

Ατομική Διπλωματική Εργασία

**ΧΡΗΣΗ ΤΕΧΝΙΚΩΝ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ ΓΙΑ ΤΗΝ ΔΙΑΜΟΡΦΩΣΗ
ΔΙΚΤΥΩΝ LORA.**

Ανδρέας Χατζηβασίλη

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2022

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Χρήση τεχνικών Μηχανικής Μάθησης για την διαμόρφωση δικτύων LoRa.

Ανδρέας Χατζηβασίλη

Επιβλέπων Καθηγητής

Δρ. Βάσος Βασιλείου

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου

Μάιος 2022

Ευχαριστίες

Αρχικά θα ήθελα να ευχαριστήσω όλα τα άτομα που με βοήθησαν και με στήριξαν στην υλοποίηση της διπλωματικής μου εργασίας. Ήταν ένα θέμα καινούργιο τόσο για εμένα όσο και στην ευρύτερη κοινότητα της Πληροφορικής για αυτό και η βοήθεια και οι συμβουλές που έλαβα από όλους αυτούς τους ανθρώπους ήταν τα θεμέλιά για την επιτυχή υλοποίηση της εργασίας.

Ξεχωριστά, θα ήθελα να ευχαριστήσω θερμά τον επιβλέπων καθηγητή μου Δρ. Βάσο Βασιλείου, ο οποίος ήταν καθοδηγητής σε κάθε βήμα της διπλωματικής εργασίας, από την εύρεση του θέματος μέχρι και την συγγραφή της διπλωματικής εργασίας.

Τέλος, θα ήθελα να ευχαριστήσω και την Κρίστια Χαριλάου η οποία ως επιβλέπων ερευνήτρια η βοήθεια και οι κατευθυντήριες γραμμές που άντλησα από την συνεργασίας μας ήταν σημαντικές για την επίλυση μεγάλου αριθμού προβλημάτων.

Περίληψη

Το διαδίκτυο των Πραγμάτων (Internet of Things - IoT) και η Μηχανική μάθηση είναι δύο από τους πιο ραγδαία αναπτυσσόμενους τομείς του κλάδου της Πληροφορικής. Η μηχανική μάθηση αποτελεί την κύρια πηγή λύσεων σε σύγχρονα προβλήματα λόγω των υψηλών επιδόσεων αλλά και της σταθερότητας που προσφέρει. Η χρήση των δικτύων LoRa για την αποστολή πληροφοριών αποτελεί μια κύρια πτυχή λειτουργίας του διαδικτύου των πραγμάτων αφού αποτελεί το δίκτυο LPWAN (Low Power Wide Area Networks) που χρησιμοποιούν οι πλείστες εφαρμογές διαδικτύου των πραγμάτων.

Στόχος αυτής της διπλωματικής εργασίας ήταν η χρήση σύγχρονων τεχνικών μηχανικής μάθησης για επίλυση κάποιων μειονεκτημάτων που εντοπίστηκαν στην διαχείριση και διαμόρφωση των πόρων δικτύων LoRa. Τα μειονεκτήματα αυτά εντοπίστηκαν στην αδυναμία του αλγορίθμου ADR (Adaptive Data Rate) να λειτουργήσει κάτω από μη σταθερές συνθήκες ραδιοκυμάτων. Επίσης, ο τρόπος λειτουργίας του αλγορίθμου προϋποθέτει την ύπαρξη 20 τουλάχιστον πακέτων προς τον κεντρικό εξυπηρετητή δικτύου κάτι που καθυστερεί την πλήρη ενεργοποίησή του. Με την εργασία αυτή προσπαθήσαμε να παρουσιάσουμε μια εναλλακτική λύση η οποία να καλύπτει αυτά τα δύο κενά που παρουσιάζει ο αλγόριθμος.

Οι πιο πάνω αδυναμίες οδήγησαν στην δημιουργία ενός μοντέλου μηχανικής μάθησης για τελικές συσκευές με την χρήση του tinyML. Η εύρεση του κατάλληλου συνόλου δεδομένων αλλά και αλγορίθμων δημιουργίας του μοντέλου είναι ζητήματα που θα αναλυθούν στην συνέχεια. Έπειτα, έχουμε την αξιολόγηση του αλγορίθμου με την χρήση προσομοιωτή συγκριτικά με υφιστάμενους αλγόριθμους. Τα αποτελέσματα και τα συμπεράσματα της δουλειάς και της λειτουργίας του μοντέλου θα αναλυθούν στο τέλος του εγγράφου.

Περιεχόμενα

Κεφάλαιο 1: Εισαγωγή	7
1.1 Γενική Εισαγωγή.....	7
1.2 Προκλήσεις στο διαδίκτυο των Πραγμάτων	7
1.3 Σκοπός της Διπλωματικής Εργασίας.....	8
1.4 Σχετικές δουλειές	9
Κεφάλαιο 2: Υπόβαθρο Δικτύων LoRa	11
2.1 Δίκτυα LoRa	11
2.2 Πρωτόκολλο LoRaWAN	13
2.2.1 Γενικές Πληροφορίες	13
2.2.2 Τοπολογία	14
2.2.3 Κατηγορίες τελικών συσκευών.....	14
2.2.4 Κατηγορίες μηνυμάτων	15
2.2.5 Συντελεστής Διασποράς	16
2.2.6 Ρυθμός μεταφοράς δεδομένων	17
Κεφάλαιο 3: Υφιστάμενοι Αλγόριθμοι	18
3.1 Υφιστάμενοι Αλγόριθμοι - ADR.....	18
3.2 Προβλήματα Υφιστάμενων Αλγορίθμων	20
3.3 Λύση LoRaWAN – Blind ADR	21
Κεφάλαιο 4: Μεθοδολογία.....	24
4.1 Μηχανική Μάθηση σε τελικές συσκευές	24
4.2 Μεθοδολογία ανάπτυξης tinyML μοντέλων	24
4.3 Υλοποίηση πρότυπου μοντέλου tinyML	26
Κεφάλαιο 5: Υλοποίηση Αλγορίθμου	30
5.1 Γενικές Πληροφορίες Υλοποίησης	30
5.2 Αρχικό σύνολο Δεδομένων	32
5.3 Δημιουργία Μοντέλου	37
5.4 Επέκταση δεδομένων	39
5.5 Αποτελέσματα με τα νέα Δεδομένα	41
Κεφάλαιο 6: Αξιολόγηση Αλγορίθμου	44
6.1 Αξιολόγηση αλγορίθμου	44
6.2 Χρήση προσομοιωτή LoRaMab	44

6.3 Επέκταση προσομοιωτή LoRaMab	46
6.4 Χρήση του προσομοιωτή και άντληση δεδομένων	47
6.5 Αποτελέσματα Αλγορίθμων	49
Κεφάλαιο 7: Συμπεράσματα.....	54
7.1 Συμπεράσματα	54
7.2 Μελλοντική Δουλειά	56
Βιβλιογραφία.....	57

Κεφάλαιο 1

Εισαγωγή

1.1 Γενική Εισαγωγή.....	7
1.2 Προκλήσεις στο διαδίκτυο των Πραγμάτων	7
1.3 Σκοπός της Διπλωματικής Εργασίας.....	8
1.4 Σχετικές δουλειές	9

1.1 Γενική Εισαγωγή

Το διαδίκτυο των Πραγμάτων είναι ένας ραγδαία αναπτυσσόμενος τομέας της πληροφορικής, ευρέως διαδεδομένος τόσο σε περιβάλλοντα οργανισμών, πόλεων αλλά και σπιτιών. Η ανάγκη για έξυπνες συσκευές έχει αλλάξει κατά πολύ τα τελευταία χρόνια με αποτέλεσμα την δημιουργία περισσότερων και πιο συγκεκριμένων συσκευών εξυπηρέτησης των αναγκών αυτών. Οι τεχνολογίες αυτές χρησιμοποιούνται για την ανάπτυξη και ευχρηστία υπηρεσιών, για την πράσινη ανάπτυξη, για την ασφάλεια κ.ο.κ.

Παράλληλα, η υλοποίηση και διαμόρφωση τέτοιων δικτύων απαιτεί την χρήση των αντίστοιχων δικτύων που να παρέχουν τις δυνατότητες αποστολής δεδομένων με χαμηλή κατανάλωση ενέργειας και σε μεγάλες αποστάσεις. Τα δίκτυα LPWAN είναι τα δίκτυα που φέρνουν την λύση στα παραπάνω προβλήματα με αρκετές τεχνολογίες να το διαδέχονται ανάμεσα τους τα DASH7, Wize, Sigfox αλλά και τα δίκτυα LoRa που θα μελετήσουμε με περισσότερη λεπτομέρεια.

1.2 Προκλήσεις στο διαδίκτυο των Πραγμάτων

Η εμφάνιση του διαδικτύου των πραγμάτων έχει εισάγει νέες προκλήσεις που δεν μπορούν να αντιμετωπιστούν από την τρέχουσα διαθέσιμη συνδεσιμότητα και πρωτόκολλα. Πιο κάτω, συζητούνται αρκετές βασικές προκλήσεις του διαδικτύου των πραγμάτων που έχουμε να αντιμετωπίσουμε σήμερα [18].

Εύρος ζώνης/Ρυθμός μετάδοσης δεδομένων: Το εύρος ζώνης και ο ρυθμός μεταφοράς δεδομένων είναι δύο αλληλένδετες έννοιες οι οποίες χρησιμοποιούνται για να περιγράψουν την ποσότητα των δεδομένων που μπορούν να μεταφερθούν μέσα σε ένα χρονικό διάστημα μέσα στο δίκτυο. Η εύρεση των κατάλληλων αυτών παραμέτρων έτσι ώστε να υπάρχει εξισορρόπηση αυτών με τις υπόλοιπες παραμέτρους όπως του εύρους κάλυψης είναι μία από τις κυριότερες προκλήσεις που έχουμε να αντιμετωπίσουμε.

Διάρκεια μπαταρίας: Πολλές από τις προκλήσεις που αντιμετωπίζουμε στο διαδίκτυο των πραγμάτων περικλείονται γύρο από την αποθήκευση ενέργειας στις έξυπνες συσκευές αφού η ανάγκη αυτές δεν μπορούν να ικανοποιηθούν από καμία υφιστάμενη τεχνολογία [19]. Οι πλείστες συσκευές τροφοδοτούνται από μη επαναφορτιζόμενες μπαταρίες για λόγους κόστους, διαθεσιμότητας και ευκολίας. Ωστόσο, η ανάγκη για αντικατάσταση λόγω περιορισμένης ενέργειας και των οικολογικών επιπτώσεων, θα αποτελέσει σοβαρό πρόβλημα για την τροφοδοσία δισεκατομμυρίων συσκευών, που χρησιμοποιούν μη επαναφορτιζόμενες μπαταρίες ως κύρια αποθήκευση ενέργειας.

Εύρος κάλυψης: Οι νέες τεχνολογίες έχουν στόχο να παρέχουν πρόσβαση πληροφοριών στο Διαδίκτυο σε άτομα/αντικείμενα απομακρυσμένες περιοχές. Για την επίτευξη μεγαλύτερων αποστάσεων, η δύναμη εκπομπής ραδιοκυμάτων θα πρέπει να αυξηθεί, προκαλώντας μεγαλύτερη κατανάλωση ενέργειας. Έτσι, τα νέα πρωτόκολλα στοχεύουν στην κάλυψη μεγαλύτερων αποστάσεων με χαμηλότερη κατανάλωση ενέργειας, ακολουθώντας την προσέγγιση της πράσινης ενέργειας.

Καθυστέρηση: Επί του παρόντος, εφαρμογές όπως τα μέσα κοινωνικής δικτύωσης έχουν αυστηρές απαιτήσεις σχετικά με την καθυστέρηση αποστολής των πακέτων. Οι νέες τεχνολογίες 5η γενιάς στοχεύουν στην αντιστάθμιση της καθυστέρησης επικοινωνίας έναντι της διάρκειας της μπαταρίας με την χρήση κλάσεων ποιότητας εξυπηρέτησης (QoS – Quality of Service).

1.3 Στόχος της Διπλωματικής Εργασίας

Στόχος της διπλωματικής εργασίας αυτής είναι η συνεισφορά στην επίλυση κάποιων από των πιο πάνω προκλήσεων του διαδικτύου των πραγμάτων με την χρήση καινοτόμων τεχνολογιών. Η χρήση τεχνητής νοημοσύνης για επίλυση τέτοιων προκλήσεων είναι μια λύση η οποία τα την τελευταία δεκαετία έχει αποδείξει ότι μπορεί να επιφέρει τα επιθυμητά αποτελέσματα με την ύπαρξη των κατάλληλων προϋποθέσεων. Πιο συγκεκριμένα, η χρήση τεχνικών μηχανικής

μάθησης είναι ευρέως διαδεδομένη σε τεράστιο αριθμό εφαρμογών με βασικό μειονέκτημα της, την ανάγκη για μεγάλη υπολογιστική ισχύ.

Το διαδίκτυο των πραγμάτων ωστόσο είναι μία τεχνολογία η οποία αποτελείται από συσκευές οι οποίες έχουν ελάχιστη υπολογιστική ισχύ, αντλώντας ενέργεια από μπαταρίες. Επομένως, ενώ η χρήση αλγορίθμων μηχανικής μάθησης μπορεί να αποτελέσει μία λύση στις προκλήσεις αυτές, ο περιορισμός της ισχύς δεν το έκανε εφικτό. Μια νέα και υποσχόμενη τεχνολογία όμως έχει κάνει την εμφάνιση της τα τελευταία χρόνια η οποία έρχεται να δώσει λύση στο πρόβλημα αυτό. Η τεχνολογία `tinyML` έχει απώτερο σκοπό να δώσει την δυνατότητα δημιουργίας και εκτέλεσης μοντέλων μηχανικής μάθησης σε συσκευές με περιορισμό στην υπολογιστική ισχύ και την ενεργειακή διαθεσιμότητα. Παρόλο που η τεχνολογία αυτή βρίσκεται σε πρώιμο στάδιο και είναι υπό ανάπτυξη, στην διπλωματική αυτή εργασία προχωρήσαμε στην υιοθέτησή της και εφαρμογής της στην βελτιστοποίηση της επίδοσης των δικτύων `LoRa` τα οποία θα επεξηγηθούν στο επόμενο κεφάλαιο.

Με την χρήση λοιπόν της τεχνολογίας `tinyML` προχωρήσαμε στην δημιουργία μοντέλων μηχανικής μάθησης τα οποία μπορούν να λειτουργήσουν σε συσκευές `IoT`. Με αυτήν την μεθοδολογία στόχος μας είναι να προσφέρουμε μια λύση αποκεντρωμένη (*decentralized*) η οποία θα δρα ξεχωριστά σε κάθε κόμβο του δικτύου ανεξάρτητα. Η προσέγγιση αυτή θα μειώσει τυχόν συμφόρηση του δικτύου από την υπερφόρτωση του κεντρικού εξυπηρετητή δικτύου όπως παράλληλα θα βελτιστοποιεί την χρήση Εύρους ζώνης και του ρυθμού μετάδοσης δεδομένων από την ίδια την συσκευή. Η προσέγγισή θα έχει ως στόχο επίσης την ελαχιστοποίηση της ενεργειακής κατανάλωσης των συσκευών.

1.4 Σχετικές δουλειές

Η ιδιαιτερότητα της διπλωματικής αυτής εργασίας κρύβεται στο ότι δεν υπάρχει κάτι αντίστοιχο με την δική μας προσπάθεια. Ενώ η βελτιστοποίηση τόσο του Εύρους ζώνης και του Ρυθμού μετάδοσης δεδομένων παράλληλα με την διατήρηση χαμηλών αριθμών κατανάλωσης ενέργειας, είναι μια πρόκληση για την οποία έχουν παρατεθεί αρκετές προτάσεις, καμία από αυτές δεν εφαρμόζει αλγορίθμους τεχνητής νοημοσύνης σε τελικές συσκευές.

Συγκεκριμένα έχουν προταθεί αρκετές λύσεις που κάνουν χρήση αλγορίθμων μηχανικής μάθησης στην πλευρά του κεντρικού εξυπηρετητή για την αντικατάσταση των υφιστάμενων αλγορίθμων. Μια τέτοια λύση βρίσκει εύκολα χώρο για την υλοποίηση της σε ένα εξυπηρετητή

λόγο της υψηλής ισχύς επεξεργασίας που παράγει μία τέτοια μηχανή. Επιπρόσθετα, τεχνικές όπως δυναμικός προγραμματισμός (dynamic programming) και άπληστοι αλγόριθμοι (greedy algorithms) είναι τεχνικές οι οποίες έχουν χρησιμοποιηθεί για την βελτιστοποίηση πληθώρας παραμέτρων τόσο στον κεντρικό επεξεργαστή όσο και σε τελικές συσκευές. Έτσι, η δουλειά που θα παρουσιάσουμε πιο κάτω αποτελεί μια πλήρως καινοτόμα προσέγγιση στα θέματα του διαδικτύου των πραγμάτων.

