D)	(None, 23, 23, 64)	36928
conv2d_184 (Conv2D)	(None, 23, 23, 64)	36928
max_pooling2d_71 (MaxPooling)	(None, 11, 11, 64)	0
dropout_91 (Dropout)	(None, 11, 11, 64)	0
conv2d_185 (Conv2D)	(None, 11, 11, 128)	73856
conv2d_186 (Conv2D)	(None, 11, 11, 128)	147584
max_pooling2d_72 (MaxPooling)	(None, 5, 5, 128)	0
dropout_92 (Dropout)	(None, 5, 5, 128)	0
conv2d_187 (Conv2D)	(None, 5, 5, 256)	295168
max_pooling2d_73 (MaxPooling)	(None, 2, 2, 256)	0
dropout_93 (Dropout)	(None, 2, 2, 256)	0
flatten_18 (Flatten)	(None, 1024)	0
dense_36 (Dense)	(None, 1024)	1049600
dropout_94 (Dropout)	(None, 1024)	0
dense_37 (Dense)	(None, 7)	7175
Total params: 1,693,799		
Trainable params: 1,693,799		
Non-trainable params: 0		

Σχήμα 3.7: Δομή νευρωνικού δικτύου [5]

Το μοντέλο εκπαιδεύτηκε χρησιμοποιώντας τις παρακάτω παραμέτρους εκπαίδευσης:

Parameter	Value
min_delta	0.0001
patience	10
optimizer	Adam
learning_rate	0.0001
loss	categorical_crossentropy
batch_size	96

Σχήμα 3.8: Παράμετροι εκπαίδευσης νευρωνικού δικτύου [5]



Σχήμα 3.9: Κανονικοποιημένος πίνακας σύγκρισης [5]

Metric	Value (Test subset)
Accuracy	0.678
Precision	0.662
F1	0.647

Σχήμα 3.10: Metrics του εκπαιδευμένου μοντέλου [5]

3.2 Αναγνώριση σκηνής

Η αναγνώριση σκηνών είναι ένας ταχέως αναπτυσσόμενος τομέας που έχει προσελκύσει σημαντική προσοχή τα τελευταία χρόνια. Είναι ένα δύσκολο πρόβλημα που απαιτεί κατάλληλες προσεγγίσεις για έχουμε τα επιθυμητά αποτελέσματα. Και σε αυτόν τον τομέα τα συνελκτικά νευρωνικά δίκτυα έχουν αποδείξει τη χρησιμότητα τους και χρησιμοποιούνται ευρέως [58]. Είναι ένα απαραίτητο βήμα για διάφορες εφαρμογές όπως η πλοήγηση ρομπότ, η κατασκευή χαρτών κ.λπ

3.2.1 Αλγόριθμος αναγνώρισης σκηνής

Η διαδικασία της αναγνώρισης σκηνής ξεκινάει παράλληλα με τη διαδικασία αναγνώρισης συναισθήματος σε κάποιο άλλο thread και τα αποτελέσματα τους παρέχονται στην κύρια εφαρμογή ασύγχρονα. Η λειτουργία σε υψηλό επίπεδο περιγράφεται στην παρακάτω εικόνα, ενώ περισσότερες λεπτομέρειες υλοποίησης δίνονται στο κεφάλαιο 4.

Συνοπτικά θα μπορούσαμε να αναφέρουμε ότι ο αλγόριθμος τίθεται σε λειτουργία μόλις εντοπιστεί κάποιο πρόσωπο στη φωτογραφία δηλαδή σε συνέχεια του βήματος 4 του αλγορίθμου αναγνώρισης προσώπου και όχι ανεξάρτητα από αυτόν. Επιλέξαμε μία τέτοια προσέγγιση αφενός διότι τα analytics έχουν νόημα για την εφαρμογή μας μόνο όταν ο χρήστης είναι ενεργός και αφετέρου διότι σκοπός μας είναι όλα τα δεδομένα να συνδέονται κάτω από ένα μοναδικό αναγνωριστικό το οποίο θα είναι το id του λογαριασμού σύνδεσης. Ένα ακόμα πλεονέκτημα αυτής της επιλογής, είναι ότι μπορεί να χρησιμοποιείται το ίδιο αντικείμενο εικόνας που δόθηκε στον ανιχνευτή προσώπου. Το αντικείμενο Image που παρέχεται στην είσοδο του αλγορίθμου μετατρέπεται σε εικόνα μεγέθους 224x224 pixel και στη συνέχεια εισάγεται σε ένα ήδη εκπαιδευμένο μοντέλο TensorFlow Lite [6] για να εντοπίσουμε τη σκηνή.

Παρακάτω εμφανίζονται τα βήματα εκτέλεσης του αλγορίθμου σε υψηλό επίπεδο. Για την διευκόλυνση της κατανόησης επαναλαμβάνονται τα προηγούμενα βήματα που είναι κοινά με τον αλγόριθμο αναγνώρισης προσώπου:

1. Δημιουργία αντικειμένου εικόνας σε κάθε frame του live video που λαμβάνει η κάμερα
2. Μετατροπή της εικόνας σε διαστάσεις 480x360 pixels
3. Εισαγωγή της εικόνας στον ανιχνευτή προσώπου (FaceDetector)
4. Ανίχνευση προσώπου από τον ανιχνευτή και εξαγωγή λίστας με αντικειμένου Face
5. Βρόγχος επανάληψης
 - (α) Δημιουργία αντικειμένου εικόνας από το αρχικό αντικείμενο εικόνας με διαστάσεις 224x224 pixels
 - (β) Εισαγωγή της εικόνας στο νευρωνικό δίκτυο
 - (γ) Ανίχνευση σκηνής
 - (δ) Δημιουργία αντικειμένου Scenery και αποθήκευσή του ως τρέχουσα σκηνή

3.2.2 Μοντέλο αναγνώρισης σκηνής

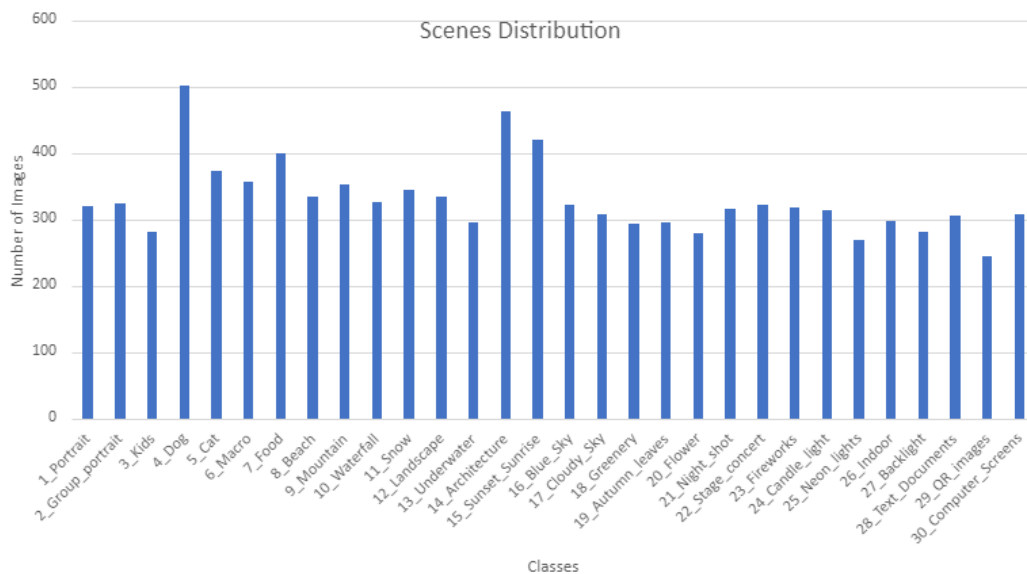
Η διαδικασία αναγνώρισης σκηνής εκτελείται τοπικά σε πραγματικό χρόνο και επιστρέφει ένα διάνυσμα πιθανοτήτων των κλάσεων εκπαίδευσης. Όπως και στη διαδικασία αναγνώρισης προσώπου χρησιμοποιούμε την συνάρτηση ArgMax για να βρούμε το στοιχείο με το επικρατέστερο score που είναι η τρέχουσα σκηνή μας.

Το συγκεκριμένο μοντέλο μπορεί να αναγνωρίσει την περιβάλλουσα σκηνή σε μία εικόνα. Το κατεβάσαμε από το TensorFlow Hub [6] και βασίζεται αρχιτεκτονικά στο EfficientNet-B1.

Οι σκηνές που ανιχνεύει φαίνονται στην παρακάτω εικόνα :



Σχήμα 3.11: Ετικέτες εκπαίδευσης μοντέλου αναγνώρισης σκηνής [6]



Σχήμα 3.12: Κατανομή δεδομένων εκπαίδευσης μοντέλου αναγνώρισης σκηνής [6]

Το μοντέλο παρέχεται για λήψη σε μορφή tflite υπό την άδεια Creative Commons Attribution 3.0 License [59] αλλά δίνεται επίσης και στην πλατφόρμα Google Colab [60] της Google υπό την άδεια Apache-2.0 [61]. Μέσω του google colab μπορούν οι προγραμματιστές χρησιμοποιώντας απλώς ένα web browser, να δουν τον κώδικα που χρησιμοποιήθηκε για την εκπαίδευση του νευρωνικού δικτύου και είτε να τον

τρέξουν είτε να τον τροποποιήσουν αν το επιθυμούν αλλάζοντας τις παραμέτρους εκπαίδευσης.

3.3 Αναγνώριση δραστηριότητας

Η αναγνώριση δραστηριότητας είναι μία σημαντική πληροφορία ειδικά όταν αναφερόμαστε σε μία εφαρμογή μηχανικής μάθησης, διότι ο χρήστης μπορεί να βρίσκεται οπουδήποτε κατά την εκπαιδευτική διαδικασία και μία τέτοια πληροφορία μπορεί να κρίνει ενδεχομένως την ποιότητα της μάθησης του χρήστη.

Στην εφαρμογής μας, βρίσκουμε τη δραστηριότητα «ρωτώντας» το Google Activity Recognition API. [43] Η απάντηση περιλαμβάνει μια λίστα με τις εντοπισμένες δραστηριότητες, καθεμία από τις οποίες περιέχει τον τύπο της δραστηριότητας και την «εμπιστοσύνη».

Η *εμπιστοσύνη* υποδεικνύει την πιθανότητα που υπάρχει ώστε ο χρήστης να εκτελεί την συγκεκριμένη δραστηριότητα. Ο τύπος αφορά μία γενικότερη κατηγορία που αντιπροσωπεύει την ανιχνευμένη δραστηριότητα της συσκευής σε σχέση με οντότητες του φυσικού κόσμου, για παράδειγμα, η συσκευή βρίσκεται πάνω σε ένα ποδήλατο ή η συσκευή βρίσκεται πάνω σε έναν χρήστη που τρέχει.

Οι τύποι δραστηριότητας που μπορεί να αναγνωρίσει το συγκεκριμένο API φαίνονται στην παρακάτω εικόνα :

int	IN_VEHICLE	The device is in a vehicle, such as a car.
int	ON_BICYCLE	The device is on a bicycle.
int	ON_FOOT	The device is on a user who is walking or running.
int	RUNNING	The device is on a user who is running.
int	STILL	The device is still (not moving).
int	TILTING	The device angle relative to gravity changed significantly.
int	UNKNOWN	Unable to detect the current activity.
int	WALKING	The device is on a user who is walking.

Σχήμα 3.13: Κατανομή δεδομένων εκπαίδευσης μοντέλου αναγνώρισης σκηνής [6]

Με αυτόν τον τρόπο, απλά ρωτώντας το Google Activity Recognition API, μπορούμε να γνωρίζουμε τη δραστηριότητα του χρήστη χωρίς να χρειάζεται να διαβάσουμε τα δεδομένα από τους αισθητήρες του τηλεφώνου και να τα αναλύσουμε προσπαθώντας να βγάλουμε συμπέρασμα.

Για να το επιτύχουμε αυτό και να έχουμε πάντα ενημερωμένη την πληροφορία της δραστηριότητας, φροντίζουμε να εκτελείται στο παρασκήνιο μία διεργασία που θα επικοινωνεί με το συγκεκριμένο API. Τεχνικές λεπτομέρειες θα αναφερθούν στο κεφάλαιο της υλοποίησης.

3.4 Εντοπισμός τοποθεσίας

Χρησιμοποιώντας το Fused Location Provider API [44] που αναφέραμε στη θεωρία, η εφαρμογή μας μπορεί να ζητήσει την τελευταία γνωστή τοποθεσία της συσκευής

ής του χρήστη. Η λήψη της τελευταίας γνωστής τοποθεσίας είναι συνήθως ένα καλό σημείο εκκίνησης για όσες εφαρμογές χρειάζονται πληροφορίες τοποθεσίας.

Η εφαρμογή EMORICON καταγράφει αυτήν την πληροφορία μαζί με τα υπόλοιπα analytics γιατί τι θεωρούμε σημαντική όχι μόνο όσον αφορά τα μαζικά δεδομένα αλλά και ώστε μελλοντικά να παρέχει δυνατότητες εξατομίκευσης της εκπαίδευσης του χρήστη (βλ επεκτάσεις).

3.5 Επισήμανση Στοιχείων Εικόνας και Εντοπισμός Αντικειμένων

Η επισήμανση εικόνας και ο εντοπισμός αντικειμένων προσφέρουν αρκετή πληροφορία στα αναλυτές της εφαρμογής μας. Για να επιτύχουμε τις συγκεκριμένες λειτουργίες χρησιμοποιούμε το ML Kit της Google και πιο συγκεκριμένα τα Image labeling [62] και Object detection and tracking [7].

Θα μπορούσε κάποιος να υποθέσει ότι τα API Image Labeling και Object Detection κάνουν το ίδιο πράγμα, κάτι το οποίο ισχύει ως ένα βαθμό μιας και υπάρχει μια μικρή επικάλυψη της λειτουργικότητάς τους. Παρόλα αυτά χρησιμοποιούνται για διαφορετικούς λόγους και αυτό διότι τα μοντέλα έχουν εκπαιδευτεί με διαφορετικό τρόπο προσφέροντας διαφορετικές δυνατότητες.

Η επισήμανση στοιχείων εικόνας (image labeling) είναι μια διαδικασία αναγνώρισης αντικειμένων και είναι παρόμοια με τον εντοπισμό αντικειμένων (object detection), ωστόσο υπάρχει μια λεπτή αλλά σημαντική διαφορά μεταξύ τους. Στην επισήμανση στοιχείων εικόνας, το μοντέλο τεχνητής νοημοσύνης αναγνωρίζει κάθε «αξιόλογο» αντικείμενο στην εικόνα ή το βίντεο και επιστρέφει μία ετικέτα για κάθε ένα από αυτά, ενώ στον εντοπισμό αντικειμένων παράλληλα με την εικόνα θα μπορούσε να δίνεται και η κλάση του αντικειμένου και να αναγνωρίζονται όλες οι εμφανίσεις του, επιστρέφοντας την ακριβή τοποθεσία του στην εικόνα και ενδεχομένως περισσότερες λεπτομέρειες για αυτό.

Στην εφαρμογή emoricon είναι σκοπός μας να μαζεύουμε αρκετή πληροφορία, ώστε να μπορεί να χρησιμοποιηθεί μέσω της εφαρμογής των Analytics είτε για διαφημιστικούς σκοπούς, είτε για να εξατομικεύσει και να βελτιώσει την λειτουργικότητα εφαρμογών.