Κεφάλαιο 2

Υπόβαθρο Δικτύων LoRa

2.1 Δίκτυα LoRa	11
2.2 Πρωτόκολλο LoRaWAN	13
2.2.1 Γενικές Πληροφορίες	13
2.2.2 Τοπολογία	14
2.2.3 Κατηγορίες τελικών συσκευών	14
2.2.4 Κατηγορίες μηνυμάτων	15
2.2.5 Συντελεστής Διασποράς	16
2.2.6 Ρυθμός μεταφοράς δεδομένων	17

2.1 Δίκτυα LoRa

Τα δίκτυα LoRa (Long-Range) είναι μια τεχνολογία, που ανήκει στην οικογένεια των LPWAN δικτύων όπου χρησιμοποιείται σε μεγάλο αριθμό εφαρμογών σε περιβάλλοντα με κατανομή κόμβων σε μεγάλες αποστάσεις.

Πιο συγκεκριμένα το LoRa είναι μια τεχνολογία στο φυσικό επίπεδο (physical layer) η οποία χρησιμοποιείται για την μεταφορά πακέτων με χρήση ραδιοκυμάτων. Για την διαμόρφωση των σημάτων γίνεται χρήση της Chirp Spread Spectrum (CSS) τεχνολογίας, η οποία επιτρέπει να γίνεται χρήση όλου του εύρους ζώνης (bandwidth). Το LoRa κάνει χρήση συχνοτήτων οι οποίες βρίσκονται στις ζώνες υπό-συχνοτήτων χωρίς άδεια χρήσης. Στην Ευρώπη γίνεται χρήση του EU868 (863–870/873 MHz), στην Αμερική του US915 (902–928 MHz), στην Ασία του AS923 (915–928 MHz) και του 2.4GHz παγκοσμίως.

Το LoRa είναι μια τεχνολογία όπου σε σύγκριση με τις υπόλοιπες τεχνολογίες στην κατηγορία των LPWAN προσφέρει μεγάλο εύρος κάλυψης αλλά και ψηλές ταχύτητες μεταφοράς δεδομένων. Πιο συγκεκριμένα σε περιβάλλοντα πόλης παρέχει κάλυψη 5 χιλιομέτρων ενώ σε περιβάλλοντα ανοικτής κάλυψης χωρίς αντικείμενα που να μεταβάλλουν την μεταφορά ραδιοκυμάτων μπορεί να φτάσει η κάλυψη μέχρι και τα 20 χιλιόμετρα. Παράλληλα η μέγιστη ταχύτητα μεταφοράς δεδομένων είναι τα 50 kbit/s.

Το LoRa προσφέρει την δυνατότητα αποστολής μηνυμάτων από και προς τις δύο κατευθύνσεις. Συγκεκριμένα, υποστηρίζονται μηνύματα uplink τα οποία έχουν κατεύθυνση από την τελική συσκευή προς τον κεντρικό εξυπηρετητή και downlink με αντίθετη κατεύθυνση δηλαδή από τον κεντρικό εξυπηρετητή προς την τελική συσκευή. Τα μηνύματα uplink που αποστέλλονται από την τελική συσκευή έχουν την ιδιαιτερότητα ότι δεν έχουν συγκεκριμένη πύλη - παραλήπτη αλλά αντίθετα αποστέλλονται σε όλο το φάσμα και τα μηνύματα γίνονται αποδεκτά από όλες τις πύλες που παραλαμβάνουν το μήνυμα.

Στην συνέχεια όλες οι πύλες προωθούν το συγκεκριμένο μήνυμα στον κεντρικό εξυπηρετητή δικτύου ο οποίος είναι υπεύθυνος για την απάλειψη διπλοτύπων του μηνύματος. Αυτό μπορεί να χαρακτηριστεί ως ένα broadcast μήνυμα προς όλες τις διαθέσιμες πύλες. Αντίθετα σε ένα μήνυμα downlink μπορούμε να έχουμε μόνο ένα παραλήπτη (τελική συσκευή) ή το μήνυμα να προωθηθεί σε όλες τις τελικές συσκευές από την πύλη. Δηλαδή το μήνυμα από τον κεντρικό εξυπηρετητή προωθείται μόνο σε μία πύλη, για μία συγκεκριμένη συσκευή, την πύλη που μπορεί να ελαχιστοποιήσει την κατανάλωση των πόρων του δικτύου.

Επιπρόσθετα, τα δίκτυα LoRa προσφέρουν κρυπτογράφηση δεδομένων αλλά και υποστήριξη για ιδιωτικά δίκτυα (private networks) για την διασφάλιση της ασφάλειας των δεδομένων. Τόσο στην ασφάλεια, τόσο και στο κομμάτι της ευχρηστίας το LoRa είναι μια τεχνολογία που μπορεί να χρησιμοποιηθεί από οποιοδήποτε χρήστη και να τοποθετήσει τόσο τις τελικές συσκευές αλλά και τους δρομολογητές του σε οποιοδήποτε σημείο επιθυμεί. Στον Πίνακα 1 μπορούμε να δούμε ολοκληρωμένα τα πλεονεκτήματα των δικτύων LoRa (σε συνδυασμό με το LoRaWAN) έναντι σε άλλες υφιστάμενες τεχνολογίες LPWAN. Τα LoRa μπορεί να μην προσφέρουν σε όλες τις παραμέτρους την βέλτιστη λύση ωστόσο αποτελούν μια εξισορροπημένη επιλογή στο σύνολο της.

Πίνακας 1: Σύγκριση τριών διαφορετικών τεχνολογιών LPWAN

	LoRa – LoRaWAN	Sigfox	NB-IoT
Ζώνες Συχνότητων	Χωρίς άδεια χρήσης	Χωρίς άδεια χρήσης	Με άδεια χρήσης
Κάλυψη (μεικτό)	5 χιλιόμετρα	10 χιλιόμετρα	1 χιλιόμετρα
Κάλυψη (ανοικτό)	20 χιλιόμετρα	40 χιλιόμετρα	10 χιλιόμετρα
Μέγιστη Ταχύτητα	50 kbit/s	0.1 kbit/s	200 kbit/s
Μέγιστος αριθμός μηνυμάτων ανά μέρα	Απεριόριστα	140 Up , 4 Down	Απεριόριστα
Κρυπτογράφηση	NAI	OXI	NAI
Ιδιωτικά Δίκτυα	NAI	OXI	OXI
Τοποθεσία πυλών	Μπορεί οποιοδήποτε	Διαχειριστής δικτύου	Διαχειριστής δικτύου

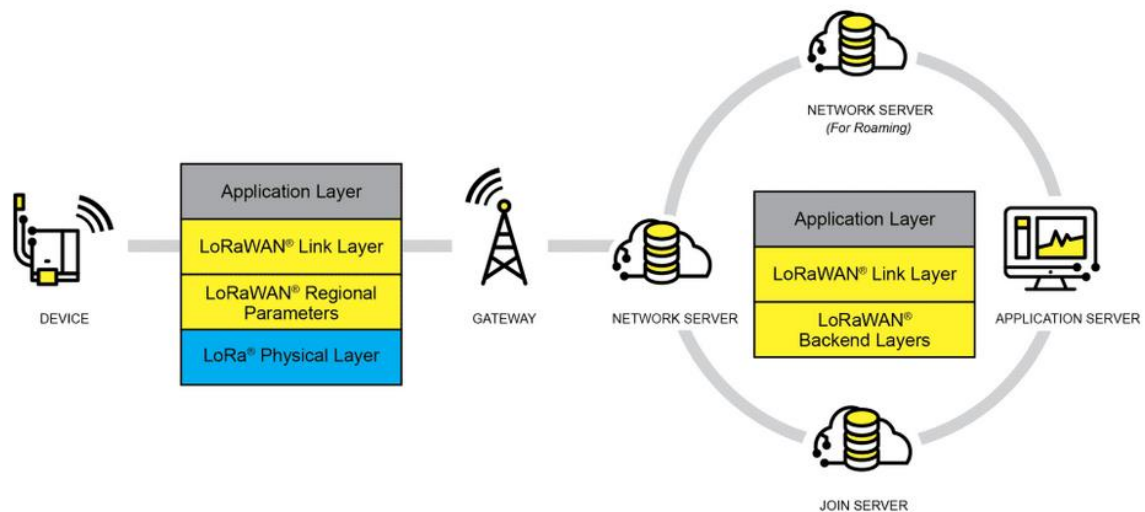
2.2 Πρωτόκολλο LoRaWAN

2.2.1 Γενικές Πληροφορίες:

Το πρωτόκολλο LoRaWAN είναι μία στοίβα από τεχνολογίες οι οποίες έχουν δημιουργηθεί πάνω στο φυσικό επίπεδο του LoRa για την εύκολη ανάπτυξη LPWAN δικτύων. Το LoRaWAN έχει αναπτυχθεί για να βοηθήσει στην ασύρματη επικοινωνία αντικειμένων στο διαδίκτυο τόσο σε τοπικά, εγχώρια αλλά και παγκόσμια δίκτυα. Στοχεύει στην ικανοποίηση των απαιτήσεων που έχει το διαδίκτυο των πραγμάτων όπως επικοινωνία διπλής κατεύθυνσης, ασφάλεια σε όλο το κομμάτι της επικοινωνίας και κινητικότητα των τελικών συσκευών.

Έτσι, το LoRaWAN προσεγγίζεται ως ένα πρωτόκολλο επικοινωνίας η και μια αρχιτεκτονική δικτύου, ενώ το φυσικό επίπεδο LoRa ενεργοποιεί τη σύνδεση μεγάλης εμβέλειας. Το πρωτόκολλο και η αρχιτεκτονική του δικτύου επηρεάζουν τον καθορισμό της διάρκειας ζωής της μπαταρίας ενός κόμβου / τελικής συσκευής, τη χωρητικότητα του δικτύου, την ποιότητα εξυπηρέτησης (QoS-Quality of Service), την ασφάλεια και αξιοπιστία των εφαρμογών στο δίκτυο. Στην Εικόνα 1 έχουμε την δομή με την οποία είναι κτισμένο ένα δίκτυο LoRaWAN τόσο για τις συσκευές που χρησιμοποιούνται αλλά και τις στοίβες πρωτοκόλλων. Σε γενικές γραμμές τα δίκτυα LoRaWAN αποτελούνται από τρία είδη συσκευών: τελικές συσκευές, πύλες αναμετάδοσης και τους κεντρικούς εξυπηρετητές. Όσον αφορά τα πρωτόκολλα όπως φαίνεται το φυσικό επίπεδο του LoRa εμφανίζεται μόνο ανάμεσα στην επικοινωνία της τελικής συσκευής με την πύλη όπου έχουμε μεταφορά ραδιοκυμάτων. Για την υπόλοιπη επικοινωνία

γίνεται η χρήση της στοίβας του LoRaWAN και της εφαρμογής που εξυπηρετεί η συγκεκριμένη τελική συσκευή.



Εικόνα 1: Αρχιτεκτονική δικτύου LoRaWAN

2.2.2 Τοπολογία:

Το LoRaWAN αναπτύσσεται σε τοπολογία star-of-stars στην οποία οι πύλες (gateways) αναμεταδίδουν μηνύματα μεταξύ των τελικών συσκευών και ενός κεντρικού διακομιστή δικτύου. Οι πύλες συνδέονται με τον διακομιστή δικτύου μέσω τυπικών συνδέσεων IP και λειτουργούν ως απλοί προωθητές, μετατρέποντας απλώς πακέτα ραδιοκυμάτων (RF) σε πακέτα IP και αντίστροφα. Για την αποστολή των πακέτων από την πύλη στον εξυπηρετητή γίνεται χρήση του πρωτοκόλλου UDP. Η ασύρματη επικοινωνία εκμεταλλεύεται τα χαρακτηριστικά μεγάλης εμβέλειας του φυσικού στρώματος LoRa, επιτρέποντας μια σύνδεση single-hop μεταξύ της τελικής συσκευής και μιας ή πολλών πυλών. Όλες οι λειτουργίες είναι ικανές για αμφίδρομη επικοινωνία και υπάρχει υποστήριξη για ομάδες διευθυνσιοδότησης πολλαπλής εκπομπής για την αποτελεσματική χρήση του φάσματος.

2.2.3 Κατηγορίες τελικών συσκευών:

Το LoRaWAN διαθέτει τρεις διαφορετικές κατηγορίες συσκευών τελικού σημείου για την αντιμετώπιση των διαφορετικών αναγκών που αντικατοπτρίζονται στο ευρύ φάσμα εφαρμογών:

Κατηγορία A – Όλες οι τελικές συσκευές LoRaWAN πρέπει να υποστηρίζουν τελικές συσκευές κατηγορίας A. Η επικοινωνία μεταξύ συσκευής και δικτύου κατηγορίας A ξεκινά πάντα από την τελική συσκευή. Μια συσκευή μπορεί να στείλει ένα μήνυμα uplink ανά πάσα

στιγμή. Μόλις ολοκληρωθεί η μετάδοση του μηνύματος, η συσκευή ανοίγει δύο σύντομα παράθυρα λήψης (downlink). Μεταξύ του τέλους της μετάδοσης του μηνύματος στο κεντρικό εξυπηρετητή και της έναρξης των παραθύρων λήψης υπάρχει μία μικρή καθυστέρηση. Εάν ο διακομιστής δικτύου δεν ανταποκριθεί κατά τη διάρκεια αυτών των δύο παραθύρων λήψης, η επόμενη κατερχόμενη ζεύξη θα είναι μετά την επόμενη μετάδοση ανερχόμενης ζεύξης. Δηλαδή, η τελική συσκευή δεν αναμένει την ανάκτηση μηνύματος επιβεβαίωσης για να προχωρήσει στην αποστολή του επόμενου μηνύματος.

Κατηγορία Β – Εκτός από τα παράθυρα λήψης που παρουσιάστηκαν από την κατηγορία Α, οι συσκευές κατηγορίας Β ανοίγουν παράθυρα προγραμματισμένης λήψης για τη λήψη μηνυμάτων downlink από τον κεντρικό διακομιστή δικτύου. Χρησιμοποιώντας χρονικά συγχρονισμένα πλαίσια που μεταδίδονται από την πύλη, οι συσκευές ανοίγουν περιοδικά παράθυρα λήψης. Επομένως, οι συσκευές κατηγορίας Β ανοίγουν επίσης παράθυρα λήψης μετά την αποστολή ενός uplink μηνύματος ωστόσο υποστηρίζουν επιπλέον την ανάληψη μηνυμάτων downlink σε ενδιάμεσα παράθυρα τα οποία ανοίγουν σε συγκεκριμένες χρονικές στιγμές μετά από συγχρονισμό.

Κατηγορία C – Η κατηγορία αυτή τελικών συσκευών επεκτείνει την διάρκεια των παραθύρων διατηρώντας συνεχώς ανοιχτό το παράθυρο για λήψη μηνυμάτων downlink. Τα παράθυρα αυτά απενεργοποιούνται μόνο όταν η συσκευή στέλνει μήνυμα uplink. Η συνεχής αυτή διαθεσιμότητα των παραθύρων απαιτεί μεγάλο όγκο ενέργειας, γι' αυτό το λόγο συσκευές αυτής της κλάσης έχουν συνεχή σύνδεση σε πηγή ενέργειας.

2.2.4 Κατηγορίες μηνυμάτων:

Εκτός από τις τρεις κατηγορίες τελικών συσκευών στο LoRaWAN συναντούμε και 8 είδη υποστηριζόμενων μηνυμάτων. Στην κεφαλή ενός πακέτου LoRa στο κομμάτι του στο κομμάτι του επιπέδου MAC (Media Access Control layer) συναντούμε 3 bits τα οποία χρησιμοποιούνται για να ορίσουν το είδος του μηνύματος που αποστέλλει μια τελική συσκευή ή ο κεντρικός εξυπηρετητής. Στον Πίνακα 2 βλέπουμε αναλυτικά στα αριστερά τον κωδικό σε δυαδική μορφή και στα δεξιά το είδος του μηνύματος. Οι δύο πρώτοι κωδικοί χρησιμοποιούνται από τις συσκευές όταν ένας νέος κόμβος εισέρχεται στο δίκτυο για να ενημερώσει για την συμμετοχή του σε αυτό. Οι επόμενοι τέσσερις κωδικοί αποτελούν τα τέσσερα πιθανά μηνύματα δεδομένων που μπορεί να ανταλλάξουν οι τελικές συσκευές με τον εξυπηρετητή δικτύου.

Πίνακας 2: Κωδικός και τύπος μηνύματος

MAC message type	Description
000	Join Request
001	Join Accept
010	Unconfirmed Data Up
011	Unconfirmed Data Down
100	Confirmed Data Up
101	Confirmed Data Down
110	RFU
111	Proprietary

Η έβδομη κατηγορία μηνυμάτων είναι μηνύματα RFU (Radio Frequency Unit) δηλαδή μηνύματα των πυλών. Όπως είδαμε στο προηγούμενο υπο-κεφάλαιο στις Κατηγορίες συσκευών B και C έχουμε και άλλα παράθυρα ανοικτά για downlink μηνύματα εκτός από αυτά μετά την αποστολή του μηνύματος όπου είναι για τα μηνύματα επιβεβαίωσης. Αυτά τα παράθυρα τις πλείστες φορές χρησιμοποιούνται από τις πύλες για να επιβεβαιωθούν ότι μία τελική συσκευή είναι ακόμη ενεργή στο δίκτυο. Αυτά τα μηνύματα κάνουν χρήση του κωδικού 7. Τέλος ο κωδικός 8 δεν είναι δεσμευμένος για κάποιο είδος μηνύματος από την στοίβα του LoRaWAN.

2.2.5 Συντελεστής Διασποράς – Spreading Factor (SF):

Όπως προαναφέρθηκε το LoRa βασίζεται στην τεχνολογία Chirp Spread Spectrum (CSS), όπου τα chirps (γνωστά και ως σύμβολα) είναι ο φορέας δεδομένων. Ο συντελεστής διασποράς (SF – Spreading Factor) ελέγχει το chirp rate και επομένως ελέγχει την ταχύτητα μετάδοσης δεδομένων. Χαμηλότεροι συντελεστές διασποράς σημαίνουν ταχύτερες μεταφορές chirps και συνεπώς υψηλότερο ρυθμό μετάδοσης δεδομένων. Για κάθε αύξηση του συντελεστή διασποράς, ο ρυθμός μεταφοράς chirps μειώνεται στο μισό και έτσι ο ρυθμός μετάδοσης δεδομένων μειώνεται στο μισό. Οι χαμηλότεροι συντελεστές διασποράς μειώνουν το εύρος των εκπομπών LoRa, επειδή μειώνουν το κέρδος επεξεργασίας και αυξάνουν τον ρυθμό μετάδοσης bit (bit rate).

Η αλλαγή του συντελεστή διασποράς επιτρέπει στο δίκτυο να αυξάνει ή να μειώνει τον ρυθμό μεταφοράς δεδομένων για κάθε τελική συσκευή με κόστος την εμβέλεια. Το δίκτυο

χρησιμοποιεί επίσης τον συντελεστή διασποράς για τον έλεγχο της συμφόρησης. Οι συντελεστές διασποράς είναι ορθογώνιοι, επομένως τα σήματα που διαμορφώνονται με διαφορετικούς συντελεστές διασποράς και μεταδίδονται στο ίδιο κανάλι συχνότητας την ίδια στιγμή δεν παρεμβαίνουν μεταξύ τους. Επιρροή των παραγόντων διάδοσης Η διαμόρφωση LoRa έχει συνολικά 6 παράγοντες διασποράς από SF7 σε SF12. Οι διάφορες τιμές που μπορεί να πάρει το SF επηρεάζουν τον ρυθμό μεταφοράς δεδομένων, το χρόνο μετάδοσης, τη διάρκεια ζωής της μπαταρίας και την ευαισθησία στις απώλειες (Sensitivity).

2.2.6 Ρυθμός μεταφοράς δεδομένων - Data Rate (DR)

Σε σύγκριση με έναν υψηλότερο συντελεστή διασποράς, ένας χαμηλότερος συντελεστής διασποράς παρέχει υψηλότερο ρυθμό μετάδοσης για σταθερό εύρος ζώνης. Για παράδειγμα, το SF7 παρέχει υψηλότερο ρυθμό μετάδοσης από το SF12. Ο διπλασιασμός του εύρους ζώνης διπλασιάζει επίσης τον ρυθμό μετάδοσης για σταθερό συντελεστή διασποράς. Ο παρακάτω πίνακας παρουσιάζει ρυθμούς μεταφοράς δεδομένων που υπολογίζονται με το SF7 για εύρη ζώνης, 125, 250 και 500 kHz.

Πίνακας 2: Ρυθμός μετάδοσης (DR) δεδομένων για σταθερό spreading factor για τρεις τιμές εύρους ζώνης (BW)

Spreading Factor	Bandwidth	Data rate (kbits/s)
7	125	5.5
7	250	10.9
7	500	21.9

Με βάση τα παραπάνω για να μεγιστοποιηθεί η διάρκεια ζωής της μπαταρίας των τελικών συσκευών και η συνολική χωρητικότητα του δικτύου, το πρωτόκολλο LoRaWAN διαχειρίζεται τη ρύθμιση DR και την ισχύ εξόδου RF για κάθε τελική συσκευή ξεχωριστά μέσω του αλγορίθμου Adaptive Data Rate (ADR).

Κεφάλαιο 3

Υφιστάμενοι Αλγόριθμοι

3.1 Υφιστάμενοι Αλγόριθμοι - ADR	18
3.2 Προβλήματα Υφιστάμενων Αλγορίθμων	20
3.3 Λύση LoRaWAN – Blind ADR	21

3.1 Υφιστάμενοι Αλγόριθμοι στα δίκτυα LoRaWAN

Γενικές Πληροφορίες:

Το πρωτόκολλο LoRaWAN υποστηρίζει την μεταβλητή ρύθμιση του ρυθμού μετάδοσης δεδομένων με την χρήση του σχήματος – αλγόριθμου Adaptive Data Rate (ADR). Το ADR μας επιτρέπει να εκμεταλλευτούμε τα πλεονεκτήματα του φυσικού στρώματος LoRa. Η προσαρμογή του ρυθμού δεδομένων σε ένα δίκτυο LoRaWAN μας επιτρέπει να κλιμακώνουμε εύκολα το δίκτυο, απλά προσθέτοντας πύλες. Επιπλέον, η χρήση ADR μπορεί να αυξήσει δραστικά τη χωρητικότητα ενός τέτοιου δικτύου.

Πιο συγκεκριμένα ο αλγόριθμος ADR για τον υπολογισμό του βέλτιστου ρυθμού μεταβολής δεδομένων γίνεται χρήση τριών μετρικών που ορίζουν την μεταφορά ενός πακέτου. Οι μετρικές αυτές είναι ο ρυθμός διάδοσης (SF – Spreading Factor), η ευαισθησία στις απώλειες (Sensitivity - RSSI) και ο χρόνος μετάδοσης του πακέτου (Time on Air).

Χρόνος μετάδοσης Δεδομένων - Time-On-Air:

Σε σύγκριση με έναν χαμηλότερο συντελεστή διασποράς, η αποστολή σταθερού όγκου δεδομένων με υψηλότερο συντελεστή διάδοσης και σταθερό εύρος ζώνης απαιτεί μεγαλύτερο χρόνο σε λειτουργία. Ο Χρόνος μετάδοσης Δεδομένων μπορεί να υπολογιστεί με βάση τρεις παράγοντες το μέγεθος του πακέτου που θα αποσταλεί σε bytes, το εύρος ζώνης και τον συντελεστή διασποράς.

Ευαισθησία δέκτη στις απώλειες – Sensitivity - RSSI:

Οι υψηλότεροι συντελεστές διασποράς παρέχουν υψηλότερη ευαισθησία του δέκτη. Συνήθως, το LoRa χρησιμοποιεί υψηλότερους συντελεστές διασποράς όταν το σήμα είναι αδύναμο. Ο πιο κάτω πίνακας δείχνει πως οι τιμές αυτές προσαρμόζονται σε ένα σενάριο όπου το εύρος ζώνης (Bandwidth) που χρησιμοποιείται είναι τα 125 kHz.

Πίνακας 3 : Ρυθμός διάδοσης (Spreading Factor), Ευαισθησία απώλειας (Sensitivity) και Χρόνος μετάδοσης (Time on Air) για σενάριο με εύρος ζώνης στα 125 kHz .

Ρυθμός διάδοσης (Spreading Factor)	Ευαισθησία απώλειας (Sensitivity - RSSI)	Χρόνος μετάδοσης (Time on Air)
SF7	-123.0 dBm	41 ms
SF8	-126.0 dBm	72 ms
SF9	-129.0 dBm	144 ms
SF10	-132.0 dBm	288 ms
SF11	-134.5 dBm	577 ms
SF12	-137.0 dBm	991 ms

Δεδομένου του χρόνου μετάδοσης, είναι σαφές ότι οι τελικές συσκευές (κόμβοι) πιο κοντά στην πύλη δεν χρειάζονται το SF12 αφού δεν χρειάζεται να μείνουν στον αέρα τόσο χρόνο. Σε αυτήν την περίπτωση, η χρήση ADR μπορεί να βελτιστοποιήσει το Spreading Factor που χρησιμοποιείται για κάθε μετρητή και να ελαχιστοποιήσει τον επόμενο χρόνο μετάδοσης στον αέρα. Ο ADR είναι ένας πολύ απλός μηχανισμός που αλλάζει τον ρυθμό δεδομένων με βάση απλούς κανόνες και προσαρμόζει τον ρυθμό δεδομένων του κόμβου στον διαθέσιμο αριθμό πόρων.

Αλγόριθμος ADR:

Για την ανάπτυξη και τον εμπλουτισμό του LoRaWAN πρωτοκόλλου έχει δημιουργηθεί ο οργανισμός The Things Network (TTN) οποίος έχει δημιουργήσει μια δική του στοίβα η οποία είναι βασισμένη πάνω στα θεμέλιά του LoRaWAN αναπτύσσοντας τις υπάρχουσες τεχνολογίες αλλά δημιουργώντας και νέα συστατικά. Η στοίβα του TTN είναι η πιο διαδεδομένη υλοποίηση του LoRaWAN που χρησιμοποιείται παγκοσμίως. Πιο κάτω θα αναπτυχθεί ο τρόπος με τον οποίο γίνεται χρήση του αλγόριθμου ADR TTN stack.

Αρχικά η χρήση του αλγόριθμου ADR αποφασίζεται από την τελική συσκευή (κόμβος). Η απόφαση αυτή γίνεται με βάση την κατάσταση στην οποία βρίσκεται η τελική συσκευή και πιο συγκεκριμένα η τελική συσκευή πρέπει να βρίσκεται σε κατάσταση σταθερή όσο αφορά την εκπομπή ραδιοκυμάτων (stable Radio Frequency conditions). Εφόσον μια συσκευή βρίσκεται σε μια τέτοια κατάσταση τότε στο επόμενο μήνυμα που θα αποσταλεί από την τελική συσκευή, μέσα στην κεφαλή του πακέτου υπάρχει το σχετικό bit ενεργοποίησης του αλγόριθμου (ADR bit) .

Με βάση λοιπόν την υλοποίηση του ADR από το TTN stack, για να προσδιοριστεί ο βέλτιστος ρυθμός μεταφοράς δεδομένων (data rate), το δίκτυο χρειάζεται κάποιες μετρήσεις. Οι μετρήσεις αυτές είναι μηνύματα uplink δηλαδή μηνύματα τα οποία έχουν αποσταλεί από την τελική συσκευή προς τον κεντρικό εξυπηρετητή. Επί του παρόντος, το The Things Stack κάνει χρήση των 20 πιο πρόσφατων μηνυμάτων, ξεκινώντας από τη στιγμή που ορίζεται το ADR bit. Αυτές οι μετρήσεις περιέχουν τον αριθμό πλαισίου (frame counter), τον λόγο σήματος προς θόρυβο (SNR – Signal to Noise Ratio) και τον αριθμό των πυλών (gateways) που έλαβαν κάθε πακέτο. Όταν μια συσκευή απενεργοποιεί το bit ADR (επειδή γνωρίζει ότι κινείται ή γνωρίζει ότι οι συνθήκες RF είναι ασταθείς), οι προηγούμενες μετρήσεις απορρίπτονται. Μόλις ρυθμιστεί ξανά το bit ADR, το δίκτυο ξεκινά ξανά τη μέτρηση.

Για καθεμία από αυτές τις μετρήσεις, παίρνουμε το SNR της καλύτερης πύλης και υπολογίζουμε το «περιθώριο» ή διαφορά, το οποίο είναι το μετρούμενο SNR μείον το απαιτούμενο SNR για την αποδιαμόρφωση (demodulation) ενός μηνύματος δεδομένου του ρυθμού μεταφοράς δεδομένων (data rate). Αυτό το περιθώριο χρησιμοποιείται για να προσδιορίσει πόσο μπορούμε να αυξήσουμε τον ρυθμό μεταφοράς δεδομένων ή να μειώσουμε την ισχύ μετάδοσης. Για παράδειγμα, όταν το δίκτυο λαμβάνει ένα μήνυμα με ρυθμό μετάδοσης δεδομένων SF12BW125 και SNR 5.0, αυτό το μήνυμα έχει περιθώριο 25 dB. Αυτό είναι σπατάλη πολύτιμου χρόνου ομιλίας και ενέργειας. Αν αυξάναμε τον ρυθμό μετάδοσης δεδομένων σε SF7BW125, θα εξακολουθούσαμε να έχουμε ένα περιθώριο 12,5 dB, αλλά αυτό θα ήταν πολλές φορές πιο αποδοτικό στον χρόνο αποστολής και στην ενέργεια. Θα μπορούσαμε ακόμη να μειώσουμε την ισχύ εκπομπής για να εξοικονομήσουμε ακόμη περισσότερη ενέργεια και να προκαλέσουμε λιγότερες παρεμβολές.

3.2 Προβλήματα Υφιστάμενων Αλγόριθμων

Ο αλγόριθμος ADR είναι μια εξαιρετική λύση για την βελτιστοποίηση των παραμέτρων που περιεγράφηκαν παραπάνω ωστόσο έχουν εντοπιστεί μερικά μειονεκτήματα που θα

προσπαθήσουμε να καλύψουμε με τον αλγόριθμό μας. Το πρώτο μειονέκτημα του ADR είναι η δυνατότητα ενεργοποίησης του μόνο σε σταθερές συνθήκες Ραδιοσυχνότητας (RF conditions). Στο LPWAN οι σταθερές συνθήκες RF δεν είναι μια κατάσταση η οποία είναι εφικτή σε πολλές περιπτώσεις λόγω της απόστασης ανάμεσα στις συσκευές και των εμποδίων που μπορούν να βρεθούν. Οι ακραίες καιρικές αλλαγές και η μετακίνηση μιας τελικής συσκευής είναι δύο από τους πιο συνηθισμένους λόγους που οι συνθήκες RF δεν μπορούν να είναι σταθερές και ως εκ τούτου το ADR δεν μπορεί να χρησιμοποιηθεί για τη βελτιστοποίηση των ρυθμού μετάδοσης δεδομένων. Οι περισσότερες εφαρμογές όπως εντοπισμός τοποθεσίας οχημάτων, κινητικότητα των ζώων σε φάρμες κλπ. Έχουν ως βασικό παράγοντα την κίνηση της τελικής συσκευής. Άλλες συχνές εφαρμογές είναι η μέτρηση διαφόρων παραμέτρων του καιρού σε απομακρυσμένες περιοχές ή δάση όπως θερμοκρασία, υγρασία κ.ά. όπου σε περιπτώσεις όπου οι καιρικές συνθήκες είναι ακραίες μπορεί να μην είναι εφικτή η ενεργοποίηση του ADR.

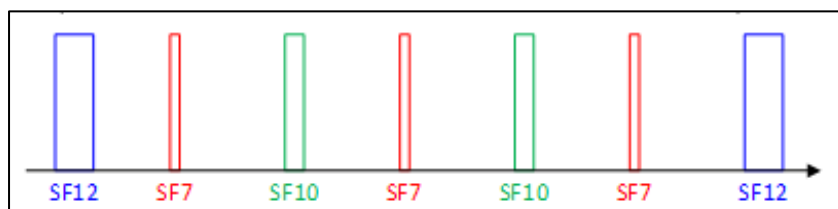
Ένα δεύτερο μειονέκτημα του αλγόριθμου ADR είναι σε περιπτώσεις όπου έχουμε σταθερές συνθήκες RF ο αλγόριθμος καθυστερεί να τεθεί σε λειτουργία. Συγκεκριμένα, από τη στιγμή που μια τελική συσκευή προσδιορίζει ότι οι συνθήκες RF είναι σταθερές και ρυθμίζει το ADR bit σε ένα, ο κεντρικός εξυπηρετητής δικτύου χρειάζεται τα 20 πιο πρόσφατα μηνύματα uplink για να καθορίσει την βέλτιστη τιμή. Έτσι, σε περίπτωση προσθήκης μιας νέας συσκευής στο δίκτυο, πρέπει να στείλουμε τουλάχιστον 20 πακέτα πρώτα πριν το ADR θα μπορεί να ξεκινήσει τη διαδικασία βελτιστοποίησης. Στα δίκτυα LPWAN η αποστολή 20 πακέτων μπορεί να πάρει αρκετό χρόνο εφόσον δεν υπάρχει συνεχόμενη αποστολή αλλά στις περισσότερες εφαρμογές όποτε υπάρξει αλλαγή στην μέτρηση ή ανά σταθερά χρονικά διαστήματα τις τάξεως δεκάδων λεπτών ή και ωρών. Επομένως, η καθυστέρηση αυτή συνάρτηση με την ιδιαιτερότητα ότι σε περίπτωση που απενεργοποιηθεί και ενεργοποιηθεί ξανά ο αλγόριθμος χρειαζόμαστε νέα 20 πακέτα uplink, κάνει την διαδικασία ενεργοποίησης και λειτουργίας του ADR πολύ χρονοβόρα.