3.5.1 Επισήμανση στοιχείων εικόνας (Image Labeling)

Η επισήμανση εικόνων αποτελεί βασικό στοιχείο για την ανάπτυξη μοντέλων μηχανικής μάθησης με επίβλεψη ειδικά όταν πρόκειται να έχουν τη δυνατότητα όρασης υπολογιστών (computer vision). Βοηθά στην εκπαίδευση μοντέλων μηχανικής μάθησης για την επισήμανση ολόκληρων εικόνων ή τον εντοπισμό κλάσεων αντικειμένων εντός μιας εικόνας. Οι ετικέτες που επιστρέφονται, καθιστούν τις εικόνες αναγνώσιμες από τις μηχανές. Ως γνωστόν οι αλγόριθμοι ταξινόμησης εικόνων λαμβάνουν εικόνες ως είσοδο και είναι σε θέση να τις ταξινομήσουν αυτόματα σε μία από τις διάφορες ετικέτες (γνωστές και ως κλάσεις). Για παράδειγμα, ένας αλγόριθμος μπορεί να είναι σε θέση να ταξινομήσει εικόνες οχημάτων σε ετικέτες όπως «αυτοκίνητο», «τρένο» ή «πλοίο». Σε ορισμένες περιπτώσεις, η ίδια εικόνα μπορεί να έχει πολλαπλές ετικέτες.

Στην εφαρμογή μας χρησιμοποιούμε το ML Kit Image Labeling το οποίο επιστρέφει ταχύτατα μία λίστα με τα αντικείμενα που αναγνωρίστηκαν στην εικόνα. Αυτά θα μπορούσαν είτε να χρησιμοποιηθούν για εκπαίδευση μοντέλων, είτε ως

είσοδος στο API object detection and tracking, είτε τελικά για analytics συγκεκριμένου σκοπού που συνήθως είναι απαραίτητα για διαφημιστικούς λόγους.

Το ML Kit εξάγει τις ετικέτες από το μοντέλο TensorFlow Lite και τις παρέχει ως περιγραφή κειμένου. Είναι ένας ισχυρός βασικός ταξινομητής γενικής χρήσης που δύναται να αναγνωρίζει περισσότερες από 400 κατηγορίες που περιγράφουν τα πιο συχνά αναγνωρίσιμα αντικείμενα που βρίσκονται στις φωτογραφίες. Μπορεί να προσαρμοστεί ώστε να χρησιμοποιηθεί με άλλα προεκπαιδευμένα μοντέλα από το TensorFlow Hub ή κάποιο δικό μας προσαρμοσμένο μοντέλο που έχει εκπαιδευτεί με το TensorFlow, το AutoML Vision Edge ή το TensorFlow Lite Model maker.

Τέλος, στο site της Google αναφέρεται χαρακτηριστικά «Να σημειωθεί ότι αυτό το API προορίζεται για μοντέλα ταξινόμησης εικόνων που περιγράφουν την πλήρη εικόνα. Για την ταξινόμηση ενός ή περισσότερων αντικειμένων σε μια εικόνα, όπως παπούτσια ή έπιπλα, το API Object Detection and Tracking μπορεί να είναι καλύτερο» [62].

3.5.2 Αλγόριθμος επισήμανσης εικόνων

Η συγκεκριμένη διαδικασία ξεκινάει παράλληλα με την διαδικασία αναγνώρισης συναισθήματος σε κάποιο άλλο thread. Χρησιμοποιούμε το αντικείμενο Image της εικόνας που έχουμε συλλέξει από το καρέ της κάμερας και το παρέχουμε σαν είσοδο στο μοντέλο που έχει κατεβάσει το ML Kit Image Labeling στη συσκευή μας. Μόλις ολοκληρωθεί η διαδικασία μας επιστρέφεται ως έξοδος μία λίστα με ετικέτες αναγνωρισμένων αντικειμένων.

Παρακάτω εμφανίζονται τα βήματα εκτέλεσης του αλγορίθμου σε υψηλό επίπεδο. Για διευκόλυνση της κατανόησης επαναλαμβάνονται τα προηγούμενα βήματα που είναι κοινά με τον αλγόριθμο αναγνώρισης προσώπου :

1. Δημιουργία αντικειμένου εικόνας σε κάθε frame του live video που λαμβάνει η κάμερα
2. Μετατροπή της εικόνας σε διαστάσεις 480x360 pixels
3. Εισαγωγή της εικόνας στον ανιχνευτή προσώπου (FaceDetector)
4. Ανίχνευση προσώπου από τον ανιχνευτή και εξαγωγή λίστας αντικειμένου Face
5. Δημιουργία αντικειμένου εικόνας από το αρχικό αντικείμενο εικόνας με διαστάσεις 224x224 pixels
6. Εισαγωγή της εικόνας στο νευρωνικό δίκτυο
7. Επισήμανση Δεδομένων Εικόνας
8. Δημιουργία λίστας με τις ετικέτες των αναγνωρισμένων αντικειμένων


3.5.3 Εντοπισμός αντικειμένων (Object Detection)

Για τον εντοπισμό των αντικειμένων εικόνας χρησιμοποιούμε το Google ML Kit και συγκεκριμένα το Object Detection and Tracking [7].

Ένας αλγόριθμος ανίχνευσης αντικειμένων έχει ως σκοπό την ανίχνευση ενός αντικειμένου σε μια εικόνα και τη θέση του στο πλαίσιο της εικόνας. Η θέση των αντικειμένων ορίζεται συνήθως με τη χρήση οριοθετούμενων πλαισίων (bounding boxes). Ένα bounding box όπως αναφέρθηκε και στην περίπτωση του face detection, είναι το μικρότερο ορθογώνιο που περιέχει ολόκληρο το αντικείμενο στην εικόνα. Από τε-

χνικής άποψης, είναι ένα σύνολο τεσσάρων συντεταγμένων, που αντιστοιχίζονται σε μια ετικέτα η οποία καθορίζει την κατηγορία του αντικειμένου. Οι συντεταγμένες των bounding boxes και οι ετικέτες τους αποθηκεύονται συνήθως σε ένα αρχείο JSON, χρησιμοποιώντας dictionary format. Ο αριθμός ή το αναγνωριστικό της εικόνας id, είναι το κλειδί της στο dictionary file.

Όταν εισάγουμε μια εικόνα στο ML Kit, αυτό ανιχνεύει έως και πέντε αντικείμενα στην εικόνα μαζί με τη θέση κάθε αντικειμένου στην εικόνα. Κατά τον εντοπισμό αντικειμένων σε video streams, κάθε αντικείμενο έχει ένα μοναδικό αναγνωριστικό που μπορεί να χρησιμοποιηθεί για την παρακολούθηση (tracking) του αντικειμένου από καρτέ σε καρτέ. Επίσης προαιρετικά μπορεί να ενεργοποιηθεί η επιλογή coarse object classification, η οποία επισημαίνει τα αντικείμενα με περιγραφές ευρείας κατηγορίας.



Tracking ID	0
Bounds	(95, 45), (496, 45), (496, 240), (95, 240)
Category	PLACE
Classification confidence	0.9296875

Σχήμα 3.14: Παράδειγμα αναγνωρισμένου αντικειμένου [7]

3.5.4 Αλγόριθμος εντοπισμού αντικειμένων

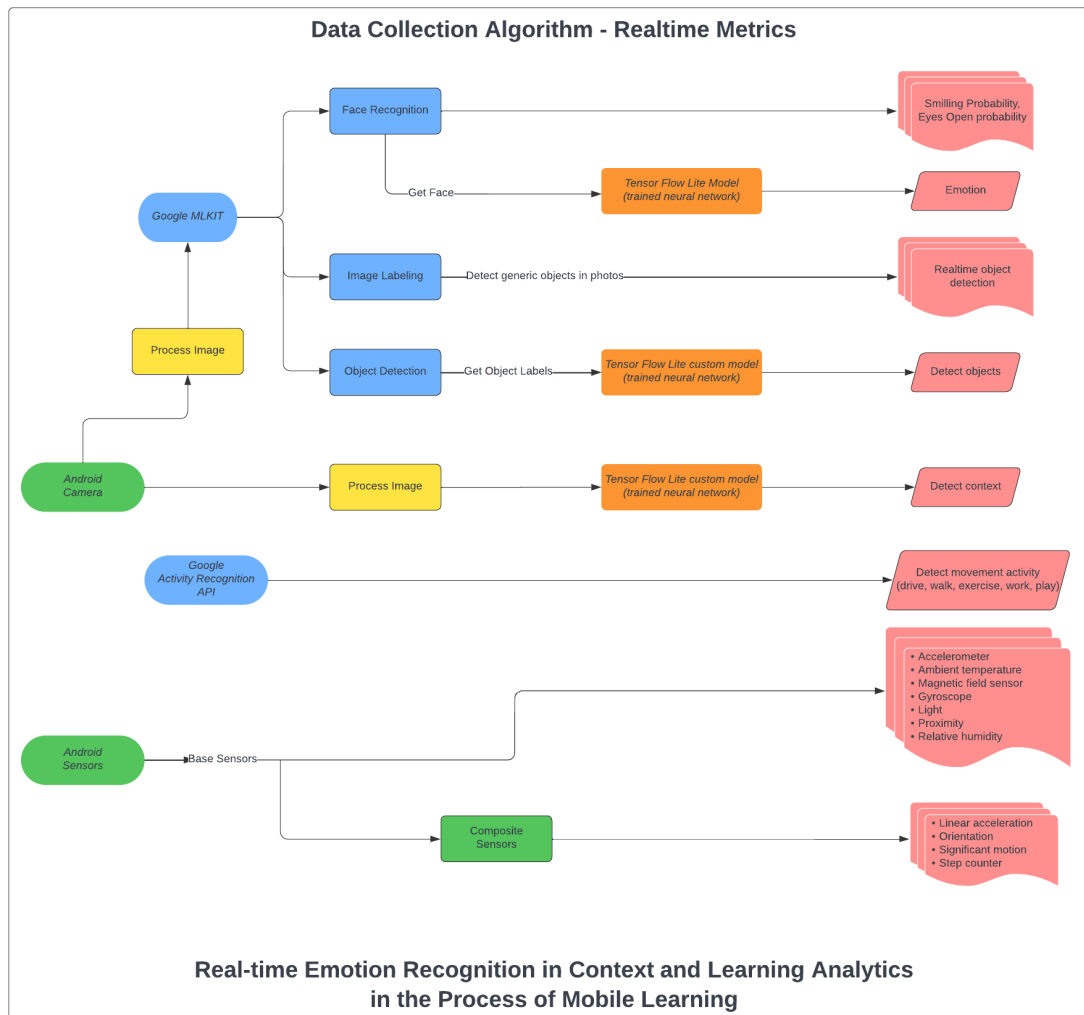
Ο αλγόριθμος επισημάνσης στοιχείων εικόνας εκτελείται επίσης παράλληλα με τους άλλους αλγορίθμους, εισάγοντας στο μοντέλο το ίδιο αντικείμενο object.

Τα βήματα εκτέλεσης του αλγορίθμου είναι τα παρακάτω:

1. Δημιουργία αντικειμένου εικόνας σε κάθε frame του live video που λαμβάνει η κάμερα
2. Μετατροπή της εικόνας σε διαστάσεις 480x360 pixels
3. Εισαγωγή της εικόνας στον ανιχνευτή προσώπου (FaceDetector)
4. Ανίχνευση προσώπου από τον ανιχνευτή και εξαγωγή λίστας με αντικειμένου Face
5. Εισαγωγή του αντικειμένου εικόνας στο νευρωνικό δίκτυο
6. Αναγνώριση και εντοπισμός αντικειμένων στην εικόνα
7. Δημιουργία λίστας τα αντικείμενα DetectedObject που επιστρέφει ο ταξινομητής

3.6 Αλγόριθμος Συλλογής Δεδομένων

Σε αυτό το κεφάλαιο παρουσιάζεται συγκεντρωτικά ο αλγόριθμος της εφαρμογής (βλ. σχήμα 3.15) για την συλλογή των δεδομένων που χρησιμοποιούνται-παρέχονται για analytics. Σκοπός είναι να μαζέψουμε όσα περισσότερα δεδομένα μπορούμε ώστε να διευκολύνουμε και να καθοδηγήσουμε τη διαδικασία της μάθησης μέσω κινητού αλλά και μέσα από τις κλάσεις που παρέχουμε να μπορούν αυτά μελλοντικά να χρησιμοποιηθούν και για άλλες χρήσεις.



Σχήμα 3.15: Αλγόριθμος συλλογής δεδομένων

Είναι σημαντικό να αναφέρουμε ότι το σχήμα 3.15, δεν δείχνει τη χρονική εκτέλεση των μεθόδων αλλά πρακτικά χρησιμοποιείται για να κατηγοριοποιήσει τους αλγόριθμους Είναι διακριτό από την εικόνα που προηγείται ότι η συλλογή των δεδομένων έχει ουσιαστικά δύο πηγές εισόδου, την κάμερα του κινητού και το γκρουπ των αισθητήρων που είναι built-in σχεδόν σε όλες τις κινητές συσκευές. Είναι λογικό να γεννηθεί το ερώτημα γιατί αναφερόμαστε στους αισθητήρες ομαδοποιώντας τους δεδομένου ότι κάθε ένας λειτουργεί διαφορετικά και παρέχει διαφορετικού τύπου πληροφορίες. Η απάντηση στο προηγούμενο ερώτημα έγκειται στον τρόπο που τους διαχειρίζεται η πλατφόρμα του Android.

3.6.1 Αισθητήρες Android

Οι περισσότερες συσκευές με Android διαθέτουν ενσωματωμένους αισθητήρες που μετρούν την κίνηση, τον προσανατολισμό και διάφορες περιβαλλοντικές συνθήκες. Αυτοί οι αισθητήρες είναι σε θέση να παρέχουν ακατέργαστα δεδομένα με μεγάλη ακρίβεια και είναι χρήσιμοι εάν θέλουμε να παρακολουθούμε την τριδιάστατη κίνηση της συσκευής ή το πώς είναι τοποθετημένη ή ακόμα και εάν θέλουμε να παρακολουθούμε τις αλλαγές στο περιβάλλον κοντά σε μια συσκευή.

Η πλατφόρμα Android υποστηρίζει τρεις μεγάλες κατηγορίες αισθητήρων :

- **Αισθητήρες κίνησης:** Αυτοί οι αισθητήρες μετρούν δυνάμεις επιτάχυνσης και περιστροφικές δυνάμεις κατά μήκος τριών αξόνων. Αυτή η κατηγορία περιλαμβάνει επιταχυνσιόμετρα, αισθητήρες βαρύτητας, γυροσκόπια και αισθητήρες διανυσμάτων περιστροφής.
- **Αισθητήρες περιβάλλοντος:** Αυτοί οι αισθητήρες μετρούν διάφορες περιβαλλοντικές παραμέτρους, όπως η θερμοκρασία, η πίεση του αέρα, ο φωτισμός και η υγρασία του περιβάλλοντος. Η κατηγορία αυτή περιλαμβάνει βαρόμετρα, φωτόμετρα και θερμομέτρα.
- **Αισθητήρες θέσης:** Αυτοί οι αισθητήρες μετρούν τη φυσική θέση μιας συσκευής. Η κατηγορία αυτή περιλαμβάνει αισθητήρες προσανατολισμού και μαγνητόμετρα.

Μπορούμε να αποκτήσουμε πρόσβαση στους αισθητήρες που είναι διαθέσιμοι στη συσκευή και να αποκτήσουμε raw sensor data μέσω χρησιμοποιώντας το Android Sensor Framework. Το framework αυτό μας παρέχει διάφορες classes και interfaces που μας βοηθούν να εκτελέσουμε μια μεγάλη ποικιλία εργασιών που σχετίζονται με αισθητήρες.

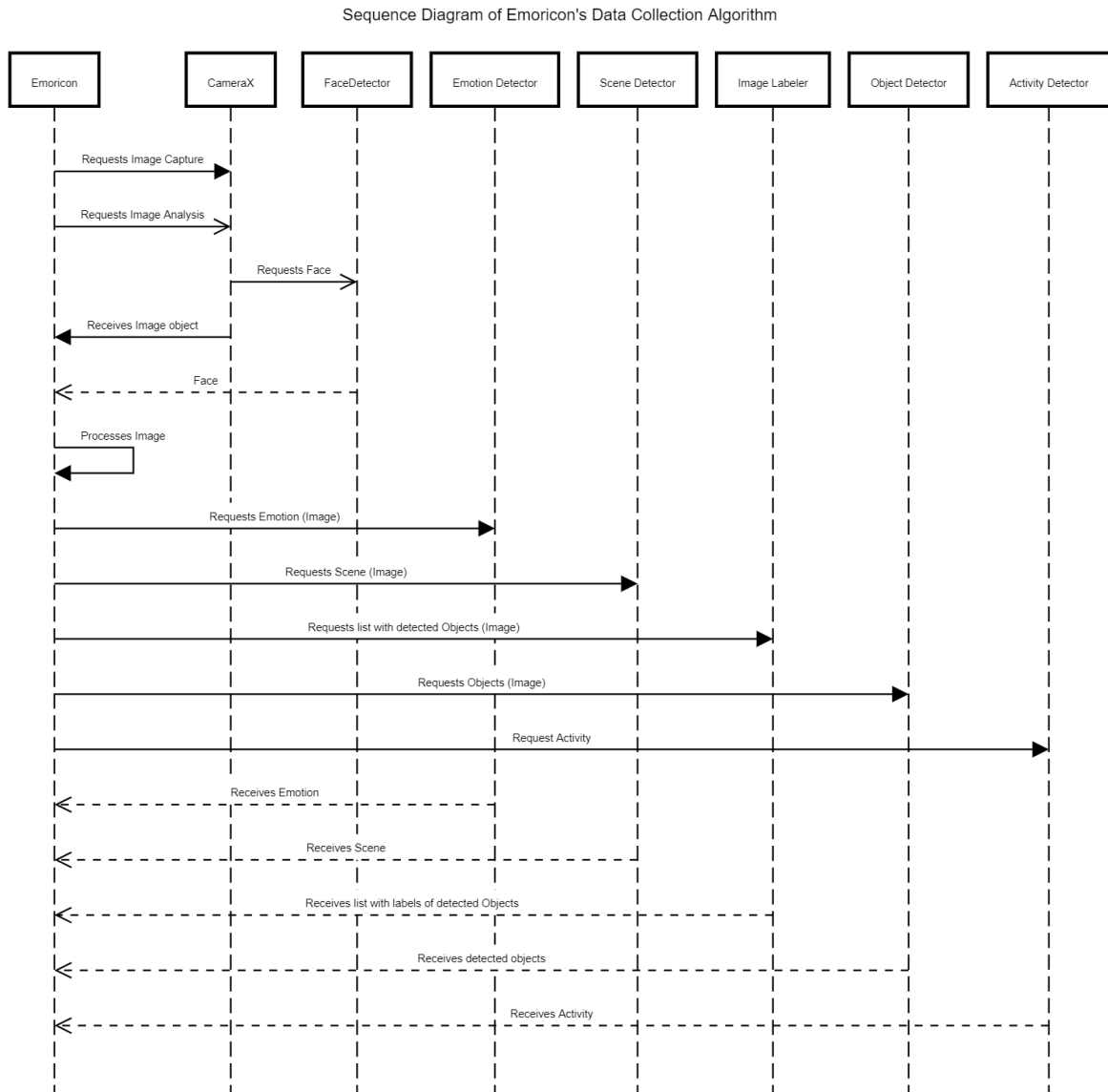
Στην εφαρμογή Emoricon συλλέγουμε raw data από όσους περισσότερους αισθητήρες έχει η συσκευή του κινητού αφού κάνουμε έναν έλεγχο για το ποιοι είναι διαθέσιμοι στην συγκεκριμένη συσκευή. Στην συνέχεια χρησιμοποιούμε sensor events για να καταγράψουμε τα δεδομένα όταν χρειάζεται έτσι ώστε να μπορούν να ληφθούν κατάλληλες αποφάσεις.

3.6.2 Εξαγωγή δεδομένων από Android Camera

Προκειμένου να χρησιμοποιήσουμε την κάμερα της συσκευής του κινητού για τη συλλογή των δεδομένων, θα πρέπει να λειτουργεί καθόσον χρησιμοποιείται η εφαρμογή. Όπως και στην περίπτωση των αισθητήρων, η λειτουργία γίνεται στο background και εφόσον ο χρήστης έχει συναινέσει.

Για να αποφύγουμε προβλήματα συμβατότητας μεταξύ των διαφορετικών συσκευών κινητών επιλέξαμε το API cameraX και συγκεκριμένα το use case Image Analysis δεδομένου ότι θέλουμε να χρησιμοποιήσουμε machine learning, αλλά και το Image Capture για τη διευκόλυνσή μας. Συνεπώς, η εφαρμογή emoricon υλοποιεί μια μέθοδο analyze() που εκτελείται σε κάθε καρέ της κάμερας. Από εκεί εξάγεται ένα αντικείμενο Image, με σκοπό να το εισάγουμε στα μοντέλα μας. Για να ακολουθήσουμε τις απαιτήσεις του Google ML Kit για realtime επεξεργασία μετατρέπουμε το αντικείμενο Image σε διαστάσεις 480x360 pixel. Αυτό εισάγεται στον face detector και μόλις εντοπιστεί ένα πρόσωπο ξεκινούν να εκτελούνται ασύγχρονα όλες οι αλγοριθμικές μέθοδοι συλλογής δεδομένων.

Για να γίνει κατανοητή η λειτουργία του αλγορίθμου χρονικά παρουσιάζουμε το παρακάτω sequence diagram.



Σχήμα 3.16: Διάγραμμα ακολουθίας του αλγορίθμου συλλογής δεδομένων

3.6.3 Φιλτράρισμα και επεξεργασία δεδομένων

Μία από τις προκλήσεις που συναντήσαμε κατά την δημιουργία της εφαρμογής Emoricon ήταν η αποθήκευση δεδομένων στην firebase και πιο συγκεκριμένα το κάθε πότε θα γίνεται. Δεδομένου ότι ο αλγόριθμος συλλογής δεδομένων εκτελείται σε κάθε καρέ της κάμερας παράγονται πάρα πολλά δεδομένα. Πολλά από αυτά επαναλαμβάνονται και όχι μόνο δεν έχει καμία χρησιμότητα η αποθήκευσή τους αλλά υπάρχει και περίπτωση να χρειαστεί να μεταβούμε από το Spark Plan που είναι το free plan της Firebase, στο Blaze plan που είναι το pay as you go, και αυτό λόγω της ανάγκης για περισσότερα read and write στη βάση και για μεγαλύτερη χωρητικότητα.

Επομένως ήταν προφανές ότι πρέπει να φιλτράρουμε τα δεδομένα και να φροντίσουμε να αποθηκεύονται μόνο όταν έχουν σημασία. Για αυτό το λόγο δημιουργήσαμε δύο κλάσεις την DataProcessing και την Analytics.

Σκοπός της DataProcessing είναι να ελέγχει αν τα δεδομένα «αξίζουν» να αποθη-

κευθούν και επίσης εξετάζει αν μπορούμε να τα εμπιστευτούμε. Ο έλεγχος γίνεται χρησιμοποιώντας την τιμή της μεταβλητής *confidence*, η οποία επιστρέφεται από τα μοντέλα μαζί με τα αποτελέσματα της αναγνώρισης. Ανάλογα με την μέτρηση έχουν οριστεί τα αποδεκτά όρια στο *confidence* και επομένως ελέγχεται αν η νέα τιμή είναι μέσα σε αυτά τα όρια, διαφορετικά απορρίπτεται. Στη συνέχεια εξετάζεται αν η νέα τιμή είναι διαφορετική από την προηγούμενη για να κρίνουμε αν έχει νόημα η αποθήκευση της. Από τη διαφορά των *timestamp* αποθήκευσης μεταξύ δύο διαδοχικών μετρήσεων προκύπτει η χρονική διάρκεια της μέτρησης.

Κάποια από τα δεδομένα που έχουν εξαχθεί, αφού φιλτραριστούν αποθηκεύονται ως έχουν στην κλάση *Analytics*, ενώ κάποια άλλα εισάγονται σαν παράμετροι στις μεθόδους της κλάσης και επεξεργάζονται περαιτέρω συνθέτοντας καινούρια δεδομένα. Κάθε φορά που γίνεται αλλαγή στην τάξη *Analytics*, τότε αποθηκεύεται στην *firebase*.

3.7 Ανάπτυξη Εφαρμογών

Έχει γίνει πλέον σαφές ότι όλοι οι προηγούμενοι αλγόριθμοι και οι διαδικασίες που περιγράψαμε, χρησιμοποιούνται μέσω μιας *android* εφαρμογής της *Emoricon*, που αναπτύχθηκε χρησιμοποιώντας το *Android Studio*.

Η *emoricon* είναι μία εφαρμογή *mobile learning* που βασίζεται τόσο στην ασύγχρονη εκπαίδευση όσο και στη σύγχρονη. Χρησιμοποιώντας το *Youtube Data API*, η εφαρμογή έχει πρόσβαση στην μεγαλύτερη *online* πλατφόρμα *video sharing*.

Σκοπός της εφαρμογής είναι να δώσει στο χρήστη τη δυνατότητα να μπορεί να προσθέτει βίντεο ή ολόκληρες λίστες βίντεο με εκπαιδευτικό περιεχόμενο, ώστε να μπορεί να τα παρακολουθήσει κατά απαίτησή του. Κατά τη διάρκεια της παρακολούθησης, δίνεται επίσης η δυνατότητα να κρατηθούν σημειώσεις σε κάθε βίντεο. Παράλληλα με τη θέαση εκτελούνται οι αλγόριθμοι συλλογής δεδομένων που περιγράψαμε στα προηγούμενα κεφάλαια και τα δεδομένα που συλλέγονται αποθηκεύονται στην *firebase*. Η αποθήκευση και ο συγχρονισμός γίνονται πολύ εύκολα, μέσω της πλατφόρμας που παρέχεται από την *Google*.

Ένας χρήστης για να χρησιμοποιήσει την εφαρμογή, θα πρέπει αρχικά να έχει κάνει εγγραφή και να συνδεθεί. Αυτό του δίνει τη δυνατότητα να συνεχίσει τη θέαση από κάποια άλλη κινητή συσκευή, δεδομένου ότι η εφαρμογή συγχρονίζει τα δεδομένα μέσω *cloud*. Η αυθεντικοποίηση είναι απαραίτητη, ώστε να γίνεται αυτός ο συγχρονισμός αλλά και για να να σώζονται τα *analytics-metrics* κάτω από το αναγνωριστικό του κάθε χρήστη (*id*) το οποίο αποτελεί και κλειδί και είναι το βασικό και υψηλότερου επιπέδου *node* (βλ. εικόνα 823). Επιπροσθέτως είναι απαραίτητη, ώστε να μπορεί γίνεται έλεγχος στους χρήστες που χρησιμοποιούν την εφαρμογή, δίνοντας έτσι την δυνατότητα σε οργανισμούς-εταιρείες να μπορούν να επιλέγουν ποιος θα αποκτάει πρόσβαση στο εκπαιδευτικό περιεχόμενο. Αυτός ο περιορισμός πρόσβασης επιτυγχάνεται μέσω της εφαρμογής *youtube*, η οποία χρησιμοποιεί τρεις διαφορετικούς τύπους *visibility* τους *public*, *private* και *unlisted*.

Περιγράφοντας εν συντομία τις διαφορές μεταξύ των τύπων:

- Το *public* (δημόσιο) *status* είναι η προεπιλεγμένη ρύθμιση και αυτό σημαίνει ότι οποιοσδήποτε μπορεί να αναζητήσει και να δει το βίντεο.
- Το *private* (ιδιωτικό) συνεπάγεται ότι για να δει κάποιος το βίντεο πρέπει να του έχει γίνει πρόσκληση και να διαθέτει λογαριασμό *Google*. Ο μέγιστος

αριθμός διαμοιρασμού ενός private video είναι 50 άτομα. Τα private video δεν εμφανίζονται ως αποτέλεσμα αναζήτησης και στη λίστα των καναλιών του δημιουργού.

- Το unlisted (μη καταχωρημένο) status έχει επίσης ως αποτέλεσμα το βίντεο να μην εμφανίζεται ούτε στα αποτελέσματα αναζήτησης ούτε στο κανάλι του δημιουργού. Για να μπορέσει να συνδεθεί κάποιος και να το παρακολουθήσει απαιτείται να γνωρίζει το σύνδεσμο του βίντεο. Ο σύνδεσμος μπορεί να διαμοιραστεί με οποιονδήποτε, ακόμα και με κάποιον που δεν έχει Google account.

Αξιοποιώντας τα παραπάνω visibility status ο εκπαιδευτικός ή ο δημιουργός περιεχομένου αποκτάει έλεγχο στην πληροφορία και μπορεί να προσομοιώσει είτε μία εκπαίδευση σε «μικρή» τάξη των 50 ατόμων με ελεγχόμενη πρόσβαση, είτε μία «αμφιθεατρική» παρουσίαση που η πρόσβαση είναι πιο ελεύθερη αλλά και πάλι υπάρχει κάποιος έλεγχος, είτε μία δημόσια ανοιχτή παρουσίαση με ελεύθερη πρόσβαση. Το πλεονέκτημα της ψηφιακής προσέγγισης σε σχέση με την φυσική παρουσία είναι ότι πέρα από τη σύγχρονη τηλεεκπαίδευση μέσω live video (ζωντανή μετάδοση), υπάρχει σαν επιλογή και η ασύγχρονη, με πολλά και διαφορετικά πλεονεκτήματα αλλά και κάποια μειονεκτήματα που όμως δεν αποτελούν αντικείμενο της συγκεκριμένης συγγραφής.

Παράλληλα με την είσοδό του στην εφαρμογή και αφού έχει δοθεί άδεια για τα κατάλληλα permission ξεκινάει ο αλγόριθμος συλλογής δεδομένων που περιγράφεται στο κεφάλαιο 3.7. Σκοπός μας είναι να αξιοποιηθούν αυτά τα δεδομένα μαζικά από τον εκπαιδευτικό ή το δημιουργό περιεχομένου, ώστε να μπορέσει είτε να θέσει περιορισμούς στη διαδικασία της εκπαίδευσης είτε να κριθεί η απήχησή της ή ακόμα και η αποδοτικότητα του συγκεκριμένου περιεχομένου.

Περισσότερες πληροφορίες για την λειτουργικότητα θα αναφερθούν στο κεφάλαιο 4.1 Εφαρμογή Android.