Γι' αυτό σε περιπτώσεις όπου η ενεργοποίηση του ADR είναι αδύνατη, η λύση θα ήταν η χρήση μιας σταθερής τιμής για τον ρυθμό μεταφοράς δεδομένων. Η χρήση σταθερής τιμής ωστόσο δεν είναι μια οικονομική λύση αφού πάντα υπάρχει προσπάθεια για εξισορρόπηση του ρυθμού μεταφοράς δεδομένων, του εύρους κάλυψης και της ενεργειακής κατανάλωσης.

3.3 Λύση LoRaWAN – Blind ADR

Οι λόγοι που προαναφέρθηκαν, καθιστούν τον αλγόριθμο ADR σε θέση όπου δεν αποτελεί μια πλήρης ικανοποιητική λύση. Τόσο εμείς όσο και το ίδιο το πρωτόκολλο αναγνωρίζουν αυτές τις αδυναμίες και προχώρησαν στην υλοποίηση του blind ADR. Ο blind ADR είναι ένας αλγόριθμος υλοποιημένος από το LoRaWAN με στόχο την επίλυση του προβλήματος της κινητικότητας με πολύ απλή μεθοδολογία. Η λογική του αλγορίθμου βασίζεται στο ότι μια συσκευή η οποία κινείται στον χώρο μέσα σε ένα χρονικό διάστημα, μπορεί να βρεθεί σε όλες τις πιθανές καταστάσεις.

Συγκεκριμένα η ιδέα πίσω από τον blind ADR είναι η χρήση περισσότερων από ένα SF μέσα σε ένα δεδομένο χρονικό διάστημα. Για παράδειγμα όπως βλέπουμε στην Εικόνα 2 γίνεται χρήση τριών διαφορετικών τιμών spreading factor. Οι τιμές αυτές στην περίπτωση αυτή είναι οι SF7, SF10 και SF12. Οι τιμές αυτές δεν είναι σταθερές αλλά επιλέγονται με βάση την εφαρμογή στην οποία θα υλοποιηθεί ο αλγόριθμος από τον διαχειριστή. Στην περίπτωση αυτή βλέπουμε ότι μέσα σε διάστημα 60 λεπτών, με βάση το ότι η τελική συσκευή αποστέλλει μήνυμα κάθε δέκα λεπτά, προς την πύλη, κάνει χρήση του SF7 τρεις φορές, του SF10 δύο και μία φορά του SF12. Άρα, έχοντας υπόψη το χρονοδιάγραμμα και την κίνηση που εκτελεί μια τελική συσκευή μπορεί να δημιουργηθεί ένας αντίστοιχο χρονοδιάγραμμα για την χρήση το spreading factor όπως στην περίπτωση που φαίνεται πιο κάτω.



Εικόνα 2: Κατανομή τιμών SF σε ένα διάστημα 60 λεπτών στον blind ADR

Με βάση την κατανομή που δίνεται και θέτοντας κάποιες τιμές όσον αφορά την ποσότητα ενέργειας που χρειάζεται η συσκευή μας για να λειτουργεί για ένα χρόνο έχουμε τα αποτελέσματα του Πίνακα 3. Όπως φαίνεται ο blind ADR είναι ένας πιο οικονομικός αλγόριθμος όσον αφορά την κατανάλωση ενέργειας με την συγκεκριμένη ανάθεση των τιμών των spreading factors. Βλέπουμε από την τελευταία γραμμή του Πίνακα ότι ο αλγόριθμος blind ADR μπορεί να έχει καλύτερη κατανάλωση ενέργειας από κάθε σταθερή ανάθεση τιμής spreading factor μεγαλύτερη ή ίση του 10.

Πίνακας 3: Κατανάλωση ενέργειας ανά SF και με την χρήση blind ADR

Data Rate	SF12	SF10	SF7	All three
Battery Capacity per packet	1100 mAh	390 mAh	135 mAh	381 mAh
Total consumption for 6 packets	6600 mAh	2340 mAh	810 mAh	2286 mAh

Ωστόσο, ο τρόπος λειτουργίας του blind ADR δεν είναι τόσο αποδοτικός αφού και πάλι δεν μπορεί να εγγυηθεί ότι στα συγκεκριμένα χρονικά διαστήματα μια τελική συσκευή θα βρίσκεται στις συγκεκριμένες θέσεις – αποστάσεις. Επίσης, είναι ένας αλγόριθμος ο οποίος πρέπει να προσαρμοστεί σε συγκεκριμένη εφαρμογή για την βέλτιστη κατανομή των Spreading factors μέσα στο χρονικό πλαίσιο. Σε εφαρμογές όπου η διαδρομή ή κίνηση που ακολουθεί μια συσκευή δεν είναι σταθερή μπορούμε να έχουμε μεγάλο αριθμό χαμένων πακέτων.

Κεφάλαιο 4

Μεθοδολογία

4.1 Μηχανική Μάθηση σε τελικές συσκευές	24
4.2 Μεθοδολογία ανάπτυξης tinyML μοντέλων	24
4.3 Υλοποίηση πρότυπου μοντέλου tinyML	26

4.1 Μηχανική Μάθηση σε τελικές συσκευές

Ο τομέας της τεχνίτης νοημοσύνης και συγκεκριμένα της μηχανικής μάθησης την τελευταία δεκαετία είναι από τα πιο πολυσυζητημένες και ραγδαία αναπτυσσόμενες τεχνολογίες του τομέα. Η δυνατότητα αυτοματοποίησης και λήψεων αποφάσεων από τις ίδιες τις συσκευές με μεγάλη ακρίβεια είναι κάτι που έχει βοηθήσει αρκετά στην ελαχιστοποίηση του χρόνου λήψεως αποφάσεων αλλά και λειτουργίας βασικών υπηρεσιών. Τα δύο κυρίως προβλήματα τα οποία δεν έκαναν εφικτή την ανάπτυξη και εύκολη χρήση της μηχανικής μάθησης ήταν και είναι ακόμη ως ένα βαθμό, η ανάγκη για τεράστιο όγκο δεδομένων για την δημιουργία ενός καλού μοντέλου αλλά και η ανάγκη για δυνατές μηχανές επεξεργασίας του τεράστιου αυτού όγκου δεδομένων. Με την ανάπτυξη του διαδικτύου και την ύπαρξη του μεγάλου όγκου αυτού δεδομένων διαθέσιμο σε αυτό ανοικτά, παράλληλα με την εκτόξευση της υπολογιστικής ισχύς σήμερα στις νέες συσκευές, η τεχνητή νοημοσύνη έχει γίνει ένα φλέγον θέμα.

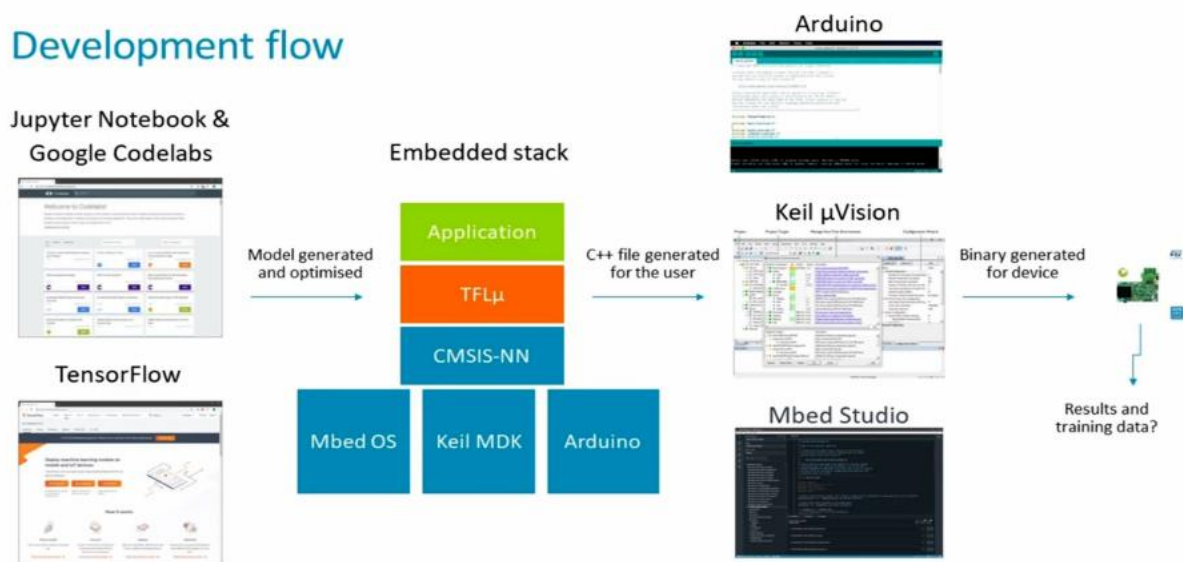
Η ανάγκη για χρήση της μηχανικής μάθησης έχει προχωρήσει και στον τομέα του διαδικτύου των αντικειμένων, όμως όπως προαναφέρθηκε η χρήση μοντέλων και αλγορίθμων προϋποθέτει την ύπαρξη ισχυρών επεξεργαστών και αρκετής μνήμης για επεξεργασία των δεδομένων εισόδου. Η επίλυση του ζητήματος αυτού είναι η χρήση tinyML μια τεχνολογία η οποία έχει εμφανιστεί τα τελευταία χρόνια και έχει ως απώτερο σκοπό την υλοποίηση αλγορίθμων μηχανικής μάθησης στις τελικές συσκευές.

4.2 Μεθοδολογία ανάπτυξης tinyML μοντέλων

Όπως είναι αναμενόμενο η δημιουργία ενός μοντέλου το οποίο θα μπορεί να τρέχει σε μία συσκευή η οποία τροφοδοτείται από μια μπαταρία δεν μπορεί να είναι η ίδια με τους

συμβατούς υπολογιστές. Για την δημιουργία τέτοιων μοντέλων μεγάλοι κολοσσοί του χώρου των ενσωματωμένων συστημάτων (embedded systems) και όχι μόνο, έχουν συνεργαστεί για να δημιουργήσουν τα απαραίτητα εργαλεία, ανάμεσα τους Google, Arduino, Qualcomm , ARM κ.α.

Για την δημιουργία ενός μοντέλου tinyML όπως φαίνεται και στην Εικόνα 3 πρέπει να ακολουθηθεί μια συγκεκριμένη διαδικασία με συγκεκριμένα εργαλεία. Αρχικά, η δημιουργία του μοντέλου προϋποθέτει την χρήση της γλώσσας προγραμματισμού Python και των βιβλιοθηκών TensorFlow Lite. Το TensorFlow Lite όπως και το κανονικό TensorFlow είναι βιβλιοθήκες που υποστηρίζουν την εύκολη και γρήγορη δημιουργία μοντέλων μηχανικής μάθησης με την χρήση γνωστών αλγορίθμων και νευρωνικών δικτύων. Η διαφορά των μοντέλων TensorFlow Lite βρίσκεται στο τελικό μέγεθος του μοντέλου, τον τρόπο αποθήκευσης των αποτελεσμάτων και κατά συνέπεια στον χρόνο εκτέλεσης ενός τέτοιου μοντέλου. Ένα μοντέλο TensorFlow Lite στην τελική του μορφή έχει μέγεθος της τάξεως των Kilobytes και τα δεδομένα αυτά είναι αποθηκευμένα με την απλούστερη και πιο συμπίεσμένη μορφή. Για να γίνει εφικτό αυτό γίνεται χρήση μεθόδων όπως το quantization το οποίο μπορεί να μειώσει ραγδαία το μέγεθος του μοντέλου έχοντας κάτω από 1% απώλειες από τα αρχικά αποτελέσματα.



Εικόνα 3: Διάγραμμα ροής ανάπτυξης μοντέλου tinyML

Για να αναπτυχθεί ένα TensorFlow Lite μοντέλο υπάρχουν δύο τρόποι ανάπτυξης. Πρώτος τρόπος είναι η ανάπτυξη ενός μοντέλου TensorFlow Lite από την αρχή χρησιμοποιώντας τις βιβλιοθήκες αυτής, όπως υπάρχει και η δυνατότητα συγγραφής μοντέλου με την κανονική βιβλιοθήκη TensorFlow και μετατροπή του έπειτα μέσω συναρτήσεων σε TensorFlow Lite.

Μετά την δημιουργία του μοντέλου ακολουθεί η διαδικασία μετατροπής του μοντέλου σε μια βιβλιοθήκη γλώσσας C++. Οι λόγοι που γίνεται η μετατροπή αυτή είναι αρκετοί, με κύριο λόγο όμως την υποστήριξη μεταγλωττιστή για C++ κώδικα από σχεδόν όλες τις κατασκευάστριες εταιρίες τελικών συσκευών. Η μετατροπή αυτή δίνει την δυνατότητα επεξεργασίας της βιβλιοθήκης αυτής στα περιβάλλοντα κάθε συσκευής ξεχωριστά κάνοντας χρήση των βιβλιοθηκών που διαχειρίζονται αυτά τα αρχεία.

Αφού δημιουργηθεί ο κώδικας στην τελική συσκευή τότε το τελικό μοντέλο καταλήγει να είναι ένα binary file το οποίο μπορεί να επεξεργαστεί μια τελική συσκευή. Η δυνατότητα επεξεργασίας τέτοιων αρχείων ωστόσο στην παρούσα φάση δεν μπορεί να γίνει από όλες τις τελικές συσκευές. Οι συσκευές που μπορούν να εκτελέσουν μοντέλα TensorFlow Lite είναι εφοδιασμένες με συγκεκριμένους επεξεργαστές και με την υποστήριξη των βιβλιοθηκών TensorFlow Lite.

4.3 Υλοποίηση πρότυπου μοντέλου tinyML

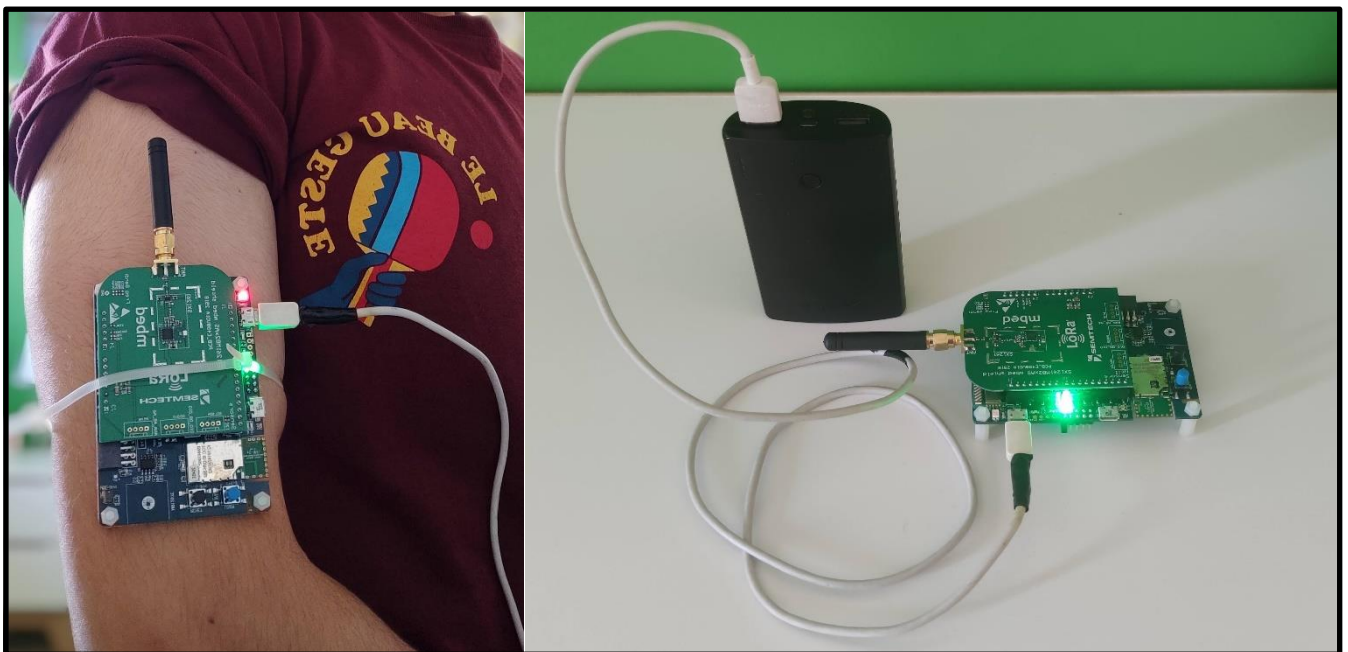
Η τεχνολογία tinyML είναι μία τεχνολογία όπου βρίσκεται σε μια πολύ πρώιμη φάση κάτι που μας καθιστούσε σε θέση όπου η λειτουργία ενός τέτοιου μοντέλου μπορούσε να αμφισβητηθεί. Έτσι αποφασίσαμε να προχωρήσουμε με την απόδειξη λειτουργίας της τεχνολογίας αυτή με την δημιουργία ενός αντίστοιχου μοντέλου μηχανικής μάθησης.

Μετά από έρευνα καταλήξαμε στην δημιουργία ενός μοντέλου μηχανικής μάθησης κατηγοριοποίησης κινήσεων με βάση την αντίστοιχη δουλειά που παρουσιάστηκε από τον Jan Jongboom (CTO at Edge Impulse) [5] στο ετήσιο συνέδριο του The Things Network. Το μοντέλο που παρουσιάσαμε ήταν ένα μοντέλο δημιουργημένο με την χρήση της πλατφόρμας Edge Impulse η οποία αποτελεί ένα μέσο ανάπτυξης τέτοιων μοντέλων χωρίς την τριβή με κώδικα που θα αναπαράγει το μοντέλο αυτό. Η αποστολή του αποτελέσματος που παράγει το μοντέλο γίνεται μέσω αποστολής πακέτου LoRa από την τελική συσκευή σε μια τοπική πύλη. Δηλαδή, το μοντέλο θα παράγει αποτελέσματα για την κατηγορία της κίνησης της οποίας εκτελεί ο χρήστης συνεχώς και αν υπάρξει αλλαγή στην κίνηση του χρήστη τότε ένα μήνυμα LoRa παράγεται και αποστέλνεται από την συσκευή για να ενημερώσει για την αλλαγή στην κίνηση του χρήστη.

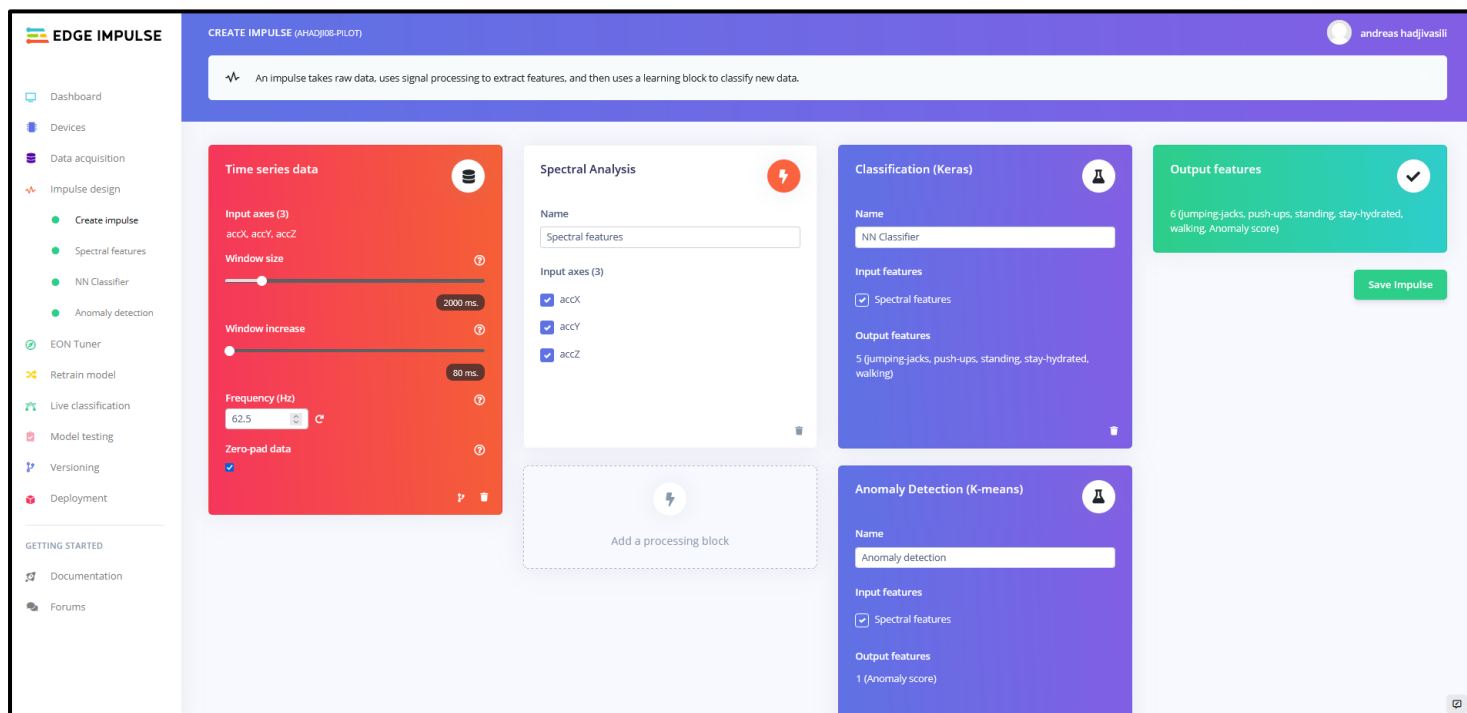
Η διαδικασία ανάπτυξης ενός τέτοιου μοντέλου μέσω της πλατφόρμας αποτελεί μια πολύ απλή και γρήγορη διαδικασία. Το σημείο το οποίο απαιτεί χρόνο στην διαδικασία αυτή είναι η άντληση των δεδομένων εκπαίδευσης μέσα από την τελική συσκευή. Η άντληση των

δεδομένων γίνεται με εύκολο τρόπο ωστόσο απαιτητέ η παραγωγή μεγάλου αριθμού αρχείων κίνησης για κάθε κατηγορία έτσι ώστε να εκπαιδευτεί σωστά το μοντέλο. Η τελική συσκευή που χρησιμοποιήθηκε είναι η προτεινόμενη ST IoT Discover Kit συσκευή με ενσωματωμένο αισθητήρα επιτάχυνσης (accelerometer) για την ανάκτηση των δεδομένων. Για την αποστολή των μηνυμάτων LoRa έγινε χρήση του αναμεταδότη της Semtech SX126xMB2xAS. Στις Εικόνες 4 και 5 βλέπουμε τις συσκευές αυτές σε λειτουργία. Στην Εικόνα 5 βλέπουμε ότι η τελική συσκευή μας τροφοδοτείται από μία μπαταρία., όπως επίσης μπορούμε να δούμε την anténna με την οποία γίνεται η αποστολή των μηνυμάτων LoRa.

Στην Εικόνα 4 φαίνεται η τοποθέτηση της συσκευής η οποία τροφοδοτείται από την μπαταρία στον χέρι του χρήστη για την άντληση των δεδομένων. Ο χρήστης επομένως καλείται να εκτελέσει κάποιες ενέργειες όπως περπάτημα, να πιει νερό, push ups κλπ. οι οποίες με την χρήση του αισθητήρα επιτάχυνσης για κάθε μια από αυτές δημιουργείται ένα σύνολο συντεταγμένων x,y,z οι οποίες περιγράφουν την κίνηση. Οι μετρήσεις αυτές λαμβάνονται απευθείας από την πλατφόρμα. Επομένως, για την άντληση των δεδομένων αυτών ο χρήστης πρέπει να εκτελέσει την κάθε κίνηση πολλαπλές φορές για ένα συγκεκριμένο χρονικό διάστημα. Στην δική μας περίπτωση έγινε άντληση δεδομένων συνολικής διάρκειας 5 λεπτών από πολλαπλές μετρήσεις για κάθε κίνηση διάρκειας 10 και 15 δευτερολέπτων κάθε μια, καταλήγοντας με 25 περίπου αρχεία για κάθε κίνηση.



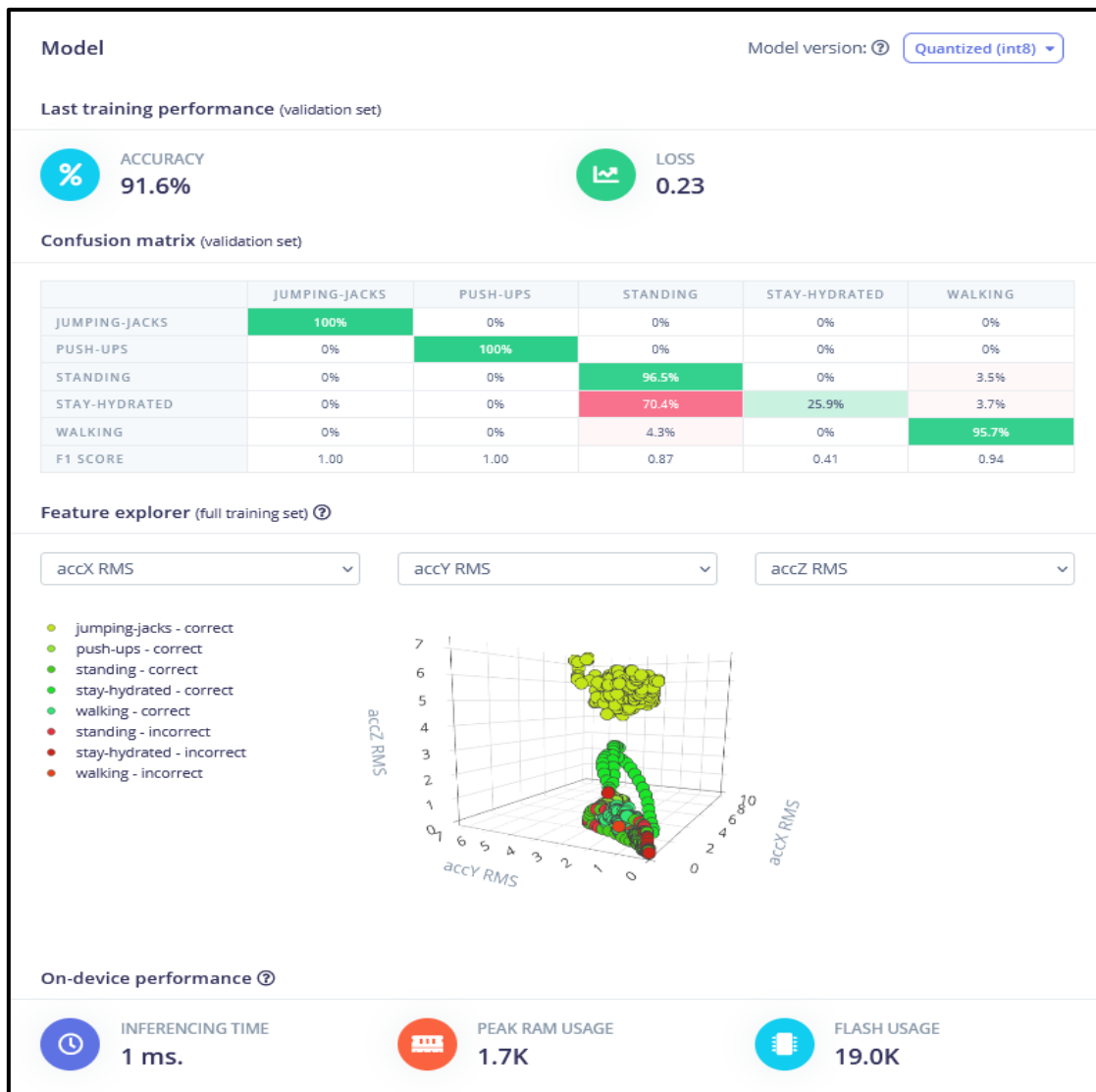
Εικόνες 4 - 5: Οι συσκευές που χρησιμοποιήθηκαν για την υλοποίηση του πρότυπου και η τοποθέτηση του στον ώμο για τη άντληση των δεδομένων



Εικόνα 6: Διαδικασία ανάπτυξης μοντέλου στην πλατφόρμα Edge Impulse

Όπως φαίνεται στην Εικόνα 6 η δημιουργία του μοντέλου είναι μια τυπική διαδικασία η οποία ακολουθείται και ολοκληρώνεται σε πολύ μικρό χρονικό διάστημα ακολουθώντας τα βήματα της πλατφόρμας. Σε γενικές γραμμές αρχικά, έχουμε την σύνδεση της συσκευής στην πλατφόρμα και στην συνέχεια την άντληση των δεδομένων μέσω της συσκευής από την πλατφόρμα. Αφού τελειώσει η άντληση των δεδομένων έπειτα η διαδικασία μεταφέρεται πλήρως στην πλατφόρμα όπου γίνεται η κατηγοριοποίηση των δεδομένων, η εκπαίδευση του μοντέλου και βελτιστοποίηση του. Τέλος έχουμε την εξαγωγή του μοντέλου πλήρως βελτιστοποιημένο σε μορφή δυαδικού αρχείου (binary file) το οποίο απλά το αντιγράφουμε στην τελική συσκευή και έχουμε έτοιμο το μοντέλο στην συσκευή.

Στην Εικόνα 7 φαίνεται η αναλυτική περιγραφή της διαδικασίας εκπαίδευσης του μοντέλου με πληροφορίες όπως το ποσοστό επιτυχίας που κατάφερε να αναπτύξει ο αλγόριθμος αλλά και μια γραφική αναπαράσταση των αποτελεσμάτων σε ένα τρισδιάστατο χώρο. Επίσης, στο κάτω μέρος της Εικόνας 5 μπορούμε να δούμε μετρικές απόδοσης του μοντέλου στην ίδια την συσκευή. Όπως βλέπουμε η εκτέλεση του μοντέλου μπορεί να γίνει σε χρόνο 1 milisecond και κάνει χρήση μόλις 19 kilobytes μνήμης στο χώρο αποθήκευσης της συσκευής και 1.7 kilobytes μνήμης RAM. Η χαμηλή αυτή χρήση εσωτερικού χώρου οφείλεται στην αποθήκευση του μοντέλου με την μορφή quantized όπου όλοι οι 32 ή 64 bit αριθμοί μετατρέπονται σε 8 bit.



Εικόνα 7: Αποτελέσματα εκπαίδευσης μοντέλου στην πλατφόρμαEdge Impulse

Αφού ακολουθήσουμε την διαδικασία εκπαίδευσης και ελέγχου του μοντέλου μας υπάρχει η επιλογή εξαγωγής του μοντέλου σε μορφή C++ βιβλιοθήκης ανάλογα με την τελική συσκευή την οποία διαθέτουμε.. Αφού ολοκληρωθεί η διαδικασία παραγωγής της βιβλιοθήκης αυτό που απομένει είναι η προσθήκη του κώδικα ο οποίος εκτελεί το μοντέλο και κάνει αποστολή του μηνύματος όταν έχουμε αλλαγή στην κίνηση του χρήστη.

Επομένως, αναπαράγοντας το πιο πάνω μοντέλο ήμασταν σε θέση να επιβεβαιώσουμε ότι η πρόταση μας μπορεί να λειτουργήσει. Ο λόγος που μπορούμε να βεβαιωθούμε για την λειτουργία του δικού μας μοντέλου σε μια τελική συσκευή είναι η απλούστερη λειτουργία που θα παρέχει. Πιο συγκεκριμένα, το μοντέλο κατηγοριοποίησης κίνησης είναι μια λειτουργία η οποία απαιτεί την συνεχόμενη άντληση δεδομένων από τον αισθητήρα επιτάχυνσης και επεξεργασία τους κάτι που επιβαρύνει πολύ περισσότερο την υπολογιστή ισχύ και κατά συνέπεια την κατανάλωση ενέργειας. Αντίθετα το δικό μας μοντέλο εκτελείται μόνο όταν υπάρχει αποστολή μηνύματος από την τελική συσκευή.