3.8 Analytics και Metrics

Παράλληλα με την emoricon που αποτελεί πρακτικά client για τους χρήστες της εφαρμογής, έχουν δημιουργηθεί και δύο python notebooks, ώστε να μπορούν οι διαχειριστές να αποκτήσουν πρόσβαση στα analytics και στα metrics τα οποία έχουν καταγραφεί για κάθε χρήστη αλλά και συνολικά.

Τα Python notebooks αναπτύχθηκαν μέσω της πλατφόρμας Google Colab, η οποία είναι ένα δωρεάν περιβάλλον για Jupyter notebook που λειτουργεί εξ ολοκλήρου στο cloud. Το πιο σημαντικό είναι ότι δεν απαιτεί την εγκατάσταση και τα notebooks που δημιουργούμε μπορούν να επεξεργαστούν ταυτόχρονα από τα μέλη μίας ομάδας, όπως ακριβώς γίνεται και με τα έγγραφα στο Google Docs. Το Colab υποστηρίζει πολλές δημοφιλείς βιβλιοθήκες μηχανικής μάθησης, οι οποίες μπορούν εύκολα να φορτωθούν στο σημειωματάριό μας.

Και στα δύο σημειωματάρια πραγματοποιούμε μία σύνδεση με την firebase και φορτώνουμε τη nosql database που βρίσκεται υπό τη μορφή αντικειμένου JSON, σε python dictionary. Από εκεί και πέρα χρησιμοποιούμε τη βιβλιοθήκη της python, pandas για να μετατρέψουμε το dictionary σε table και να τρέξουμε query σαν να ήταν κάποια σχεσιακή βάση δεδομένων. Στη συνέχεια, με την βιβλιοθήκη matplotlib βγάζουμε διαγράμματα για να ομαδοποιήσουμε τα δεδομένα και να παρατηρήσουμε τη συμπεριφορά τους.

Ο λόγος που χρησιμοποιούμε *python notebooks* στο Google Colab είναι για να παρέχουμε τη δυνατότητα στο διαχειριστή να τροποποιήσει τα δεδομένα με ευκολία *out of the box*, χωρίς να χρειαστεί να εγκαταστήσει κανένα *framework* καθώς και την *python* και τις βιβλιοθήκες της. Έχοντας έτοιμα τα *notebooks* και με ελάχιστες γνώσεις του *panda framework*, μπορούν όλα τα *analytics* να τροποποιηθούν κατά απαίτηση.

3.8.1 User Analytics

Στο σημειωματάριο με τα *User Analytics* ο διαχειριστής της εφαρμογής (εκπαιδευτικός-δημιουργός περιεχομένου) μπορεί να βάλει το email ενός συγκεκριμένου χρήστη και να δει τα βίντεο που έχει προσθέσει ο χρήστης καθώς και στατιστικά σχετικά με το δεδομένα που συλλέγουμε.

Για παράδειγμα μπορούμε να δούμε :

- το επικρατές συναίσθημα του χρήστη συναρτήσει του χρόνου, καθώς και το τελευταίο συναίσθημα που αναγνωρίστηκε
- το ποσοστό των μαθημάτων που παρακολούθησε ο χρήστης εν κινήσει
- το ποσοστό των μαθημάτων που παρακολούθησε μόνος του σε σύγκριση με αυτά που υπήρχαν και άλλοι άνθρωποι στο χώρο
- μαθήματα σε σχέση με το *context*
- την τελευταία τοποθεσία του χρήστη
- τα υπόλοιπα *metrics* που διαβάστηκαν από τον αισθητήρα του κινητού
- λίστα με αντικείμενα που αναγνωρίστηκαν κατά την διάρκεια συλλογής δεδομένων

3.8.2 Mass Analytics

Στο σημειωματάριο με τα μαζικά *analytics* ο διαχειριστής της εφαρμογής βλέπει συγκεντρωτικά δεδομένα από όλους τους χρήστες και όλα τα βίντεο.

Μερικά από αυτά αναφέρονται παρακάτω :

- Το top 10 των μαθημάτων
- Τα τελευταία *trends*
- Τι ποσοστό των χρηστών διαβάζει στο γραφείο του, στον καναπέ και στο κρεβάτι
- Τι ποσοστό των χρηστών της εφαρμογής έχει κατοικίδιο
- Τι ποσοστό διαβάζει μόνο του και τι με παρέα
- Τα top 10 αντικείμενα που αναγνωρίστηκαν στο χώρο
- Τα top 10 από τα *context* που αναγνωρίστηκαν

3.9 Διαδικασία Ανάπτυξης Λογισμικού

Για την ανάπτυξη του λογισμικού πραγματοποιήθηκε ένας πλήρης κύκλος ανάπτυξης λογισμικού χρησιμοποιώντας το μοντέλο του καταρράκτη.

Οι φάσεις στο μοντέλο του καταρράκτη που ακολουθήσαμε είναι οι παρακάτω :

1. Συγκέντρωση των απαιτήσεων (*Gather Information and Planning*)

2. Ανάλυση και καθορισμός των δεδομένων (Analysis)
3. Σχεδιασμός συστήματος και Αρχιτεκτονική λογισμικού (Design)
4. Ανάπτυξη Λογισμικού (Implementation)
5. Δοκιμές (Testing)
6. Λειτουργία και Εγκατάσταση (Deployment)

4 Υλοποίηση

4.1 Εφαρμογή Android

Για την υλοποίηση του Emoricon δημιουργήσαμε μία εφαρμογή Android χρησιμοποιώντας το Android Studio. Όλα τα δεδομένα που αποθηκεύει ο χρήστης καθώς και αυτά που συλλέγονται ή παράγονται από την εφαρμογή, σώζονται σε Realtime Database στο Firebase API και έχει ενεργοποιηθεί η offline λειτουργία της Realtime Database έτσι ώστε να μπορεί η εφαρμογή να δουλέψει αν χαθεί προσωρινά η σύνδεση με το internet, ωστόσο για τη σωστή χρήση της εφαρμογής απαιτείται σύνδεση με το internet.

4.1.1 Οθόνη Splash

Η οθόνη Splash είναι η πρώτη επαφή του χρήστη της εφαρμογής με την Emoricon. Εμφανίζεται κατά το άνοιγμα της εφαρμογής και παραμένει για μερικά δευτερόλεπτα. Η εμφάνιση μίας splash οθόνης είναι πλέον εκτός από κλασσική και απαραίτητη για το user experience.

Για τη δημιουργία μίας οθόνης τύπου Splash υπάρχουν δύο τρόποι:

1. Χρησιμοποιώντας Timers
2. Χρησιμοποιώντας το ένα Launcher Theme

Ο πρώτος τρόπος ήταν ο παραδοσιακός τρόπος που χρησιμοποιούνταν στις παλαιότερες εκδόσεις του Android αλλά δεν είναι καλύτερος, γιατί θα οδηγήσει σε ψυχρή εκκίνηση (cold start) [63]. Η ψυχρή εκκίνηση εμφανίζεται επειδή η εφαρμογή χρειάζεται χρόνο για να φορτώσει το αρχείο διάταξης της δραστηριότητας Splash. Έτσι, αντί να δημιουργήσουμε το layout, θα χρησιμοποιήσουμε τη δύναμη που μας προσφέρει το Application Theme του Android για να επιτύχουμε την ίδια εμφάνιση και λειτουργικότητα. Αυτός ο δεύτερος τρόπος, είναι και ο προτεινόμενος από την Google.

Το θέμα της εφαρμογής γίνεται instantiate πριν από τη δημιουργία του layout. Ορίσαμε ένα drawable μέσα στο θέμα (theme.xml) που περιλαμβάνει το φόντο της δραστηριότητας και το εικονίδιο της εφαρμογής μας χρησιμοποιώντας layer-list. Στην συνέχεια ορίσαμε αυτό το προσαρμοσμένο θέμα στο Manifest για το δραστηριότητα SplashScreen. Εφόσον φορτώνουμε την οθόνη Splash από το Manifest δεν χρειάζεται να χρησιμοποιήσουμε setContentView() με κάποιο xml για layout.

Κατά τη φόρτωση της εφαρμογής μας στην δραστηριότητα SplashScreen, πραγματοποιείται η σύνδεση με το API της Firebase, γίνεται instantiate η Realtime Database και επιχειρείται authentication. Για τη συγκεκριμένη λειτουργία έχουμε δημιουργήσει έναν Session Manager ο οποίος μας βοηθάει να αποθηκεύουμε τα δεδομένα του χρήστη εκτός της εφαρμογής, έτσι ώστε την επόμενη φορά που ο χρήστης θα χρησιμοποιήσει την εφαρμογή μας, να μπορούμε εύκολα να ανακαλέσουμε της πληροφορίες σύνδεσης του και να κάνουμε το authentication αυτόματα.

Με αυτόν τον τρόπο, αν η emoricon καταφέρει να κάνει authentication με τα στοιχεία που θα συλλέξει από τον session manager προχωράει στην επόμενη δραστηριότητα (MainApp), η οποία είναι και η κεντρική δραστηριότητα της εφαρμογής μας, διαφορετικά οδηγεί το χρήστη στην οθόνη σύνδεσης (LogIn).

Η splash screen εμφανίζεται στην οθόνη για τέσσερα δευτερόλεπτα και πέρα από την πρακτική χρησιμότητα που έχει, να παίρνει τη θέση κάποιας κενής άσπρης

οθόνης όσο γίνεται η φόρτωση, δεν παύει να αποτελεί και την πρώτη εμπειρία του χρήστη με την εφαρμογή μας. Οπότε επιδιώξαμε ο σχεδιασμός να τραβάει την προσοχή του χρήστη και ελπίζουμε να του προσελκύσει το ενδιαφέρον.

Για να επιτύχουμε την επιθυμητή καθυστέρηση ώστε ο χρήστης να προλάβει να παρατηρήσει την οθόνη Splash, δημιουργήσαμε έναν Handler με τη μέθοδο `postDelayed()`. Ο χρόνος `SPLASH_DELAY` καθώς και όλοι οι άλλοι παράμετροι της εφαρμογής μας καθορίζονται στο αρχείο `Constants.java`.

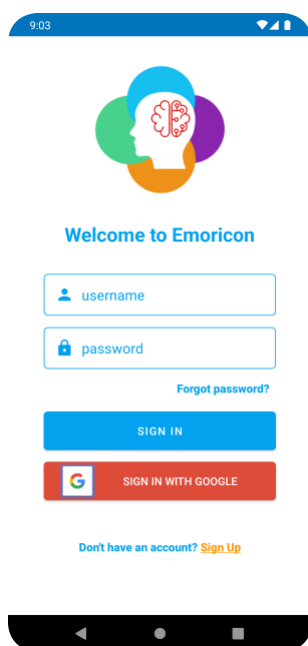


Σχήμα 4.1: Οθόνη Splash

4.1.2 Οθόνη LogIn

Η οθόνη LogIn είναι η πρώτη οθόνη που πρακτικά απαιτεί αλληλεπίδραση με το χρήστη και του δίνει τη δυνατότητα να συνδεθεί στην εφαρμογή, χρησιμοποιώντας το email και τον κωδικό του ή να κάνει authentication χρησιμοποιώντας το google account του. Ακόμα, όπως σε όλες σχεδόν τις οθόνες σύνδεσης, ο χρήστης μπορεί να αιτηθεί να κάνει reset το password του.

Αν ο χρήστης δεν έχει λογαριασμό μπορεί να επιλέξει SignUp και να μεταβεί στη δραστηριότητα Register.

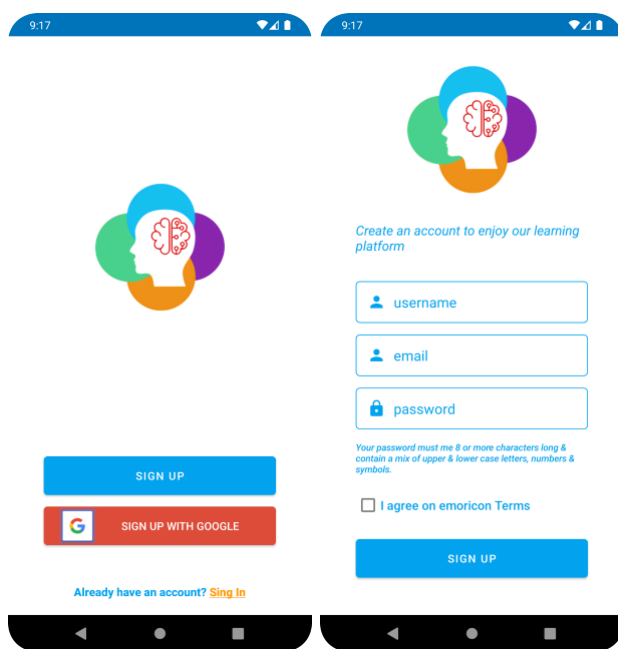


Σχήμα 4.2: Οθόνη Splash

Όταν πατηθεί κάποιο από τα κουμπιά SIGN IN, τότε χρησιμοποιείται το Firebase Authentication και αποστέλλονται τα στοιχεία του χρήστη για να γίνει η ταυτοποίησή τους. Αν τα στοιχεία είναι έγκυρα, γίνεται authentication και τότε ο χρήστης μεταφέρεται στην οθόνη «MainApp», ενώ στην περίπτωση που δεν είναι έγκυρα, τότε ενημερώνεται ο χρήστης για το σφάλμα που παρουσιάστηκε .

4.1.3 Οθόνη Register

Στην οθόνη αυτή παρέχεται στο χρήστη η δυνατότητα να δημιουργήσει ένα νέο λογαριασμό για να μπορέσει να συνδεθεί στην εφαρμογή. Ο λογαριασμός δημιουργείται με την χρήση της βιβλιοθήκης Firebase Authentication, χρησιμοποιώντας για στοιχεία σύνδεσης ένα Email και έναν κωδικό πρόσβασης. Παράλληλα αποθηκεύεται στην Realtime Database, βάση του αναγνωριστικού (id), το προφίλ του χρήστη που περιέχει το όνομά του και τα στατιστικά του με τις default τιμές. Αφού ολοκληρωθεί επιτυχώς η δημιουργία του λογαριασμού, τότε ο χρήστης μεταφέρεται στην οθόνη «MainApp». Σε περίπτωση που παρουσιαστεί οποιαδήποτε δυσλειτουργία, ο χρήστης ενημερώνεται για το σφάλμα.



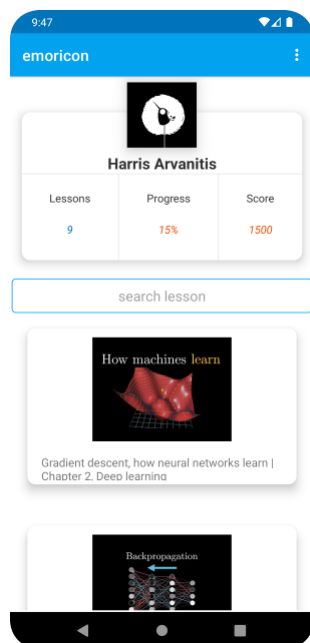
Σχήμα 4.3: Οθόνες δημιουργίας λογαριασμού

4.1.4 Κεντρική οθόνη Emoricon

Η κεντρική οθόνη της εφαρμογής εμφανίζεται μέσω της δραστηριότητας MainApp. Σε αυτήν ο χρήστης μπορεί να δει τη λίστα με τα μαθήματα που έχει προσθέσει, να αναζητήσει κάποιο από αυτά και τελικά να επιλέξει την θέασή του. Στο πάνω μέρος της οθόνης παρουσιάζονται κάποια στοιχεία από το προφίλ του χρήστη και κάποια στατιστικά και ακολουθεί μία λίστα με τα μαθήματα που έχει προσθέσει ο χρήστης. Το προφίλ και το κάθε μάθημα εμφανίζονται χρησιμοποιώντας το UI component cardview [64] ενώ η λίστα με τα μαθήματα προσαρτάται σε ένα RecyclerView [65].