Κεφάλαιο 5

Υλοποίηση Αλγορίθμου

5.1 Γενικές Πληροφορίες Υλοποίησης	30
5.2 Αρχικό σύνολο Δεδομένων	32
5.3 Δημιουργία Μοντέλου	37
5.4 Επέκταση δεδομένων	39
5.5 Αποτελέσματα με τα νέα Δεδομένα	41

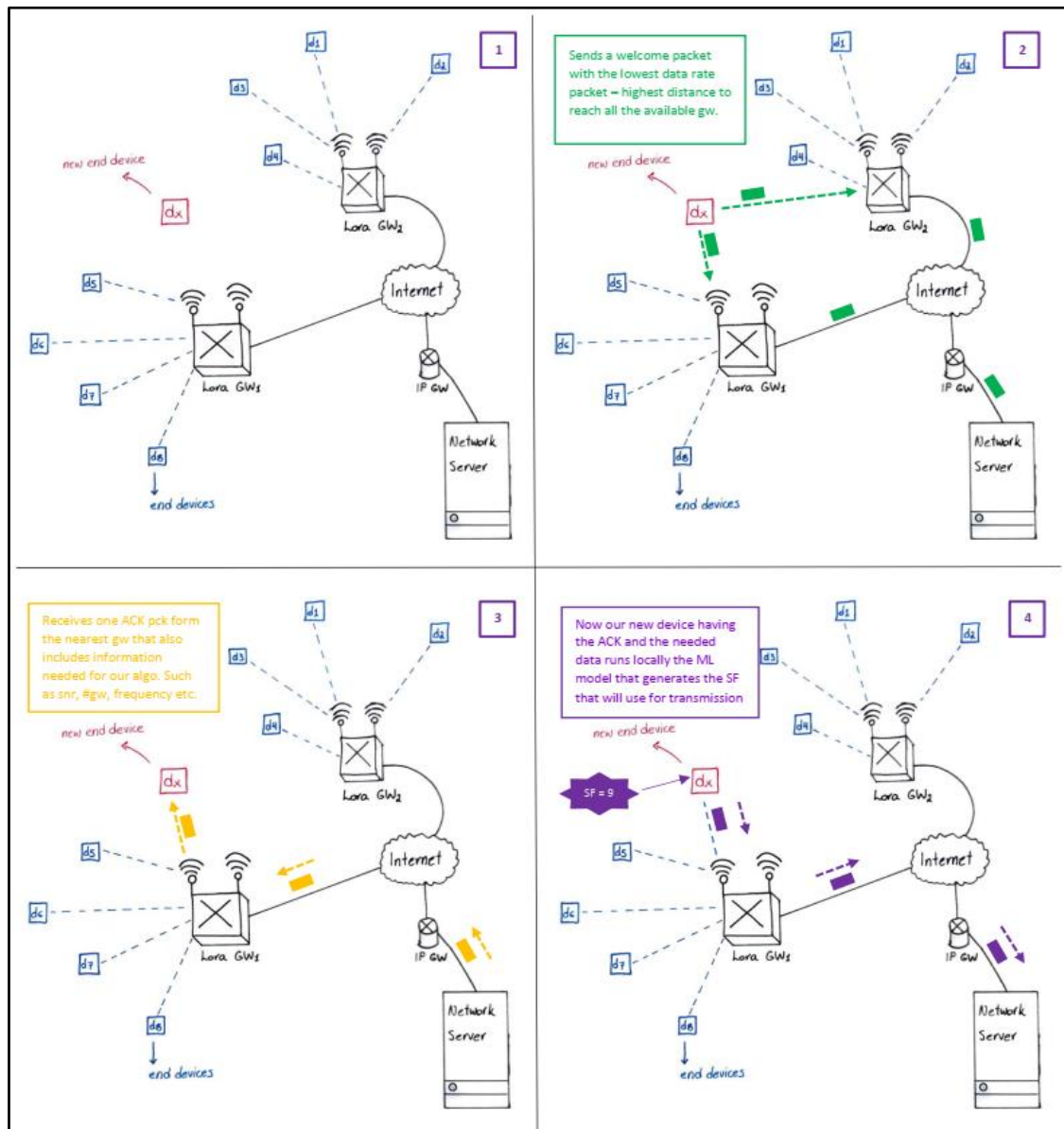
5.1 Γενικές Πληροφορίες Υλοποίησης

Με βάση λοιπόν όλα τα πιο πάνω δεδομένα καταλήξαμε στην δημιουργία ενός σεναρίου το οποίο αποτέλεσε την βάση για να κτίσουμε το δικό μας μοντέλο. Το σενάριο αυτό φαίνεται στην Εικόνα 8 και αποτελείται από τέσσερα βήματα που περιγράφουν την λειτουργία του αλγορίθμου μας και πως αυτός θα επιλύει τα δύο προβλήματα που εντοπίσαμε στον αλγόριθμο ADR.

Στο σενάριο αυτό θεωρούμε ότι έχουμε ένα υφιστάμενο δίκτυο LoRa το οποίο αποτελείται από πολλαπλές τελικές συσκευές και πύλες μέσα σε ένα χώρο. Οι συσκευές αυτές μπορεί να είναι είτε κινητές είτε σταθερές. Στο δίκτυο αυτό λοιπόν προστίθεται μία νέα συσκευή η οποία δεν γνωρίζει το δίκτυο αλλά ούτε κάποιες παραμέτρους για την αποστολή πακέτου επομένως αποστέλλει ένα μήνυμα ένταξης της συσκευής (join request) στο δίκτυο. Για τα μηνύματα ένταξης της συσκευής στο δίκτυο γίνεται αυτόματα η χρήση του SF12. Αυτό γίνεται αφού η συσκευή δεν γνωρίζει την τοποθεσία των πυλών και κατά πόσο μπορεί να επικοινωνήσει με κάποια πύλη με την χρήση χαμηλότερου spreading factor. Αν η συσκευή δεν καταφέρει να επικοινωνήσει με καμία πύλη με την χρήση του SF12 τότε η συσκευή βρίσκεται σε πολύ μεγάλη απόσταση από κάθε διαθέσιμη πύλη και πρέπει να μετακινηθεί η τελική συσκευή ή να προστεθεί κάποια νέα πύλη στον χώρο που δρα η συγκεκριμένη συσκευή.

Αφού γίνει η επιτυχής ένταξη της συσκευής στο δίκτυο τότε αποστέλλεται το πρώτο μήνυμα από την συσκευή προς τον εξυπηρετητή όπως βλέπουμε στο βήμα 2. Στην παρούσα φάση η συσκευή και πάλι θα κάνει χρήση του SF12 μέχρις ότου λάβουμε την πρώτη απάντηση από

τον εξυπηρετητή δικτύου. Επομένως, στην πρώτη περίπτωση θεωρούμε ότι η συσκευή είναι εφοδιασμένη με ένα μοντέλο μηχανικής μάθησης το οποίο έχει εκπαιδευτεί από την άντληση δεδομένων από τις υπόλοιπες συσκευές του δικτύου. Σε αυτή την περίπτωση λοιπόν μετά την αποστολή του πρώτου μηνύματος από την συσκευή αναμένουμε την απάντηση επιβεβαίωσης από τον κεντρικό εξυπηρετητή όπου όπως βλέπουμε και στο βήμα 3 περιλαμβάνει εκτός από την επιβεβαίωση και τις πληροφορίες που χρειαζόμαστε για το πακέτο που στάλθηκε που όπως το SNR και το RSSI.



Από αυτό το σημείο και έπειτα ο αλγόριθμος μας είναι σε θέση να λειτουργεί κανονικά χωρίς να χρειαστεί οποιαδήποτε άλλη διαδικασία ή την συσκευή μας να βρίσκεται κάτω από σταθερές συνθήκες ραδιοκυμάτων. Έχοντας, όλες τις πληροφορίες που χρειάζεται η συσκευή, όπως φαίνεται στο βήμα 4 μπορεί να αποφασίσει η ίδια για την επιλογή του spreading factor για το επόμενο μήνυμα που θα στείλει μέσω του μοντέλου.

Αντίθετα, στην πλευρά των υφιστάμενων λύσεων σε περίπτωση όπου η συσκευή είναι κινητή ο αλγόριθμος ADR δεν θα ενεργοποιηθεί και έτσι η συσκευή θα κάνει χρήση είτε σταθερής τιμής spreading factor είτε του αλγόριθμου blind ADR αν αυτός υποστηρίζεται από την τελική συσκευή. Στην περίπτωση όπου η συσκευή είναι κάτω από σταθερές συνθήκες για την ενεργοποίηση του ADR χρειάζονται όπως είχαμε αναφέρει τα 20 τελευταία πακέτα uplink. Αυτό σημαίνει ότι χρειάζονται ακόμα 19 πακέτα για να λειτουργήσει πλήρως ο ADR, ενώ το μοντέλο μηχανικής μάθησης βρίσκεται ήδη σε πλήρη λειτουργία.

Μετά από αυτή την παρατήρηση καταλήξαμε στο συμπέρασμα ότι ο αλγόριθμος μας θα μπορούσε να λειτουργήσει εκτός από ανεξάρτητα και ως συμπληρωματικός αλγόριθμος για τον ADR. Συγκεκριμένα μπορεί το μοντέλο μηχανικής μάθησης να αποτελεί το μέσο διαρρύθμισης του spreading factor για τα επόμενα 19 πακέτα μέχρις ότου ο αλγόριθμος ADR βρίσκεται σε θέση να λειτουργήσει πλήρως.

Να διευκρινιστεί ότι η διαδικασία λειτουργίας του μοντέλου υποστηρίζεται και από τις τρεις κατηγορίες των συσκευών που υποστηρίζει το LoRaWAN. Αυτό είναι εφικτό αφού η πληροφορία αυτές λαμβάνονται στο παράθυρο που ανοίγει η συσκευή μετά την αποστολή του μηνύματος όπου είναι το παράθυρο για το μήνυμα επιβεβαίωσης το οποίο υποστηρίζεται και στις τρεις κατηγορίες.

5.2 Αρχικό σύνολο Δεδομένων

Αφού η διαδικασία και ο τρόπος λειτουργίας του αλγορίθμου έχουν οριστεί η πρώτη φάση υλοποίησης είχε ως στόχο την εύρεση του απαραίτητου αριθμού δεδομένων για να εκπαιδύσουμε ένα καλό μοντέλο. Στην φάση αυτή θέλαμε να δούμε κατά πόσο ένα μοντέλο μηχανικής μάθησης μπορεί να επιτύχει καλά αποτελέσματα σε δεδομένα πραγματικού περιβάλλοντος. Όπως προαναφέρθηκε η εκπαίδευση ενός μοντέλου μηχανικής μάθησης απαιτεί ένα τεράστιο όγκο δεδομένων. Η εύρεση ενός καθαρού και μεγάλου συνόλου δεδομένων ή δημιουργία ενός, είναι μία πολύ σημαντική παράμετρος για την επιτυχία ενός τέτοιου αλγορίθμου.

Η δημιουργία ενός νέου συνόλου δεδομένων για την εκπαίδευση του μοντέλου μέσα από δικές μας μετρήσεις δεν ήταν μια εφικτή λύση λόγω του περιορισμένου χρόνου, της έλλειψης ικανοποιητικού αριθμού τελικών συσκευών, αλλά και η προσομοίωση των μη σταθερών συνθηκών (όπως κινητικότητα) για να έχουμε ένα πλήρες σύνολο δεδομένων. Για όλους του προαναφερθέν λόγους, προχωρήσαμε στην εύρεση ενός υπάρχον συνόλου δεδομένων. Το

σύνολο δεδομένων που επιλέξαμε προέρχεται από μία ομάδα του Imperial College London [4], οι οποίοι προχώρησαν στην δημιουργία ενός συνόλου δεδομένων μέσα από ένα δίκτυο που έχουν δημιουργήσει στην πόλη του Λονδίνου.

Το σύνολο δεδομένων αποτελείται από συνολικά 11 εκατομμύρια γραμμές και 9 στήλες. Από το σύνολο των γραμμών αφαιρέθηκαν όλες οι γραμμές με κενές τιμές σε στήλες απαραίτητες για την εύρεση αποτελέσματος. Οι στήλες που περιέχονται μέσα στο σύνολο αυτό των δεδομένων είναι:

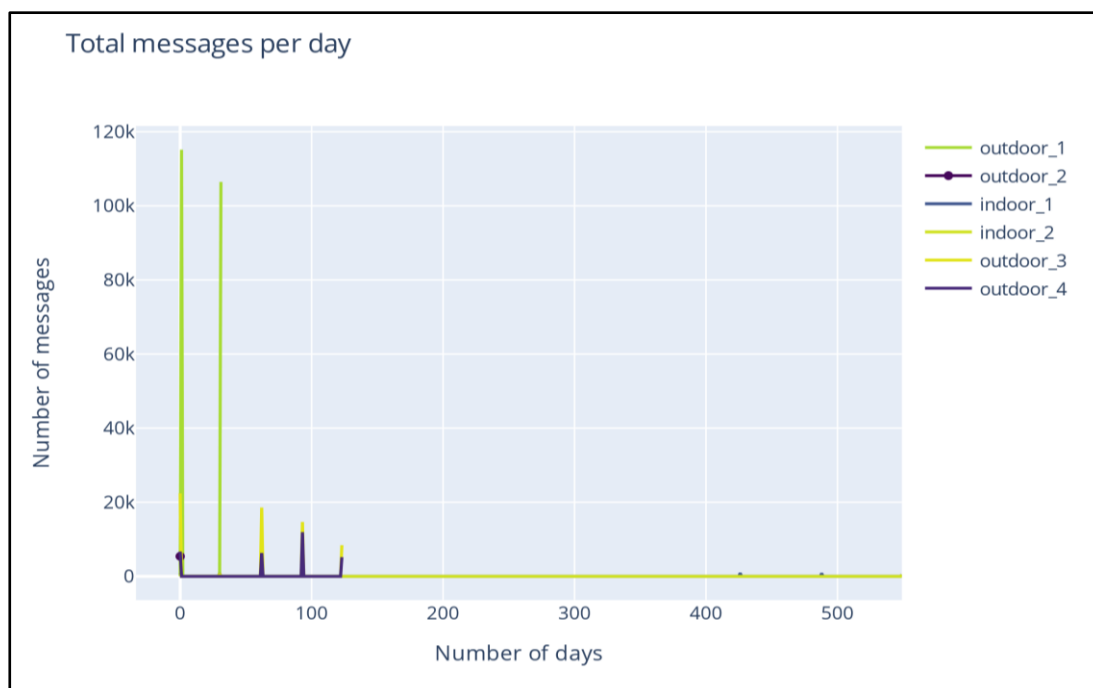
- Χρόνος - Time: Ο χρόνος παραλαβής του πακέτου στον κεντρικό εξυπηρετητή.
- Πύλη - Gateway: Η πύλη που παρέλαβε το πακέτο.
- Συχνότητα - Frequency: Συχνότητα αποστολής του πακέτου.
- Spreading Factor: Συντελεστής Διασποράς αποστολής πακέτου.
- Εύρος ζώνης - Bandwidth: Εύρος ζώνης του πακέτου.
- Code Rate: ρυθμός μετάδοσης σήματος.
- RSSI: Ένδειξη ισχύος λήψης σήματος για το πακέτο που λήφθηκε.
- Snr: Ο λόγος σήματος προς θόρυβο για το πακέτο.
- Τύπος - Mtype: Ο τύπος του πακέτου που λήφθηκε.

Τα δεδομένα αυτά αντλήθηκαν με την χρήση εννέα συνολικά πυλών. Οι τρεις από αυτές τις πύλες είχαν αντλήσει πολύ μικρό αριθμό πακέτων για αυτό και δεν συμπεριλήφθηκαν στα πιο κάτω αποτελέσματα. Από τις έξι εναπομείναντες πύλες οι τέσσερις ήταν τοποθετημένες σε εξωτερικούς χώρους και οι δύο σε εσωτερικούς. Οι εξωτερικές πύλες τοποθετήθηκαν στην οροφή ψηλών κτηρίων για να μεγιστοποιηθεί το εύρος κάλυψης τους ενώ οι εσωτερικές κοντά σε παράθυρα και πάλι για τον ίδιο λόγο.

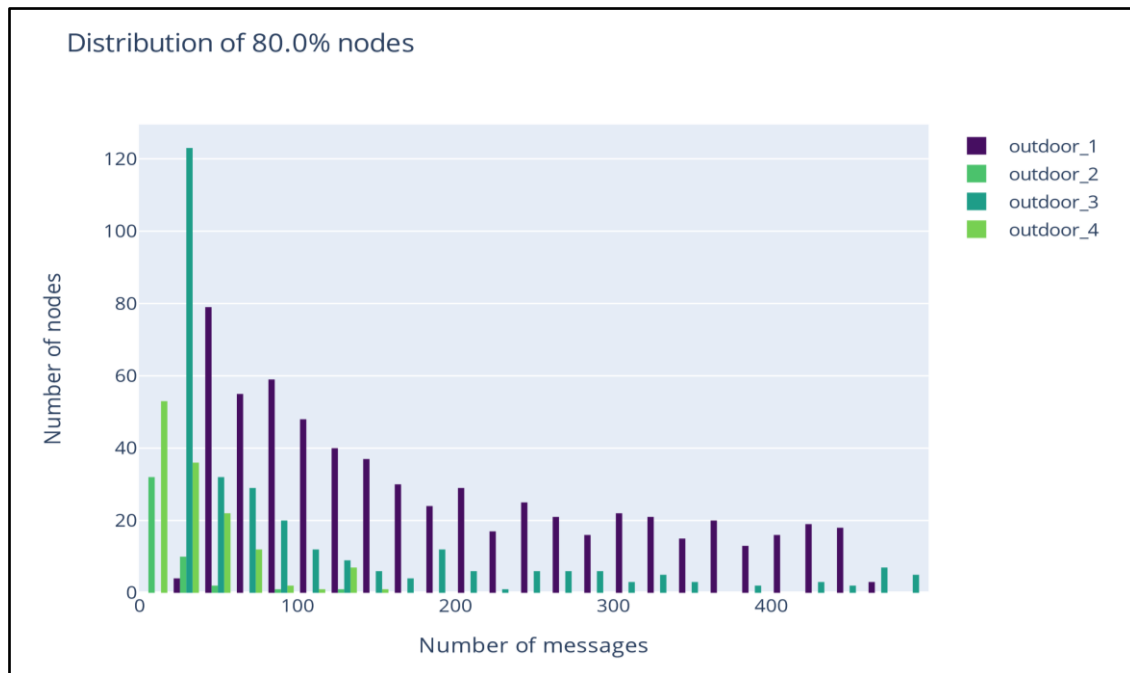
Πρώτο βήμα ήταν η ανάλυση των δεδομένων αυτών παράγοντας κάποιες γραφικές παραστάσεις. Με την διαδικασία αυτή στόχος μας ήταν αν εξάγουμε όσο το δυνατό περισσότερα συμπεράσματα για την μορφή του δικτύου αλλά και για τα δεδομένα τα ίδια. Επίσης μέσα από τις πιο κάτω γραφικές παραστάσεις μπορούμε να έχουμε μια πρώτη εικόνα για διάφορες παραμέτρους των δεδομένων που θα μας οδηγήσουν στην καταλληλότητα της χρήσης του για την εκπαίδευση του μοντέλου μας'.

Στις δύο πρώτες γραφικές παρουσιάζεται σε γενικές γραμμές η κατανομή των μηνυμάτων ανα πύλη. Στην πρώτη γραφική παράσταση έχουμε τον συνολικό αριθμό πακέτων που λήφθηκαν από κάθε πύλη. Μπορούμε να δούμε ξεκάθαρα ότι οι τέσσερις πύλες που βρίσκονται

τοποθετημένες σε εξωτερικούς, ανοικτούς χώρους με μεγάλη κάλυψη έχουν παραλάβει το 80% περίπου των μηνυμάτων που λήφθηκαν όλες τις μέρες. Η πρώτη εξωτερική πύλη έχει παραλάβει τα περισσότερα συνολικά μηνύματα με την πρώτη μόνο μέρα να φτάνει τα 120 χιλιάδες μηνύματα. Αυτό μπορεί να επιβεβαιωθεί και στην δεύτερη γραφική παράσταση που έχουμε τον αριθμό των τελικών συσκευών που εξυπηρετήσαν οι τέσσερις εξωτερικές πύλες συνάρτηση του αριθμού των μηνυμάτων. Βλέπουμε και πάλι ότι η πύλη 1 έχει λάβει σταθερά καθ' όλη την διάρκεια του χρόνου τα περισσότερα πακέτα. Ο λόγος φαίνεται και πάλι μέσα από την γραφική παράσταση 2 όπου μέσα από τον άξονα y παρατηρούμε ότι η πύλη 1 εξυπηρετεί τον μεγαλύτερο αριθμό τελικών συσκευών από κάθε άλλη πύλη.



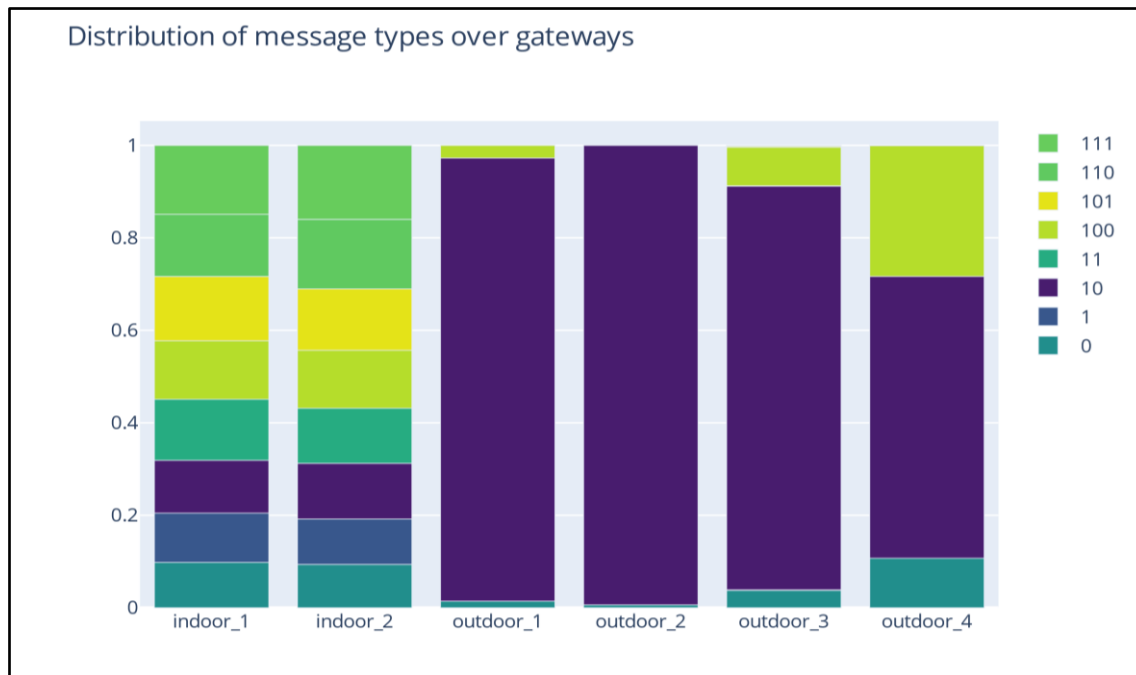
Γραφική Παράσταση 1: Ο συνολικός αριθμός πακέτων ανά μέρα



Γραφική Παράσταση 2: Ο συνολικός αριθμός τελικών συσκευών ανά πύλη

Στην Τρίτη γραφική παράσταση έχουμε τον καταμερισμό του είδους των μηνυμάτων που λήφθηκαν ανά πύλη. Αυτό μας βοηθά στην κατανόηση της ροής των μηνυμάτων στις πύλες. Στις πύλες εξωτερικών χώρων μπορούμε να διακρίνουμε ξεκάθαρα ότι τα πλείστα είναι μηνύματα πληροφοριών χωρίς ή με επιβεβαίωση. Αυτό μας βοηθά να δούμε ότι το δίκτυο μας αποτελείται κυρίως από σταθερό αριθμό τελικών συσκευών οι οποίες αποστέλλουν μηνύματα δεδομένων. Αν αυτό δεν ίσχυε τότε θα διακρίναμε μεγαλύτερο αριθμό μηνυμάτων αιτήματος εισόδου στο δίκτυο ή αποδοχής εισόδου (join request or join accept). Αντίθετα, στις δύο πύλες που είναι τοποθετημένες σε εσωτερικούς χώρους βλέπουμε μια ίση κατανομή των μηνυμάτων.

Η κατανομή αυτή αποτελεί ένα ενδιαφέρον αποτέλεσμα αφού η ύπαρξη μεγάλου αριθμού τόσο μηνυμάτων εισόδου στο δίκτυο αλλά και μηνυμάτων downlink είναι περιπτώσεις οι οποίες δεν καλύπτονται από τις εξωτερικές πύλες. Αυτά τα μηνύματα θα δώσουν την δυνατότητα στο μοντέλο μας να εκπαιδευτεί καλύτερα και να παράγει αποτελέσματα ορθά ακόμη και σε πιο ακραίες περιπτώσεις.

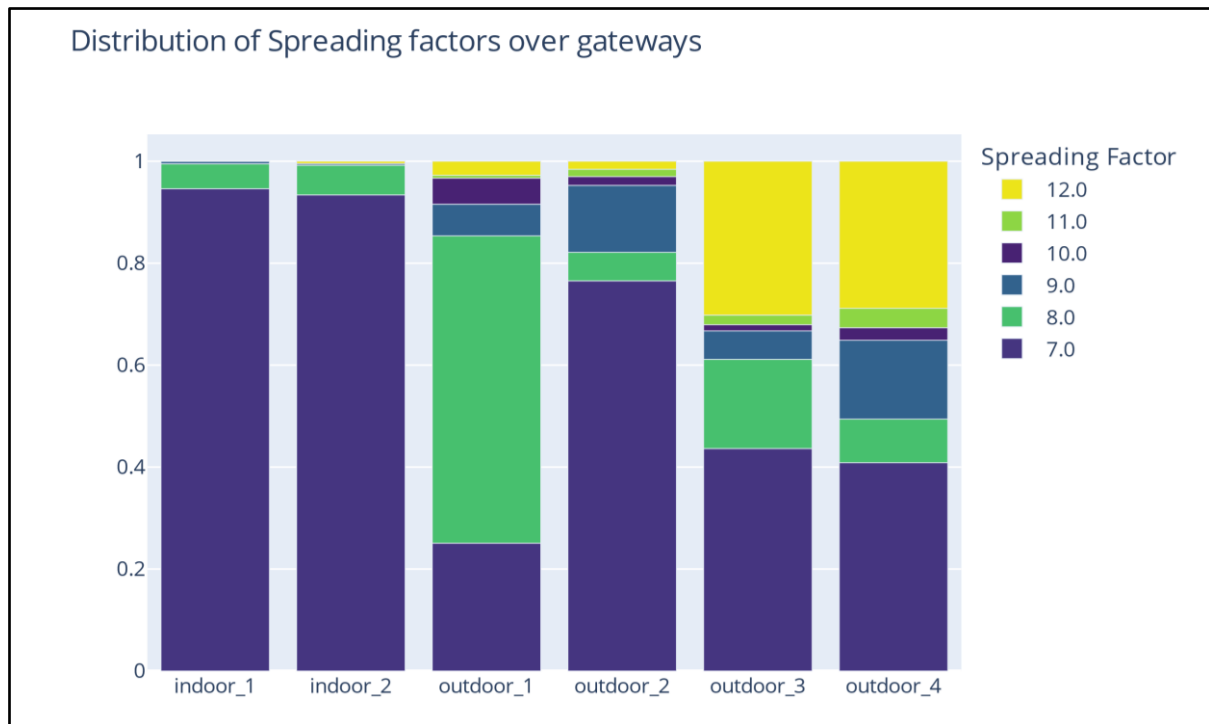


Γραφική Παράσταση 3: Καταμερισμός του είδους μηνύματος που λήφθηκαν ανά πύλη

Στην τελευταία και πιο σημαντική γραφική παράσταση έχουμε τον καταμερισμός των τιμών Spreading Factor ανά πύλη. Όπως μπορούμε να δούμε από την γραφική παράσταση οι τιμές των Spreading Factor δεν είναι ίσα κατανομημένες σε όλες τις τιμές. Όπως διακρίνεται από την γραφική παράσταση στις πύλες που είναι τοποθετημένες σε εξωτερικούς χώρους υπάρχει μία άνιση κατανομή, ωστόσο βλέπουμε ότι έστω και σε χαμηλά ποσοστά γίνεται η χρήση όλου του φάσματος των τιμών. Αυτό θα μας δώσει την δυνατότητα να έχουμε έστω κάποιο μικρό αριθμό δεδομένων για κάθε τιμή.

Αντίθετα και πάλι παρατηρούμε μια διαφορετική προσέγγιση στις εσωτερικές πύλες οι οποίες παρουσιάζουν μια άνιση κατανομή των τιμών. Στις πύλες αυτές γίνεται χρήση μόνο των τιμών 7 και 8. Το αποτέλεσμα αυτό είναι αναμενόμενο αφού οι πύλες αυτές σκοπό έχουν την εξυπηρέτηση συσκευών εντός ενός κτηρίου. Επομένως, οι συσκευές αυτές θα βρίσκονται σε μικρές αποστάσεις, για τα δεδομένα των δικτύων LoRa, της τάξεως των δεκάδων μέτρων.

Επομένως έχοντας αναλύσει σε υψηλό επίπεδο αυτά τα δεδομένα αποφασίσαμε ότι αποτελεί ένα αρκετό καλό σύνολο. Έτσι, προχωρήσαμε στην υλοποίηση του μοντέλου μηχανικής μάθησης με την χρήση των δεδομένων αυτών.



Γραφική Παράσταση 4: Καταμερισμός των τιμών Spreading Factor ανά πύλη

5.3 Δημιουργία Μοντέλου

Για την δημιουργία του μοντέλου μηχανικής μάθησης σε αρχική φάση έγινε χρήση γνωστών αλγορίθμων κατηγοριοποίησης. Ο λόγος της επιλογής αυτής είναι ότι το πρόβλημα μας είναι ένα πρόβλημα κατηγοριοποίησης έξι συνολικά κατηγοριών όσες και οι τιμές των spreading factors (SF7 - SF12). Δηλαδή, δεδομένου ενός αριθμού από στήλες εισόδου η επιθυμητή έξοδος μας είναι μία από τις έξι αυτές τις κατηγορίες οι οποίες θα καθορίζουν spreading factor και κατά συνέπεια τον ρυθμό μετάδοσης δεδομένων.

Για τα δεδομένα εισόδου του μοντέλου δεν χρησιμοποιήθηκαν όλες οι στήλες που υπάρχουν μέσα στο σύνολο δεδομένων μας. Ο λόγος είναι ότι κάποιες από αυτές δεν μπορούν να προσφέρουν κάποια χρήσιμη πληροφορία που να επηρεάζει την επιλογή του spreading factor. Η επιλογή των στηλών έχει γίνει με βάση τα δεδομένα που χρησιμοποιεί ο αλγόριθμος ADR αλλά και πληροφορίες που θεωρήσαμε εμείς χρήσιμες. Οι τελικές έξι στήλες που χρησιμοποιήθηκαν είναι Frequency, RSSI, SNR, bandwidth, packet Size και message type.

Οι πρώτες τρεις πληροφορίες Frequency, RSSI, bandwidth και SNR είναι πληροφορίες οι οποίες είναι πολύ σημαντικές για την απόφαση του ρυθμού μετάδοσης δεδομένων και χρησιμοποιούνται επίσης και από τον αλγόριθμο ADR. Οι πληροφορίες αυτές μας δίνουν περιγράφουν την κατάσταση στην οποία βρίσκεται το δίκτυο την παρούσα στιγμή.

Λαμβάνοντας υπόψη τις πληροφορίες αυτές μπορούμε να καταλάβουμε την χωρητικότητα του δικτύου αλλά και την συμμόρφωση που μπορεί να υπάρχει την δεδομένη στιγμή.

Οι άλλες δύο τιμές εισόδου, packet size και mtype δεν χρησιμοποιούνται από τον αλγόριθμο ADR όμως για το δικό μας μοντέλο είναι πολύ σημαντικές. Ο λόγος είναι ότι αυτές οι δύο πληροφορίες όπου για κάθε πακέτο είναι διαφορετικές και είναι τιμές οι οποίες τοποθετούνται από την τελική συσκευή στην κεφαλή του πακέτου. Δηλαδή, ο κεντρικός εξυπηρετητής δεν μπορεί να γνωρίζει τι είδος μπορεί να είναι το επόμενο πακέτο που θα στείλει η τελική συσκευή όπως ούτε το μέγεθος του πακέτου. Στην περίπτωση μας όμως, η τελική συσκευή η οποία αποφασίζει τις πληροφορίες αυτές είναι αυτή που θα αποφασίσει και την τιμή του spreading factor επομένως μπορούμε να λάβουμε και αυτές τις παραμέτρους υπόψη.

Όπως προαναφέρθηκε σε αρχική φάση προχωρήσαμε με την δοκιμή δημιουργίας μοντέλων με γνωστούς αλγορίθμους. Η προσέγγιση αυτή θα μας βοηθήσει στο να έχουμε μια γενική εικόνα για το αναμενόμενο ποσοστό επιτυχίας ενός μοντέλου, με αλγόριθμους οι οποίοι χρησιμοποιούνται σε πολλές εφαρμογές με υψηλά ποσοστά επιτυχίας. . Για την εκπαίδευση των αλγορίθμων το σύνολο δεδομένων χωρίστηκε σε 60% δεδομένα εκπαίδευσης (training set), 20% δεδομένα επικύρωσης (validation set) και 20% δεδομένα ελέγχου (test set). Οι αλγόριθμοι που χρησιμοποιήθηκαν για την δημιουργία των μοντέλων είναι:

- K-nearest neighbors' classifier (KNN) [k=6]
- Support vector classifier (SVM)
- Gaussian naive bayes classifier (GNB)
- Decision tree classifier (DTC)
- Linear discriminant analysis classifier (LDA)
- Random forest classifier (RFC)

Στον Πίνακα 5 έχουμε τα αποτελέσματα των 6 αλγορίθμων. Τα καλύτερα αποτελέσματα παράχθηκαν από τους αλγόριθμους RFC και KNN οι οποίοι είχαν ποσοστό επιτυχίας κοντά στο 78%. Πολύ κοντά ήταν και ο αλγόριθμος DTC με ποσοστό 77% κάτι που ήταν αναμενόμενο αφού ο τρόπος λειτουργίας του είναι πολύ κοντά στην λογική του RFC. Τα αποτελέσματα αυτά ωστόσο όπως μπορούμε να διακρίνουμε δεν είναι και τόσο ικανοποιητικά γνωρίζοντας ότι τέτοια μοντέλα μπορούν εύκολα να φτάσουν ποσοστό επιτυχίας πάνω από 90%. Τα πιο κάτω αποτελέσματα ήταν τα καλύτερα που κατάφεραν οι αλγόριθμοι να παρουσιάσουν με διαφορετικούς αριθμούς εποχών και παραμέτρων εκπαίδευσης. Με γνώμονα λοιπόν τις δυνατότητες των αλγορίθμων αυτών και τα αποτελέσματα που παρουσίασαν δεν

είμασταν σε μία ικανοποιητική θέση έτσι προχωρήσαμε στον εμπλουτισμό των δεδομένων μας.