Το Android RecyclerView είναι μια πιο προηγμένη, ισχυρή και ευέλικτη έκδοση του ListView. Στο RecyclerView μπορούμε να φορτώσουμε μεγάλο όγκο δεδομένων και τα στοιχεία του RecyclerView μπορούν να έχουν custom design. Λειτουργεί με βάση το πρότυπο σχεδίασης ViewHolder, οπότε πρέπει να δημιουργήσουμε μια κλάση Data που κρατάει τα δεδομένα για το RecyclerView και μια κλάση ViewHolder που θα θέτει δεδομένα σε κάθε στοιχείο του RecyclerView. Όπως υποδηλώνει και το όνομα, το Android RecyclerView χρησιμοποιείται για την επαναχρησιμοποίηση κελιών κατά την κύλιση προς τα πάνω και κάτω με την ανακύκλωση των στοιχείων της λίστας. Τέλος με το RecyclerView ορίζουμε τους LayoutManagers δυναμικά κατά την εκτέλεση, σε αντίθεση με το ListView που ήταν διαθέσιμο μόνο σε μια λίστα κάθετης κύλισης. [65]

Μόλις ο χρήστης επιλέξει κάποιο στοιχείο της λίστας, ανοίγει η οθόνη Youtube-Player για να ξεκινήσει η παρακολούθηση του συγκεκριμένου μαθήματος.



Σχήμα 4.4: Κεντρική οθόνη εφαρμογής

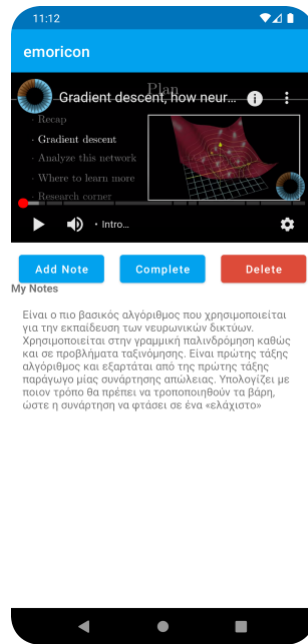
4.1.5 Οθόνη Προβολής Μαθήματος

Σε αυτή την οθόνη, ο χρήστης πραγματοποιεί τη θέαση των μαθημάτων του και μπορεί να κρατήσει σημειώσεις για αυτά. Η προβολή του βίντεο γίνεται χρησιμοποιώντας το `Android-youtube-player` [66] το οποίο είναι ένα customizable open source youtube player για το Android. Παρέχει μια απλή προβολή τη `YouTubePlayerView`, που μπορεί εύκολα να ενσωματωθεί σε οποιαδήποτε δραστηριότητα.

Η βιβλιοθήκη είναι ένα wrapper πάνω από το `IFrame Player API` [67], το οποίο τρέχει σε ένα `WebView`. Ως εκ τούτου, δεν απαιτείται η εφαρμογή `YouTube` στη συσκευή του χρήστη και επίσης δεν υπάρχουν ζητήματα με τους όρους χρήσης του `YouTube`. Χρησιμοποιήθηκε η συγκεκριμένη βιβλιοθήκη αντί της official, διότι παρέχει περισσότερη λειτουργικότητα και έχει λυμένα κάποια προβλήματα που αναφέρθηκαν στην official εφαρμογή.

Σε αυτή την οθόνη ο χρήστης μπορεί να προσθέσει τις σημειώσεις του για κάθε ένα από τα μαθήματα που παρακολουθεί. Μπορεί να χαρακτηρίσει ένα βίντεο ως "completed" και να ενημερώσει το progress στο προφίλ του ή να διαγράψει ένα βίντεο από τη λίστα του.

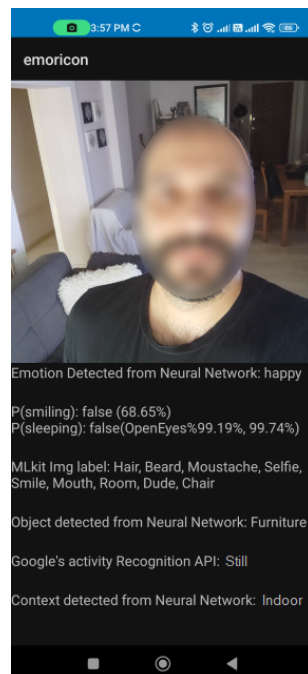
Όσο ο χρήστης παρακολουθεί τα μαθήματα του, ο αλγόριθμος συλλογής δεδομένων εκτελείται στο παρασκήνιο. Για να μπορέσουμε να δοκιμάσουμε τη λειτουργία της εφαρμογής, δημιουργήσαμε μία δραστηριότητα την `"GetEmotionActivity"`, στην οποία μπορούμε να δούμε live την εκτέλεση των αλγορίθμων μηχανικής μάθησης να μας δείχνουν κάποια από τα δεδομένα που συλλέγονται.



Σχήμα 4.5: Οθόνη Προβολής Μαθήματος

4.1.6 Οθόνη Live Capture

Η Οθόνη Live Capture (δραστηριότητα GetEmotionActivity) δημιουργήθηκε για να διευκολυνθεί η ανάπτυξη λογισμικού. Υπάρχει στο Administrator Menu και μέσω αυτής μπορούν να ελεγχθούν οι αλγόριθμοι συλλογής δεδομένων. Ένα παράδειγμα εκτέλεσης των αλγορίθμων εμφανίζεται στην παρακάτω εικόνα.



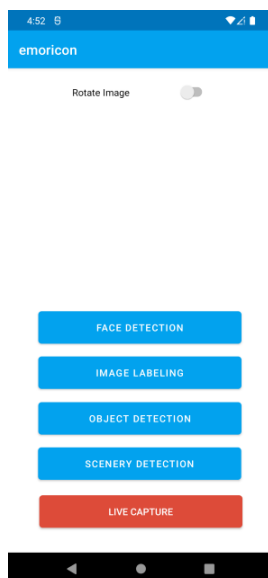
Σχήμα 4.6: Οθόνη Live Capture

Τα δεδομένα που φαίνονται στην εικόνα προκύπτουν από τον αλγόριθμο συλλογής δεδομένων και στη συνέχεια φιλτράρονται μέσω της κλάσης DataProcessing. Για παράδειγμα μπορούμε να αναφέρουμε ότι το χαμόγελο που εντοπίζεται στην εικόνα

4.6 είναι 68,65% και επειδή δεν ξεπερνάει το 70% που ορίσαμε σαν όριο, στο Smiling flag επιστρέφεται η τιμή false.

4.1.7 Οθόνη Test Models

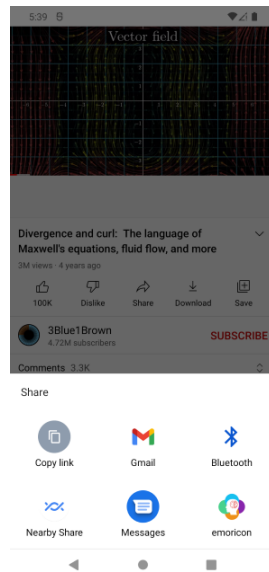
Όπως και στην περίπτωση της προηγούμενης δραστηριότητας, η συγκεκριμένη οθόνη δημιουργήθηκε για να μπορεί να ελεγχθεί η λειτουργία των μοντέλων που χρησιμοποιεί η emoricon, αλλά αυτή τη φορά αφορά μέσω εικόνας που συλλαμβάνουμε με τη χρήση της κάμερας. Η πρόσβαση στη συγκεκριμένη οθόνη γίνεται μέσω του Administrator menu της εφαρμογής μας. Κάθε μπλε κουμπί ελέγχει ένα διαφορετικό μοντέλο που χρησιμοποιεί η emoricon εκτός από το τελευταίο που μας οδηγεί στην οθόνη live capture. Αυτή η οθόνη καταγράφει στο log file την έξοδο ή/και τα αποτελέσματα αναγνώρισης των μοντέλων. Η εμφάνιση του log στο Android studio γίνεται μέσω του logcat, ενώ μέσω της εφαρμογής από το menu option *Event Log*.



Σχήμα 4.7: Οθόνη δοκιμαστικής λειτουργίας μοντέλων

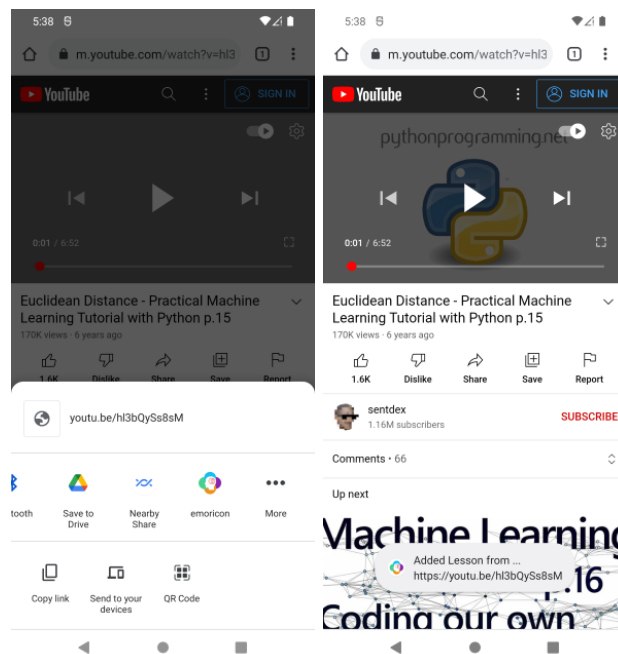
4.1.8 Εισαγωγή Μαθημάτων

Η εισαγωγή μαθημάτων στην emoricon γίνεται μέσω της λειτουργίας που υποστηρίζει το android για τη λήψη δεδομένων από άλλες εφαρμογές. Ο χρήστης μπορεί να κάνει share οποιοδήποτε youtube video και να επιλέξει την emoricon και αυτό θα προστεθεί αμέσως στην λίστα του.



Σχήμα 4.8: Εισαγωγή μαθημάτων από Youtube

Το import δεν απαιτείται να γίνει μέσω της εφαρμογής youtube mobile αλλά υποστηρίζεται μέσω οποιασδήποτε εφαρμογής ή σελίδας στο web δείχνει embedded video. Για παράδειγμα παρουσιάζεται η εισαγωγή ενός video μέσω του Chrome web browser. Ο χρήστης ενημερώνεται με μήνυμα μόλις ολοκληρωθεί η εισαγωγή του video και πλέον είναι άμεσα διαθέσιμο σε όλες του τις συσκευές μέσω της emoricon.



Σχήμα 4.9: Εισαγωγή Youtube ιδεο από Chrome Web Browser

Η συγκεκριμένη λειτουργικότητα επιτυγχάνεται μέσω της δραστηριότητας 'AddVideo' και του android framework. Αρχικά φροντίσαμε στο Android Manifest να κάνουμε export τη δραστηριότητα και να δηλώσουμε action και data στο intent-filter όπως φαίνεται παρακάτω.

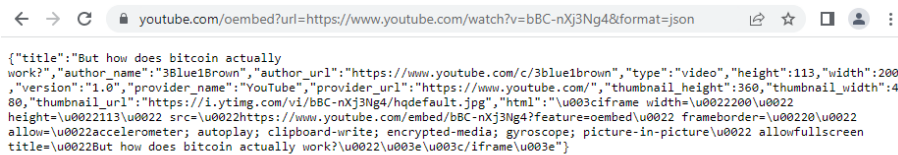
```

<activity
    android:name=".AddVideo"
    android:exported="true">
    <intent-filter>
        <action android:name="android.intent.action.SEND" />
        <category android:name="android.intent.category.DEFAULT" />
        <data
            android:host="www.youtube.com"
            android:mimeType="text/*" />
    </intent-filter>
</activity>

```

Σχήμα 4.10: Manifest file για την δραστηριότητα προσθήκης βίντεο

Στη συνέχεια παραλαμβάνουμε το αποτέλεσμα στην δραστηριότητα 'AddVideo' αποθηκεύοντας το 'videoURL', από το οποίο δημιουργούμε το 'embedURL' ώστε να μπορέσουμε να πάρουμε κάποιες βασικές πληροφορίες για το συγκεκριμένο video χωρίς να κάνουμε κλήση στο Youtube Video API. Στο embedURL προσθέτουμε ως παράμετρο την &format=json ώστε να μας επιστρέψει ένα json object όπως φαίνεται παρακάτω.



```

{"title":"But how does bitcoin actually work?","author_name":"3Blue1Brown","author_url":"https://www.youtube.com/c/3blue1brown","type":"video","height":113,"width":200,"version":"1.0","provider_name":"YouTube","provider_url":"https://www.youtube.com/","thumbnail_height":360,"thumbnail_width":480,"thumbnail_url":"https://i.ytimg.com/vi/bBC-nXj3Ng4/hqdefault.jpg","html":"\u003ciframe width=\u0022200\u0022 height=\u0022113\u0022 src=\u0022https://www.youtube.com/embed/bBC-nXj3Ng4?feature=oembed\u0022 frameborder=\u00220\u0022 allow=\u0022accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture\u0022 allowfullscreen title=\u0022But how does bitcoin actually work?\u0022\u003e\u003c\/iframe\u003e"}

```

Σχήμα 4.11: Youtube Embedded URL

Στη συνέχεια, χρησιμοποιώντας ένα JsonTask το οποίο εκτελείται ασύγχρονα, κάνουμε κλήση στο 'embedURL', ώστε να κατεβάσουμε το json και να δημιουργήσουμε ένα αντικείμενο GSON. Το τελευταίο το κάνουμε deserialize σε ένα αντικείμενο YoutubeVideo και έτσι προσθέτουμε κάποιες βασικές πληροφορίες. Τελικά, το αποθηκεύουμε στην firebase μέσω της μεθόδου 'addVideoToDatabase'

```

Bundle extras = getIntent().getExtras();
if (extras!=null)
{
    videoURL = extras.getString(Intent.EXTRA_TEXT);
    embedURL = "https://www.youtube.com/oembed?url="+ videoURL + "&format=json";
    new JsonTask().execute(embedURL);
}

private void addVideoToDatabase(String videoLink) {
    videoID = getYoutubeVideoId(videoLink);
    newLesson = new YoutubeVideo(videoID,videoName,videoLink);
    if(STORE_TO_CLOUD)
    {
        myRef.child(userID).child(newLesson.getVideoId()).setValue(newLesson);
        Toast.makeText(getApplicationContext(), text "Added Lesson from ... \n"+
            videoLink, Toast.LENGTH_LONG).show();
    }
    finish();
}
}

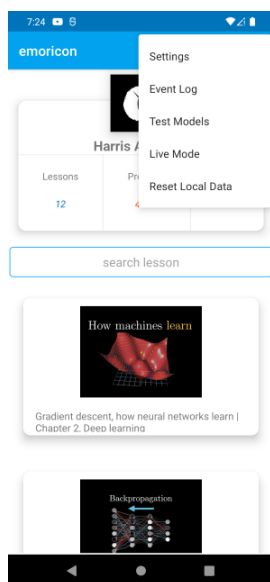
```

Σχήμα 4.12: addVideoToDatabase method

4.1.9 Menu Διαχείρισης

Το Μενυ διαχείρισης μας δίνει τις παρακάτω επιλογές.

- Ρυθμίσεις
- Event Log
- Test Models
- Live Mode
- Reset Local Data



Σχήμα 4.13: addVideoToDatabase method

Οι πρώτες τέσσερις επιλογές ανοίγουν διαφορετικές δραστηριότητες εκ των οποίων έχουμε ήδη αναφερθεί σε κάποιες από αυτές, ενώ η τελευταία επιλογή 'Reset Local Data' μας δίνει τη δυνατότητα να καθαρίσουμε την τοπική βάση δεδομένων που σώζουμε μέσω shared preferences, ώστε να επαναρχικοποιήσουμε την εφαρμογή.

4.1.10 Οθόνη Ρυθμίσεων

Η οθόνη ρυθμίσεων δημιουργήθηκε για να μας δώσει τη δυνατότητα να αλλάξουμε μέσω της εφαρμογής Android κάποιες βασικές τιμές ρυθμίσεων που ελέγχουν την λειτουργία του προγράμματος και που έχουν οριστεί στην κλάση 'Constants.java'.

Παρακάτω εξηγούνται εν συντομία οι επιλογές που δίνονται μέσω της οθόνης των ρυθμίσεων.

- Enable Logger: ενεργοποιεί/απενεργοποιεί την καταγραφή στο 'Event Log' της εφαρμογής
- Preserve Log: καταχωρεί τα log σε Shared Preferences ώστε να διατηρηθούν μετά την επανεκκίνησή της εφαρμογής.
- Store To Cloud: ελέγχει το αν η εφαρμογή θα αποθηκεύει τα δεδομένα της στο cloud ή όχι.
- Clear my Online Data: το συγκεκριμένο switch καθαρίζει τα analytics του συγκεκριμένου χρήστη από την Online Firebase.

- **Detection Interval:** είναι ο χρόνος που θα περιμένει εφαρμογή πριν εκτελεστεί ξανά ο αλγόριθμος συλλογής δεδομένων
- **Update Interval:** είναι ο ελάχιστος χρόνος που θα περιμένει η εφαρμογή πριν γράψει τα Analytics στην Online Firebase.

```
Constants.java
1 package com.unipi.xarvanitis.emoricon;
2
3 import java.text.DecimalFormat;
4
5 public class Constants {
6     public static Boolean DEBUG=false;
7     public static Boolean ENABLE_LOGGER=true;
8     public static Boolean PRESERVE_LOG=false;
9     public static Boolean STORE_TO_CLOUD=true;
10    public static final int MY_PERMISSIONS_REQUEST_ACTIVITY_RECOGNITION = 10;
11    public static long DETECTION_INTERVAL_IN_MILLISECONDS=5000;
12    public static int CONTEXT_LIST_SIZE=8;
13    public static final int SPLASH_DELAY=4000;
14    public static int UPDATE_PROFILE_IN_DATABASE_IN_SECONDS=15;
15    public static DecimalFormat df = new DecimalFormat( pattern: "0.00");
```

Σχήμα 4.14: Constants class

4.1.11 Οθόνη Logger

Στην οθόνη logger μπορεί ο χρήστης της εφαρμογής να δει την καταγραφή πολλών βασικών δεδομένων που αφορούν την λειτουργία της Emoricon.

Πιο συγκεκριμένα στο event log της εφαρμογής καταγράφονται τα παρακάτω :

- σύνδεση με την Firebase και μήνυμα από user authentication
- Start/Stop από τα διάφορα Services της εφαρμογής
- την έξοδο των μοντέλων με αφιльтράριστα όμως δεδομένα (raw data)
- διάφορα άλλα δεδομένα απαραίτητα για έλεγχο της λειτουργίας της εφαρμογής

Παράδειγμα εκτέλεσης μπορεί να φανεί στην παρακάτω εικόνα.



Σχήμα 4.15: Οθόνη Logger

4.2 Εφαρμογή διαχείρισης Analytics

Η υλοποίηση των analytics έχει γίνει μέσω της Online πλατφόρμας Google Colab σε python jupyter notebook. Έχουμε δημιουργήσει δύο διαφορετικά notebook, ένα για τα analytics του χρήστη 'UserAnalytics.ipynb' και ένα για τα μαζικά analytics 'MassAnalytics.ipynb'.

Για να κάνουμε τη σύνδεση με τη firebase κάναμε generate το API key από τη firebase console και κατεβάσαμε το credential key το οποίο το αποθηκεύσαμε στο Google Drive. Στην συνέχεια κάνουμε mount το Google Drive ώστε να μπορούμε να διαβάζουμε και να αποθηκεύουμε αρχεία και ξεκινάμε τη σύνδεση με τη firebase. Αφού ολοκληρωθεί το authentication, τραβάμε το json από την firebase και το σώζουμε σαν python dictionary. Στη συνέχεια το μετατρέπουμε ως pandas table ώστε να εκμεταλλευτούμε τις μεθόδους που παρέχονται για data analysis και data manipulation.

Τέλος, χρησιμοποιούμε την βιβλιοθήκη matplotlib για να δημιουργήσουμε διάφορα διαγράμματα και να βγάλουμε analytics, ώστε να εκτιμήσουμε την μαθησιακή διαδικασία.

4.2.1 UserAnalytics.ipynb

Στις παρακάτω εικόνες παρουσιάζονται παραδείγματα από διαφορετικά user metrics που θα μπορούσαν να εξαχθούν από τα δεδομένα του χρήστη που αποθηκεύονται online.

Analytics for user [xarci@tuta.io](#)(f282b0wVfG6nfp00f45FPhsYbY1)

	detectedLabelsList	detectedObjectsList	detectedSceneryList	emotion	leftEyeOpenProbability	rightEyeOpenProbability	scenery	sleeping	smiling	smilingProbability	timestamp
-	[Beard, Hair, Moustache, Selfie, Dude, Mouth, Eyelash]	[Person, Clothing]	[Landscape, Macro, Indoor, Blue Sky, Greenery, Kids, Food, Sunset, Sunrise]	{'anotation': 'neutral', 'id': 4}	0.996141	0.996212	{'anotation': 'Waterfall', 'id': 10}	False	False	0.014574	1665170307351
-	[Beard, Hair, Moustache, Selfie, Dude, Mouth, Eyelash]	[Person, Clothing]	[Landscape, Macro, Indoor, Blue Sky, Greenery, Kids, Sunset, Sunrise, Food]	{'anotation': 'neutral', 'id': 4}	0.997919	0.996212	{'anotation': 'Waterfall', 'id': 10}	False	False	0.011907	1665170307353
-	[Beard, Hair, Moustache, Selfie, Dude, Mouth, Eyelash]	[Person, Clothing]	[Landscape, Macro, Indoor, Blue Sky, Greenery, Kids, Sunset, Sunrise, Food]	{'anotation': 'neutral', 'id': 4}	0.995686	0.996313	{'anotation': 'Waterfall', 'id': 10}	False	False	0.011408	1665170307355
-	[Beard, Hair, Moustache, Selfie, Dude, Mouth, Eyelash]	[Person, Clothing]	[Landscape, Macro, Indoor, Blue Sky, Greenery, Kids, Sunset, Sunrise]	{'anotation': 'neutral', 'id': 4}	0.998789	0.997491	{'anotation': 'Waterfall', 'id': 10}	False	False	0.016266	1665170307356

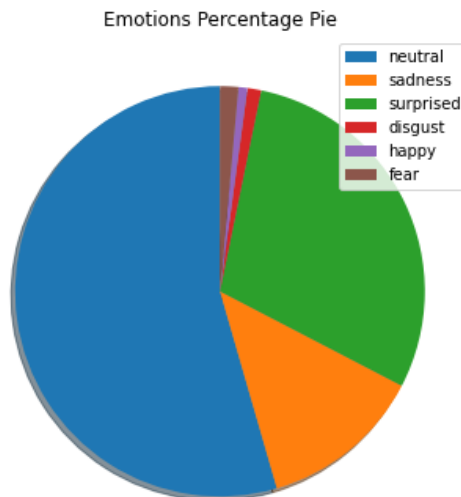
Σχήμα 4.16: Παράδειγμα από pandas dataframe αφού έχει εισαχθεί το json από την firebase

Ένα βασικό στατιστικό που αποτελεί και βάση της διπλωματικής είναι τα διαφορετικά συναισθήματα που αναγνωρίστηκαν κατά τη διαδικασία της μάθησης. Για να μπορέσουμε να παρουσιάσουμε το ποσοστό αναγνώρισης ξεκινάμε μετασχηματίζοντας τα δεδομένα και αυτό επειδή μέσα στην firebase σώζουμε ολόκληρο το αντικείμενο Emotion και οπότε πρέπει να κάνουμε deserialize.

Για παράδειγμα σε κάθε γραμμή του dataframe έχουμε ένα αντικείμενο Series σαν το παρακάτω:

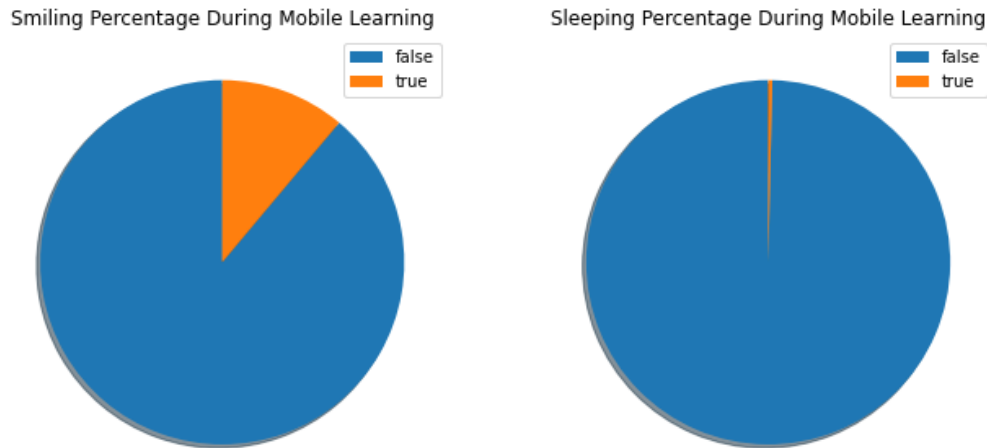
```
{'anotation': 'neutral', 'id': 4}
```

Οπότε μετατρέπουμε το αντικείμενο σε λίστα, μετά σε dictionary και τελικά μπορούμε πλέον να προχωρήσουμε στη δημιουργία των γραφημάτων.



Σχήμα 4.17: Πίτα διαφορετικών συναισθημάτων που ανιχνεύθηκαν κατά τη διαδικασία μάθησης ενός συγκεκριμένου χρήστη

Ένα ακόμα ενδιαφέρον στατιστικό που προκύπτει είναι το κατά πόσο ήταν για το χρήστη ευχάριστο το μάθημα ή όχι και σε τι ποσοστό. Αυτό μπορεί να φανεί από το παρακάτω γράφημα πίτας. Δίπλα του και προσπαθώντας να εκτιμήσουμε το κατά πόσον ο χρήστης είχε την προσοχή του κατά τη διαδικασία της μάθησης εμφανίζουμε ένα ακόμα γράφημα με το ποσοστό που ο χρήστης κοιμόταν ή για να είμαστε πιο ακριβείς με το ποσοστό που ο χρήστης είχε και τα δύο του μάτια κλειστά.

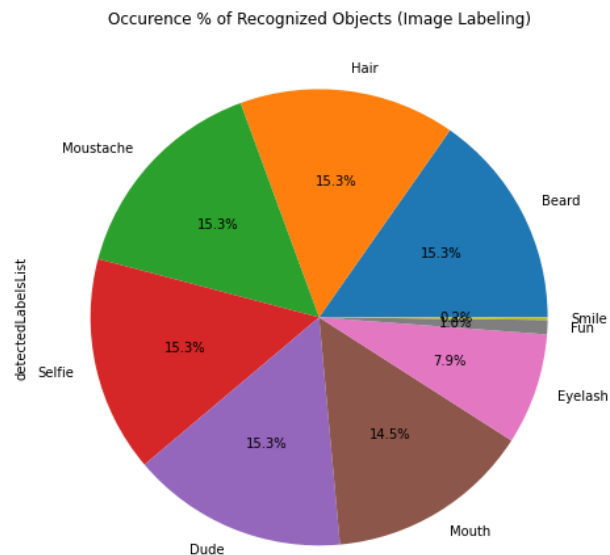


Σχήμα 4.18: Πίτες που εκφράζουν σε τι ποσοστό χαμογελούσε ο χρήστης κατά τη διάρκεια της μάθησης και σε τι ποσοστό κοιμόταν αντίστοιχα

Στο σχήμα 4.20 φαίνονται τα διαφορετικά αντικείμενα που αναγνωρίστηκαν κατά τη διαδικασία μάθησης μέσω κινητού από το ML Kit image labeling. Τα δεδομένα εμφανίζονται ανά γραμμή με μορφή λίστα και χρησιμοποιούμε την μέθοδο `explode()` του `pandas framework` ώστε να μπορέσουμε να μετατρέψουμε κάθε στοιχείο της λίστας σε μια γραμμή, αναπαράγοντας τις τιμές των δεικτών. Εν συνέχεια με `inline methods` και `ripped methods` χρησιμοποιούμε `group by` και `count values` ώστε να βρούμε τα διαφορετικά ποσοστά και να τα εμφανίσουμε σε κάποιο διάγραμμα.

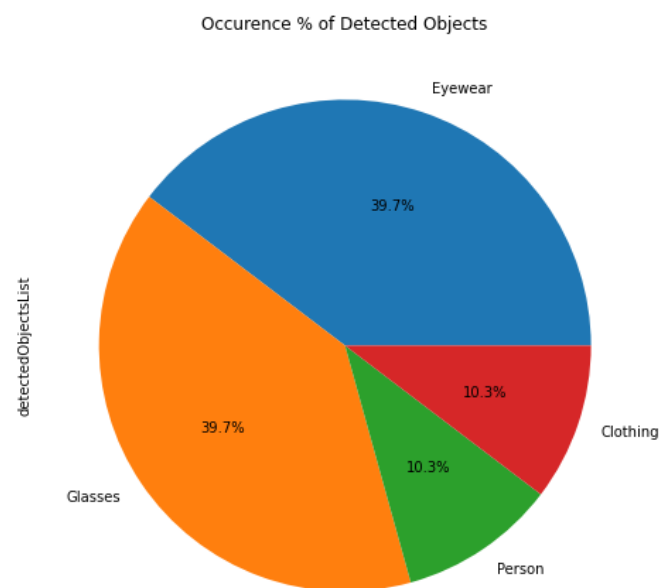
```
Python
df_analytics['detectedObjectsList'].explode().value_counts().plot.pie(figsize=(8,8),title='Occurence % of Detected Objects',autopct='%1.1f%%')
```

Σχήμα 4.19: Παράδειγμα εκτέλεσης κώδικα για την δημιουργία του παρακάτω διαγράμματος



Σχήμα 4.20: Διαφορετικά αντικείμενα που αναγνωρίστηκαν από το Image Labeling

Αντίστοιχα, εμφανίζουμε τα διαφορετικά αντικείμενα που αναγνωρίστηκαν από τον αλγόριθμο αναγνώρισης αντικειμένων.



Σχήμα 4.21: Διαφορετικά αντικείμενα που ανιχνεύθηκαν από το Object Detection

Τέλος δημιουργούμε ένα dataframe με τα μαθήματα 'Lessons' που έχει προσθέσει ο κάθε χρήστης για εκμάθηση και εμφανίζουμε κάποια βασικά τα στατιστικά.

	completed	timestamp	videoId	videoNote	videoTitle	videoURL
IHZwWFHwa-w	True	1664132799176	IHZwWFHwa-w	something about this video..	Gradient descent, how neural networks learn Chapter 2, Deep learning	NaN
Ilg3gGewQ5U	True	1664132811354	Ilg3gGewQ5U	NaN	What is backpropagation really doing? Chapter 3, Deep learning	NaN
O85OWBJ2ayo	True	1664132893652	O85OWBJ2ayo	NaN	How (and why) to raise e to the power of a matrix DE6	NaN
PFDu9oVAE-g	False	1664135327406	PFDu9oVAE-g	NaN	Eigenvectors and eigenvalues Chapter 14, Essence of linear algebra	NaN
ToiXSwZ1pJU	False	1664132881282	ToiXSwZ1pJU	NaN	Solving the heat equation DE3	NaN
aircAruvnKk	False	1664132783108	aircAruvnKk	NaN	But what is a neural network? Chapter 1, Deep learning	NaN
hl3bQySs8sM	True	1664721534393	hl3bQySs8sM	NaN	Euclidean Distance - Practical Machine Learning Tutorial with Python p.15	NaN
ly4S0oi3Yz8	False	1664132872130	ly4S0oi3Yz8	NaN	But what is a partial differential equation? DE2	NaN
m2MlpDrF7Es	True	1664818889466	m2MlpDrF7Es	NaN	What's so special about Euler's number e? Chapter 5, Essence of calculus	https://youtu.be/m2MlpDrF7Es
p_di4Zn4wz4	NaN	1664132856339	p_di4Zn4wz4	NaN	Differential equations, a tourist's guide DE1	NaN
r6sGWTCMz2k	False	1664132887658	r6sGWTCMz2k	NaN	But what is a Fourier series? From heat flow to drawing with circles DE4	NaN
tleHLnjs5U8	False	1664132818139	tleHLnjs5U8	Hjfd	Backpropagation calculus Chapter 4, Deep learning	NaN

Σχήμα 4.22: Dataframe διαφορετικών μαθημάτων

```
User videos: 12
Completed videos: 5(42.0%)
User Score: 500.0
```

Σχήμα 4.23: Εικόνα βασικών στατιστικών

4.2.2 MassAnalytics.ipynb

Για τη δημιουργία των MassAnalytics χρησιμοποιήσαμε αντίστοιχη λειτουργικότητα με αυτήν των UserAnalytics αλλά συμπεριλάβαμε το `userId` στο σχήμα των πινάκων.

	completed	timestamp	videoId	videoNote	videoTitle	videoURL	userId
0	True	1664132799176	IHZwWFHwa-w	something about this video..	Gradient descent, how neural networks learn Chapter 2, Deep learning	NaN	FZBzbUmViFg6nfpoDF45FPhsYBy1
1	True	1664132811354	Ilg3gGewQ5U	NaN	What is backpropagation really doing? Chapter 3, Deep learning	NaN	FZBzbUmViFg6nfpoDF45FPhsYBy1
2	True	1664132893652	O85OWBJ2ayo	NaN	How (and why) to raise e to the power of a matrix DE6	NaN	FZBzbUmViFg6nfpoDF45FPhsYBy1
3	False	1664135327406	PFDu9oVAE-g	NaN	Eigenvectors and eigenvalues Chapter 14, Essence of linear algebra	NaN	FZBzbUmViFg6nfpoDF45FPhsYBy1
4	False	1664132881282	ToiXSwZ1pJU	NaN	Solving the heat equation DE3	NaN	FZBzbUmViFg6nfpoDF45FPhsYBy1

Σχήμα 4.24: Dataframe διαφορετικών μαθημάτων ανά χρήστη

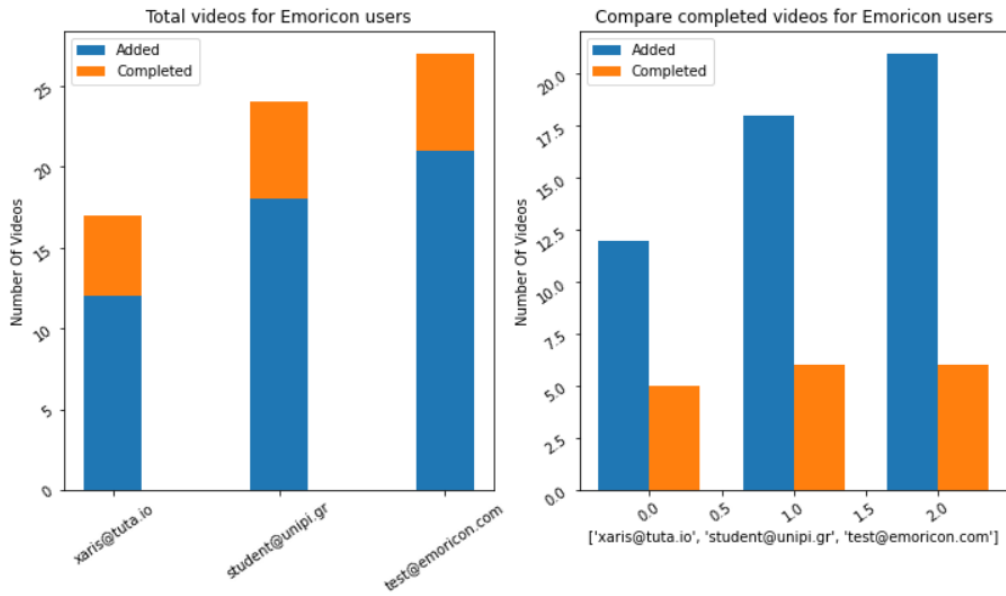
Στη συνέχεια με *data aggregation* δημιουργήσαμε αναλυτικά στοιχεία για το σύνολο των δεδομένων των χρηστών της εφαρμογής.

	completed	videoId	videoNote	percentage	score
FZBzbUmViFg6nfpoDF45FPhsYBy1	5	12	2	41.666667	60
aRiiifYUlyNk1ICgBnUI10yelos2	6	18	8	33.333333	108
sfYBluz3NsP1abnrN8duL30147V2	6	21	8	28.571429	126

...

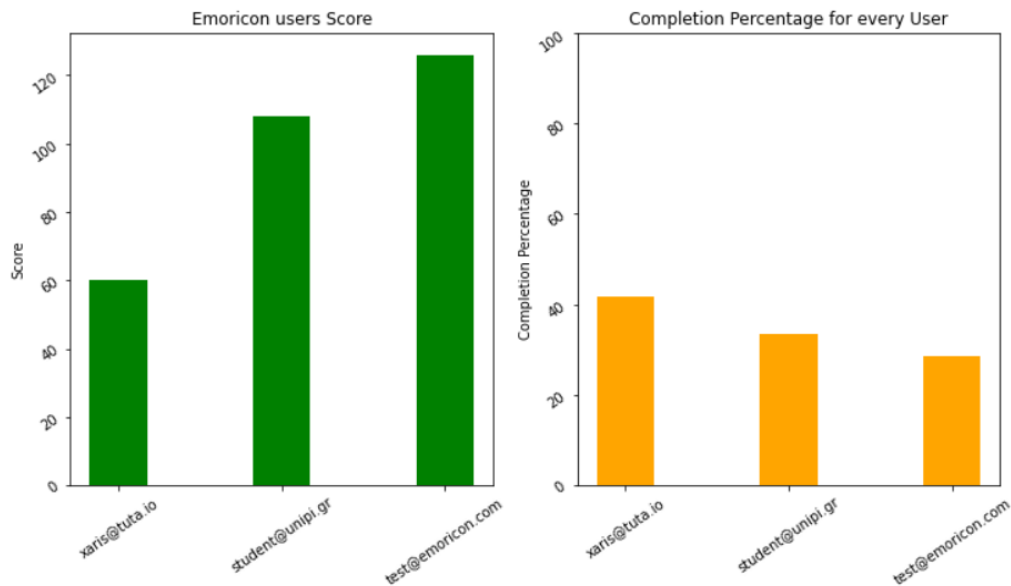
Σχήμα 4.25: Στατιστικά μαθημάτων ανά χρήστη

Ένα παράδειγμα αναλυτικών στοιχείων, είναι η πρόοδος των μαθητών αναφορικά με το πόσα μαθήματα έχουν παρακολουθήσει σε σχέση με αυτά που έχουν προσθέσει στην εφαρμογή.



Σχήμα 4.26: Στατιστικά μαθημάτων ανά χρήστη

Ακόμα παράλληλα με το ποσοστό ολοκλήρωσης των μαθημάτων θα μπορούσαμε να δούμε και το score που έχει πετύχει ο κάθε χρήστης.

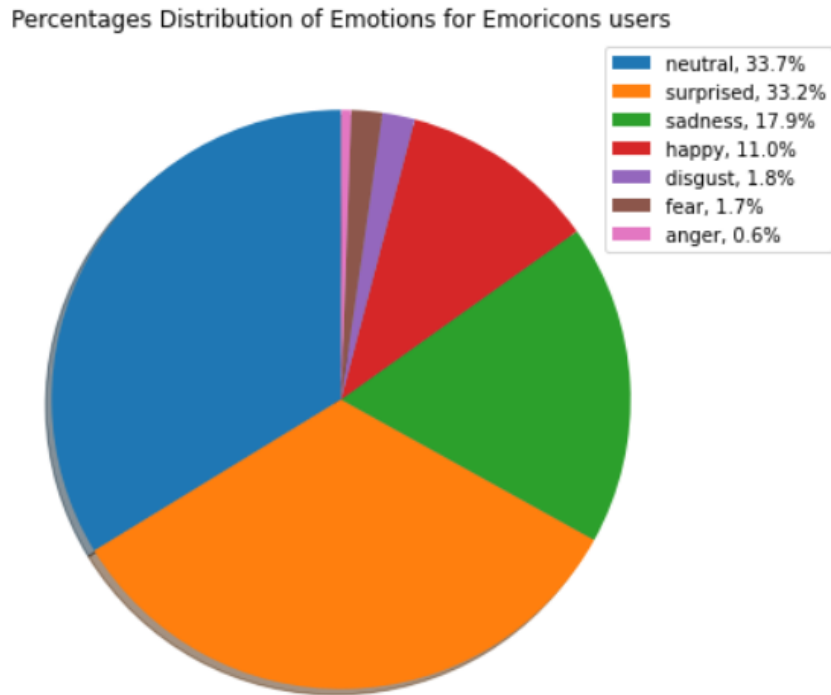


Σχήμα 4.27: Στατιστικά μαθημάτων ανά χρήστη

Ένα ακόμα πολύ ενδιαφέρον στατιστικό που αποτελεί και βάση της διπλωματικής είναι τα διαφορετικά συναισθήματα που αναγνωρίστηκαν κατά τη διαδικασία της μάθησης. Αναλυτικά μπορούμε να δούμε το επικρατέστερο συναίσθημα για κάθε χρήστη και ξεφεύγοντας από το ατομικό συναίσθημα μπορούμε πλέον να παρακολουθούμε το ποσοστό κατανομής συναισθημάτων στο σύνολο των χρηστών της εφαρμογής μας.

	userId	emotion	anotation	email
3	FZBzbUmVIFg6nfpoDF45FPhsYBy1	neutral	152	xaris@tuta.io
12	aRiilfYUlyNk1ICgBnUI10yelos2	surprised	252	student@unipi.gr
19	sfYBluz3NsP1abnrN8duL30147V2	surprised	977	test@emoricon.com

Σχήμα 4.28: Επικρατέστερο συναίσθημα ανά χρήστη



Σχήμα 4.29: Ποσοστιαία κατανομή συναισθημάτων εφαρμογής emoricon

5 Συζήτηση

5.1 Προσωπικά δεδομένα

Είναι πολύ σημαντικό να προσέχουμε τον τρόπο με τον οποίο διαχειριζόμαστε τα δεδομένα του χρήστη καθώς και τις πληροφορίες που συλλέγουμε από την συσκευή του. Η όλη διαδικασία θα πρέπει να χαρακτηρίζεται από διαφάνεια και να αποκαλύπτεται ο σκοπός και ο στόχος για τον οποίο γίνεται η συλλογή.

Βάση του Γενικού Κανονισμού για την Προστασία Δεδομένων (GDPR), προσωπικά δεδομένα είναι όλες αυτές οι πληροφορίες με τις οποίες μπορεί να γίνει (άμεσα ή έμμεσα) η αναγνώριση ενός ατόμου. Τα προσωπικά δεδομένα που συλλέγονται, πρέπει να είναι τα ελάχιστα απαραίτητα ώστε να επιτυγχάνεται ο σκοπός για τον οποίο προορίζονται και να διατηρούνται για το ελάχιστο δυνατό διάστημα. Ο σκοπός θα πρέπει να φανερώνεται στο χρήστη με σαφή τρόπο και θα πρέπει να ζητάμε τη συναίνεσή του πριν προχωρήσουμε στην επεξεργασία των δεδομένων του.

Η εφαρμογή Emoricon είναι μια εφαρμογή ακαδημαϊκού ενδιαφέροντος που συλλέγει δεδομένα και τα αναλύει προσπαθώντας αφενός να βοηθήσει τη διαδικασία εκμάθησης και να κρίνει την αποδοτικότητα του περιεχομένου και αφετέρου να αναδείξει τον τρόπο συλλογής των δεδομένων και τους βέλτιστους τρόπους επεξεργασίας ώστε αυτή να επιτυγχάνεται σε πραγματικό χρόνο.

6 Συμπεράσματα

Η συγκεκριμένη διπλωματική είχε ως στόχο την ανάπτυξη μίας εφαρμογής mobile learning για το android μέσω της οποίας ένας χρήστης (μαθητής) θα μπορεί να παρακολουθεί εκπαιδευτικά βίντεο και να κρατήσει σημειώσεις.

Ειδικά τα τελευταία χρόνια της πανδημίας, η τεχνολογία έδωσε λύσεις και βοήθησε να συνεχιστεί η καθημερινότητα των μαθητών προσπαθώντας να υποκαταστήσει την ανάγκη για φυσική παρουσία. Το πόσο όμως αποδοτική είναι η νέα πραγματικότητα και το πόσο αξιόπιστα επιτυγχάνεται η απομακρυσμένη μάθηση με τη χρήση υπολογιστών και κινητών συσκευών, είναι ερωτήματα που ενδέχεται να απασχολήσουν στο μέλλον, δεδομένου ότι δεν έχουν ξεκάθαρη απάντηση.

Προσπαθώντας να βοηθήσουμε την εκπαιδευτική διαδικασία με την εφαρμογή μας, φροντίζουμε παράλληλα με τη θέαση να χρησιμοποιούμε την κάμερα του κινητού για να παίρνουμε εικόνες σε κάθε frame και να τις εισάγουμε σε μοντέλα μηχανικής μάθησης για να εξάγουμε συμπεράσματα. Με αυτά τα δεδομένα δημιουργούμε πληροφορίες, τις οποίες αναλύουμε ατομικά για κάθε μαθητή ή συλλογικά για ένα σύνολο ατόμων που παρακολούθησαν συγκεκριμένα μαθήματα, και οι οποίες διατίθενται στους εκπαιδευτικούς. Τέτοιες πληροφορίες είναι το συναίσθημα του μαθητή, το πλαίσιο στο οποίο λαμβάνει χώρα η εκμάθηση, η δραστηριότητά του, τα αντικείμενα που υπάρχουν στο χώρο, η τοποθεσία του και άλλα δεδομένα από τους αισθητήρες της συσκευής του.

Τα analytics που παρουσιάσαμε στο κεφάλαιο 4, αποτελούν παραδείγματα από dummy data που εισήχθησαν ενδεικτικά και δεν αποτελούν πραγματικά δεδομένα. Εξάλλου σκοπός είναι να παρουσιαστούν οι διαφορετικές πληροφορίες που μπορούν να εξαχθούν και όχι να τις αξιοποιήσουμε για να βγάλουμε συμπεράσματα για την μάθηση μέσω κινητού.

Τα παρέχουμε στον εκπαιδευτικό για να μπορέσει να κρίνει το κατά πόσον η εκπαιδευτική διαδικασία μέσω των βίντεο ήταν αποδοτική. Καταγράφουμε το συναίσθημα του μαθητή διότι είναι δεδομένο ότι ένας μαθητής που αντιδρά συναισθηματικά, θετικά είτε αρνητικά, απέναντι σε κάτι που παρατηρεί, έχει πολύ περισσότερες πιθανότητες να συγκρατήσει μια πληροφορία, σε σχέση με κάποιον ο οποίος παραμένει απαθής. Συλλέγουμε δεδομένα για το περιβάλλον (context) στο οποίο λαμβάνει χώρα η εκπαίδευση μέσω κινητού, και το τι υπάρχει στο χώρο γιατί όπως είναι λογικό, ένας μαθητής που μελετάει μόνος του σε ένα ήσυχο περιβάλλον δεν έχει περισπασμούς από εξωτερικούς παράγοντες. Η δραστηριότητά του, η ώρα της μάθησης ακόμα και η θερμοκρασία του χώρου είναι παράγοντες που ενδέχεται να επηρεάζουν τη δυνατότητα μας να συγκρατήσουμε τη γνώση. Ακόμα ελέγχοντας τα μαζικά analytics, μπορούν να βγουν συμπεράσματα για τη μεταδοτικότητα του εκπαιδευτικού κατά τη διαδικασία της μάθησης και για το αν υπήρχε αλληλεπίδραση με τους μαθητές. Αυτά τα συμπεράσματα μπορούν να βοηθήσουν να βελτιωθεί το περιεχόμενο ώστε να γίνει αποδοτικότερο.

Μια τεχνολογία ικανή να αναγνωρίζει συναισθήματα με τον ίδιο τρόπο που το κάνουν οι άνθρωποι δεν έχει εφαρμογή μόνο στο mobile learning αλλά και στην αλληλεπίδραση ανθρώπου-υπολογιστή, στις τεχνολογίες υποβοήθησης του ανθρώπου και στην διαδικτυακή εκπαίδευση. Παρόλα αυτά η συλλογή δεδομένων είναι πολύ ευκολότερη μέσω μιας συσκευής κινητού και τα προσωπικά μας δεδομένα μας πολύτιμα για αυτό οφείλουμε να τα προστατέψουμε. Όπως ανέφερε και ο Έντουαρντ Σνόουντεν, «το να υποστηρίζεις ότι δεν σε ενδιαφέρει το δικαίωμα σου στην ιδιωτι-

κότητα επειδή δεν έχεις τίποτα να κρύψεις, είναι σαν να λες ότι δεν σε ενδιαφέρει η ελευθερία του λόγου επειδή δεν έχεις τίποτα να πεις».

Α΄ Πίνακας ορολογίας

Απόδοση

γνώρισμα
διαπροσωπεία
μάθηση διαμέσου κινητών συσκευών
ενσωματωμένο
αναλυτικά στοιχεία
περιβάλλον
μετρικές
σχολιασμός
πλαίσιο
αντικείμενο
σταθερές
τάξη
σύνολο δεδομένων
ταξινόμηση
επισημανση
σιγμοειδής
οριοθετημένο
ορόσημο
ιχνηλάτηση
συνάρτηση απώλειας με χρήση
τριάδων εικόνων

Ξενόγλωσσος όρος

attribute
interface
mobile learning
built-in
analytics
context
metrics
annotation
framework
object
constants
class
dataset
classification
labeling
sigmoid
bounding
landmark
tracking

triplet loss function

Β' Ακρωνύμια και συντομογραφίες

API	Application Programming Interface
AI	Artificial Intelligence
ML	Machine Learning
DL	Deep Learning
NN	Neural Network
MLP	Multilayer Perceptron
DNN	Deep Neural Network
FACS	Facial Action Coding System
GPS	Global Positioning System
LSTM	Long Short Term Memory
E-GAN	Evolutionary Generative Adversarial Network
ANI	Artificial Narrow Intelligence
AGI	Artificial General Intelligence
ASI	Artificial Super Intelligence
CNN	Convolutional Neural Network
ANN	Artificial Neural Network
SIANN	Space Invariant Artificial Neural Network
R-CNN	Region Based Convolutional Neural Network
CPU	Central Processing Unit
GPU	Graphics Processing Unit
SSD	Single Shot Detection
YOLO	You Only Look Once
FN	Facenet
TF	TensorFlow
REST	Representational State Transfer
RPC	Remote Procedure Call
URL	Uniform Resource Locator
GDPR	General Data Protection Regulation
βλ.	βλέπε
κ.α.	και άλλα
κ.λπ.	και λοιπά

Bibliography

- [1] M. Piyush and M. Samaya, “An introduction to deep learning.” <https://developer.ibm.com/articles/an-introduction-to-deep-learning/>, 2020. Accessed: 2022-02-16.
- [2] K. Chethan, “Artificial intelligence: Definition, types, examples, technologies.” <https://chethankumargn.medium.com/artificial-intelligence-definition-types-examples-technologies-962ea75c7b9b>. Accessed: 2022-02-20.
- [3] J. Peng, E. Jury, P. Dönnies, and C. Ciurtin, “Machine learning techniques for personalised medicine approaches in immune-mediated chronic inflammatory diseases: Applications and challenges,” *Frontiers in Pharmacology*, vol. 12, 09 2021.
- [4] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. <http://www.deeplearningbook.org>.
- [5] “Emotionrecognition model.” <https://github.com/lampadovnikita/EmotionRecognition>. Accessed: 2021-11.
- [6] “Scene detection model.” <https://tfhub.dev/sayannath/lite-model/image-scene/1>. Accessed: 2022-04.
- [7] “Google developers: Object detection.” <https://developers.google.com/ml-kit/vision/object-detection>. Accessed: 2022-01.
- [8] X. Gu, Y. Zhu, and X. Guo, “Meeting the “digital natives”: Understanding the acceptance of technology in classrooms,” *Journal of Educational Technology and Society*, vol. 16, no. 1, pp. 392–402, 2013.
- [9] P. Ekman, “Emotions revealed, recognizing faces and feelings to improve communication and emotional life,” 2003.
- [10] J. Dow, *Aristotle’s Theory of the Emotions: Emotions as Pleasures and Pains*, pp. 47–74. Times Books, 2011.
- [11] W. A. Mason and J. P. Capitanio, “Basic emotions: A reconstruction,” *Emotion Review*, vol. 4, no. 3, pp. 238–244, 2012. PMID: 27110280.
- [12] R. Plutchik, “The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice,” *American Scientist*, vol. 89, no. 4, pp. 344–350, 2001.
- [13] W. G. Parrott, *Emotions in Social Psychology*. Psychology Press, 2001.
- [14] P. Ekman and W. Friesen, “Constants across cultures in the face and emotions,” vol. 17, no. 2, pp. 124–129, 1971.
- [15] R. Jack, W. Sun, I. Delis, O. Garrod, and P. Schyns, “Four not six: Revealing culturally common facial expressions of emotion,” pp. 708–730, 2016.
- [16] A. Freitas-Magalhães, “Facial expression of emotion,” in *Encyclopedia of Human Behavior (Second Edition)* (V. Ramachandran, ed.), pp. 173–183, San Diego: Academic Press, second edition ed., 2012.
- [17] A. Cowen, D. Keltner, and F. Schroff, “Sixteen facial expressions occur in similar contexts worldwide,” *Nature*, vol. 589, pp. 251–257, 2021.

- [18] I. Vlahavas, P. Kefalas, N. Bassiliades, F. Kokkoras, and I. Sakellariou, *Artificial Intelligence*. University of Macedonia Press, third edition ed., 2011.
- [19] J. McCarthy, “What is artificial intelligence?.” <http://www-formal.stanford.edu/jmc/whatisai/whatisai.html>, 1997. Accessed: 2022-02-10.
- [20] H. Arend, “Understanding the four types of artificial intelligence.” <https://www.govtech.com/computing/understanding-the-four-types-of-artificial-intelligence.html>, 2016. Accessed: 2022-02-17.
- [21] J. Johnson, “Artificial intelligence types.” <https://www.bmc.com/blogs/artificial-intelligence-types/>, 2020. Accessed: 2022-02-17.
- [22] B. Marr, “Understanding the four types of artificial intelligence.” <https://bernardmarr.com/understanding-the-4-types-of-artificial-intelligence/>. Accessed: 2022-02-17.
- [23] T. Panagiotopoulos and G. Anastasakis, *Logic Programming*. University of Piraeus, third edition ed., 2012.
- [24] A. Likas, *Τεχνητά Νευρωνικά Δίκτυα - Εφαρμογές*. Greek Open University, 2009.
- [25] S. Smith, ““convolution”. the scientist and engineer’s guide to digital signal processing,” 1997.
- [26] H. Zhang, Z. Qin, and H. Xie, “Image recognition and detection based on fast area convolutional neural network,” *Journal of Physics: Conference Series*, vol. 1976, jul 2021.
- [27] W. Zhang, K. Itoh, J. Tanida, and Y. Ichioka, “Parallel distributed processing model with local space-invariant interconnections and its optical architecture,” *Appl. Opt.*, vol. 29, pp. 4790–4797, Nov 1990.
- [28] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 1, pp. I-I, 2001.
- [29] Z. Jun, H. Jizhao, T. Zhenglan, and W. Feng, “Face detection based on lbp,” in *2017 13th IEEE International Conference on Electronic Measurement & Instruments (ICEMI)*, pp. 421–425, 2017.
- [30] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C. Fu, and A. C. Berg, “SSD: single shot multibox detector,” *CoRR*, vol. abs/1512.02325, 2015.
- [31] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788, 2016.
- [32] V. Bazarevsky, Y. Kartynnik, A. Vakunov, K. Raveendran, and M. Grundmann, “Blazeface: Sub-millisecond neural face detection on mobile gpus,” *CoRR*, vol. abs/1907.05047, 2019.
- [33] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” *CoRR*, vol. abs/1311.2524, 2013.

- [34] S. Ren, K. He, R. B. Girshick, and J. Sun, “Faster R-CNN: towards real-time object detection with region proposal networks,” *CoRR*, vol. abs/1506.01497, 2015.
- [35] M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L. Chen, “Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation,” *CoRR*, vol. abs/1801.04381, 2018.
- [36] F. Schroff, D. Kalenichenko, and J. Philbin, “Facenet: A unified embedding for face recognition and clustering,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 815–823, 2015.
- [37] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” *CoRR*, vol. abs/1704.04861, 2017.
- [38] S. Chen, Y. Liu, X. Gao, and Z. Han, “Mobilefacenets: Efficient cnns for accurate real-time face verification on mobile devices,” *CoRR*, vol. abs/1804.07573, 2018.
- [39] C. O’Malley, G. Vavoula, J. Glew, M. Sharples, and P. Lefrere, “Mobilearn: D4.1 guidelines for learning/teaching/tutoring in a mobile environment,” 01 2005.
- [40] C. Pimmer and N. Pachler, *Mobile learning in the workplace. Unlocking the value of mobile technology for work-based education*, pp. 193–204. 01 2014.
- [41] C. Quinn, “mlearning: Mobile, wireless, in-your- pocket learning,” 07 2001.
- [42] “Google developers: Machine learning kit.” <https://developers.google.com/ml-kit>. Accessed: 2021-12-10.
- [43] “Google developers:activity recognition.” <https://developers.google.com/location-context/activity-recognition>. Accessed: 2021-12.
- [44] “Google developers: Fused location provider.” <https://developers.google.com/location-context/fused-location-provider>. Accessed: 2022-06.
- [45] “Google firebase.” <https://firebase.google.com/>. Accessed: 2022-03.
- [46] “Firebase console.” <https://console.firebase.google.com/>. Accessed: 2022-01.
- [47] “Tensor flow lite.” <https://www.tensorflow.org/lite>. Accessed: 2022-04.
- [48] “Youtube data api.” <https://developers.google.com/youtube/v3>. Accessed: 2022-04.
- [49] “Android developers:camerax.” <https://developer.android.com/training/camerax>. Accessed: 2022-02.
- [50] “Android developers: Surface object.” <https://developer.android.com/reference/android/view/Surface.html>. Accessed: 2022-05.
- [51] “Google developers:face detection.” <https://developers.google.com/ml-kit/vision/face-detection/android>. Accessed: 2022-02.

- [52] “Real-time face detection.” https://developers.google.com/ml-kit/vision/face-detection/android#real-time_face_detection. Accessed: 2022-04.
- [53] lampadovnikita, “Emotion recognition model license.” <https://github.com/lampadovnikita/EmotionRecognition/blob/master/LICENSE>. Accessed: 2022-02.
- [54] “Jaffe dataset.” <https://zenodo.org/record/3451524>. Accessed: 2022-02.
- [55] “Ck dataset.” https://www.researchgate.net/publication/224165246_The_Extended_Cohn-Kanade_Dataset_CK_A_complete_dataset_for_action_unit_and_emotion-specified_expression. Accessed: 2022-02.
- [56] “Fer2013 dataset.” <https://www.kaggle.com/deadskull17/fer2013>. Accessed: 2022-02.
- [57] “Raf-db dataset.” <http://whdeng.cn/RAF/model1.html>. Accessed: 2022-02.
- [58] A. Pouget, S. Ramesh, M. Giang, R. Chandrapalan, T. Tanner, M. Prussing, R. Timofte, and A. Ignatov, “Fast and accurate camera scene detection on smartphones.”
- [59] “Creative commons license.” <https://creativecommons.org/licenses/by/3.0/>. Accessed: 2022-02.
- [60] “Scene detection model (google colab).” https://colab.research.google.com/github/sayannath/Image-Scene-TF-Hub/blob/main/src/Image_Scene_TFLite.ipynb. Accessed: 2022-03.
- [61] “Apache-2.0 license.” <https://opensource.org/licenses/Apache-2.0>. Accessed: 2022-02.
- [62] “Google developers: Image labeling.” <https://developers.google.com/ml-kit/vision/image-labeling>. Accessed: 2022-01.
- [63] “Android developers: Splash screen.” <https://developer.android.com/develop/ui/views/launch/splash-screen>. Accessed: 2022-03.
- [64] “Google developers: Cardview.” <https://developer.android.com/develop/ui/views/layout/cardview>. Accessed: 2022-05.
- [65] “Android developers: Recyclerview.” <https://developer.android.com/develop/ui/views/layout/recyclerview>. Accessed: 2022-04.
- [66] PierfrancescoSoffritti, “Opensource youtube player for android.” <https://github.com/PierfrancescoSoffritti/android-youtube-player>. Accessed: 2022-04.
- [67] “Google developers: I-frame api.” https://developers.google.com/youtube/iframe_api_reference. Accessed: 2022-05.