Πίνακας 5: Αποτελέσματα των αλγορίθμων, Ποσοστό επιτυχίας και τυπική απόκλιση.

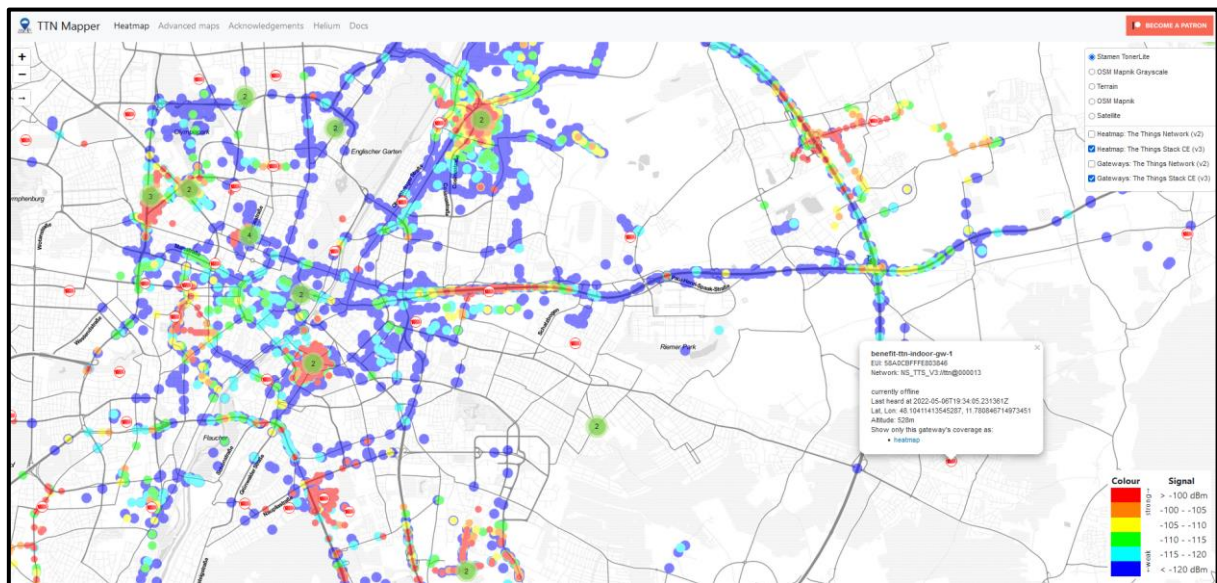
Αλγόριθμος	Ποσοστό Επιτυχίας	Τυπική Απόκλιση
KNN	0.7756798	0.00179
SVM	0.4533025	0.00188
GNB	0.5389549	0.00270
DTC	0.7698443	0.00204
LDA	0.5730204	0.00184
RFC	0.7787461	0.00134

5.4 Επέκταση δεδομένων

Με βάση τα αποτελέσματα της προηγούμενης φάσης προχωρήσαμε στην εύρεση και άλλων δεδομένων. Ο λόγος αυτής της απόφασης ήταν κάποια προβλήματα που αντιμετωπίζει το συγκεκριμένο σύνολο δεδομένων. Τα προβλήματα αυτά είναι η έλλειψη πληροφορίας για την εφαρμογή της τελικής συσκευής και η άνιση άντληση πληροφοριών από κάθε πύλη. Για το πρώτο σκέλος όπως είχαμε παρατηρήσει και στην γραφική παράσταση τρία, διακρίνουμε ότι οι πλείστες πληροφορίες είναι μηνύματα πληροφοριών χωρίς ή με επιβεβαίωση. Η παρατήρηση αυτή μας επιβεβαιώνει ότι οι εφαρμογές στις τελικές συσκευές λειτουργούν σε σταθερές συνθήκες ραδιοκυμάτων και δεν υπάρχει ούτε κινητικότητα. Επομένως, τα δεδομένα αυτά δεν αντιπροσωπεύουν πλήρως όλες τις περιπτώσεις που αναζητούμε. Επίσης, τα δεδομένα αυτά δεν είχαν τα επιθυμητά αποτελέσματα όσον αφορά το ποσοστό επιτυχίας.

Για την επέκταση του συνόλου δεδομένων επομένως χρειάστηκε η αναζήτηση δεδομένων όπου είτε να υπάρχει ως στήλη το είδος της εφαρμογής είτε να υπάρχει δεδομένη η σχετική πληροφορία για όλες τις εφαρμογές. Καταλήξαμε, στην άντληση των δεδομένων από την εφαρμογή TTN Mapper. Η συγκεκριμένη εφαρμογή έχει δημιουργηθεί για να προσφέρει σε ανοιχτή μορφή τα δεδομένα που διαπερνούν όλες τις ανοιχτές πύλες που υπάρχουν. Υποστηρίζει κυρίως πύλες του The Things Network αλλά όχι μόνο. Όπως φαίνεται και στην Εικόνα 6 η ποσότητα της αποστολής πακέτων αλλά και εύρεσης πυλών παρουσιάζεται με την μορφή θερμικού χάρτη (heat map). Για κάθε πύλη όπως μπορούμε να δούμε υπάρχει η μοναδική της ταυτότητα με την οποία μπορούμε να αντλήσουμε τα δεδομένα που έχουν ληφθεί

η αποσταλεί από τη συγκεκριμένη συσκευή. Τα δεδομένα αυτά μπορούν να αντληθούν πλοηγώντας στην επιλογή advanced map στο κεντρικό μενού όπως φαίνεται και στην Εικόνα 7. Το TTN Mapper υποστηρίζει την αποθήκευση των τελευταίων 100 000 μηνυμάτων κάθε συσκευής. Επομένως, με την βοήθεια του χάρτη προχωρήσαμε στην εύρεση πυλών όπου λαμβάνουν μεγάλο αριθμό πακέτων και έχουν δυνατό σήμα δηλαδή έχουν κόκκινο χρώμα στον θερμικό χάρτη. Τα δεδομένα που αντλήθηκαν ήταν από διάφορες χώρες και πόλεις ώστε να καταφέρουμε να έχουμε όσο το δυνατό πιο μεγάλο εύρος περιπτώσεων.



Εικόνα 6: Αποτελέσματα εύρεσης πυλών και τελικών συσκευών στο Μόναχο.

The image displays the 'Advanced Maps' section of the TTN Mapper application. It contains two main search forms: 'Device data' and 'Gateway data'. Each form includes a text input for the ID, date range selectors for 'Start Date' and 'End Date', and checkboxes for visualization options like 'Show markers for gateways' and 'Draw lines between gateway and measurement location'. At the bottom of each form are buttons for 'View Map' and 'CSV data'. The interface is clean and functional, designed for data analysis and visualization.

Εικόνα 7 : Αναζήτηση δεδομένων Πύλης ή συσκευής με την χρήση το device id

Τα δεδομένα αυτά περιέχουν ως στήλες όλες τις πληροφορίες που είχαμε δει στο προηγούμενο σύνολο δεδομένων όπως επίσης και επιπρόσθετες στήλες όπως ο τύπος της εφαρμογής που τρέχει στην τελική συσκευή και ακριβές συντεταγμένες για την τοποθεσία της συσκευής. Οι πληροφορίες αυτές ήταν χρήσιμες στην δημιουργία ενός συνόλου όπου μπορούμε να επιβεβαιώσουμε ότι υπάρχει ένα ικανοποιητικό ποσοστό συσκευών οι οποίες είναι συσκευές με κινητικότητα, κάτι που σημαίνει ότι ο αλγόριθμος ADR δεν είναι ενεργοποιημένος. Επιπλέον, όπως είχαμε δει και προηγουμένως η εφαρμογή TTN Mapper δίνει την δυνατότητα αναζήτησης δεδομένων συγκεκριμένης περιόδου, έτσι η άντληση δεδομένων από κάθε περιοχή έγινε για διαφορετικές χρονικές περιόδους.

Με αυτό τον τρόπο στα δεδομένα μας συμπεριλαμβάνονται δεδομένα από διαφορετικές γεωγραφικές θέσεις, διάφορων καιρικών συνθηκών και με τις πλείστες συσκευές με κινητικότητα. Για σκοπούς κάλυψης όλων των πιθανόν συνθηκών στο σύνολο αυτό προστέθηκε και ένας μικρός αριθμός δεδομένων και από το πρώτο σύνολο δεδομένων που αναλύσαμε. Καταλήξαμε, σε ένα σύνολο δεδομένων 5.3 εκατομμυρίων γραμμών από 6 διαφορετικές χώρες και 9 συνολικά πόλεις.

5.5 Αποτελέσματα με τα νέα Δεδομένα

Για το νέο σύνολο δεδομένων προχωρήσαμε στην ίδια ακριβώς διαδικασία και υλοποίηση που ακολουθήθηκε στο πρώτο σύνολο. Βασική διαφορά ήταν η εκτέλεση μόνο τριών από τον έξι αλγορίθμων που χρησιμοποιήθηκαν στην προηγούμενη φάση. Οι τρεις αλγόριθμοι ήταν αυτοί που είχαν παρουσιάσει τα καλύτερα αποτελέσματα.

Πίνακας 6: Αποτελέσματα των αλγορίθμων, Ποσοστό επιτυχίας και τυπική απόκλιση.

Αλγόριθμος	Ποσοστό Επιτυχίας	Τυπική Απόκλιση
KNN	0.856714	0.00073
LDA	0.813647	0.00070
RFC	0.861188	0.00054

Στον Πίνακα 6 μπορούμε να δούμε τα αποτελέσματα των αλγορίθμων, με τα καλύτερα αποτελέσματα και πάλι να προέρχονται από τον αλγόριθμο RFC με ποσοστό επιτυχίας το 86% και τυπική απόκλιση στα επίπεδα των μερικών χιλιοστών. Τα αποτελέσματα αυτά είναι πάρα πολύ ικανοποιητικά ωστόσο συνεχίζουν να βρίσκονται κάτω του 90% που ήταν το κατώτατο

ώριο που θέσαμε σε αποτελέσματα εκπαίδευσης. Το αποτέλεσμα της εκπαίδευσης είναι αναμενόμενο ότι δεν θα είναι πλήρως αντιπροσωπευτικά σε πραγματικά περιβάλλοντα αφού υπάρχουν πολλοί παράγοντες τους οποίους δεν μπορούμε να προβλέψουμε.

Για να καταφέρουμε να πάρουμε το επιθυμητό προχωρήσαμε στην υλοποίηση ενός νευρωνικού δικτύου για να ελέγξουμε κατά πόσο είναι εφικτό να πάρουμε τα επιθυμητά αποτελέσματα. Το πρόβλημα της επιλογής του spreading factor όπως αναφέραμε αντιμετωπίζεται ως ένα πρόβλημα κατηγοριοποίησης 6 διαφορετικών κατηγοριών με 6 δεδομένα εισόδου. Με βάση τα δεδομένα αυτά προχωρήσαμε στην δημιουργία ενός μοντέλου νευρωνικού δικτύου με ένα επίπεδο εισόδου με έξι νευρώνες, δύο κρυφά επίπεδα νευρώνων και ένα νευρώνα εξόδου.

Ο αριθμός νευρώνων στα δύο κρυφά επίπεδα ήταν μια μεταβλητή η οποία λόγω της δομής των νευρωνικών δικτύων δεν μπορούμε να είμαστε βέβαιοι για τον κατάλληλο αριθμό, έτσι ήταν δύο μεταβλητές για τις οποίες δοκιμάστηκαν διάφορες τιμές. Οι τιμές οι οποίες επιλέξαμε ήταν το σύνολο $\{4,6,8,10,16\}$ και για τα δύο επίπεδα. Αυτό μας οδήγησε στην δημιουργία 25 συνολικά μοντέλων νευρωνικών δικτύων με όλους τους πιθανούς συνδυασμούς των τιμών του πιο πάνω συνόλου. Όλα τα μοντέλα εκπαιδεύτηκαν επίσης για διάφορους αριθμούς εποχών μέσα στο σύνολο $\{25,50,100\}$. Όπως αναμέναμε παρατηρήσαμε ότι το νευρωνικό μας δίκτυο στις τιμές των 8 και πάνω νευρώνων δεν παρουσίαζε κάποια διαφορά στα αποτελέσματα, όπως και στο αριθμό των εποχών στις 100 εποχές αντιμετωπίζαμε προβλήματα υπερπροσαρμογής (overfitting).

Στα μοντέλα όπου ο αριθμός των νευρώνων για κάθε κρυφό επίπεδο ήταν διαφορετικός (πχ. 4-6, 8-10 κλπ.) το αποτέλεσμα δεν είχε κάποια αισθητή βελτίωση σε σχέση με τα μοντέλα με τον ίδιο αριθμό νευρώνων και στα δύο κρυφά επίπεδα. Επιπλέον, διαφοροποίηση στα αποτελέσματα δεν παρουσιάστηκε ούτε και στα μοντέλα όπου ο αριθμός των νευρώνων ήταν μεγαλύτερος του 8. Όπως βλέπουμε από τους 6 νευρώνες είχαμε φθάσει σε ένα πάρα πολύ καλό επίπεδο επιτυχίας λόγω του μεγέθους του προβλήματος μας που είναι οι έξι κατηγορίες ωστόσο με 8 νευρώνες είχαμε ελάχιστα βελτιωμένο μέσο τετραγωνικό σφάλμα.

Στον πίνακα 7 λοιπόν βλέπουμε τα καλύτερα αποτελέσματα που καταφέραν να παρουσιάσουν τα διάφορα μοντέλα. Από τους 4 κιάλας νευρώνες το μοντέλο μας κατάφερε να παρουσιάσει ποσοστό επιτυχίας 93% κάτι που ήδη είχε ξεπεράσει το κατώτατο όριο που θέσαμε. Παρόλα αυτά το η χρήση 6 και 8 νευρώνων αντίστοιχα κατάφεραν να παρουσιάσουν ποσοστό επιτυχίας 97% το οποίο αποτελεί ένα εξαιρετικό αποτέλεσμα. Έχοντας αντλήσει τέτοια υψηλά ποσοστά

επιτυχίας βρισκόμασταν σε θέση να δείξουμε ότι ένα τέτοιο μοντέλο μπορεί να εξάγει τα αποτελέσματα που αναζητούμε.

Πίνακας 7: Αποτελέσματα εκπαίδευσης μοντέλων νευρωνικών δικτύων

Αριθμός νευρώνων 1 ^ο επιπέδου	Αριθμός νευρώνων 2 ^ο επιπέδου	Ποσοστό Επιτυχίας	Μέσο τετραγωνικό σφάλμα
4	4	93%	0.0841504
6	6	97%	0.0383710
8	8	97%	0.0319805

Κεφάλαιο 6

Αξιολόγηση Αλγορίθμου

6.1 Αξιολόγηση αλγορίθμου	44
6.2 Χρήση προσομοιωτή LoRaMab	44
6.3 Επέκταση προσομοιωτή LoRaMab	46
6.4 Άντληση δεδομένων από τον προσομοιωτή	47
6.5 Αποτελέσματα Αλγορίθμων	49

6.1 Αξιολόγηση αλγορίθμου

Μετά την δημιουργία του μοντέλου επόμενο βήμα ήταν η αξιολόγηση του αλγορίθμου συγκριτικά με τις υφιστάμενες λύσεις. Ο έλεγχος του αλγορίθμου σε πραγματικό περιβάλλον ήταν μια διαδικασία που δεν ήταν εφικτή λόγω χρόνου και έλλειψης των απαραίτητων συνθηκών και υλικού. Λόγο αυτών των αδυναμιών προχωρήσαμε στην εύρεση ενός προσομοιωτή ο οποίος θα μας βοηθήσει στην αξιολόγηση του αλγορίθμου μας.

Η εύρεση προσομοιωτή με υποστήριξη τόσο uplink και downlink μηνυμάτων όπως και διαχειριζόμενες τελικές συσκευές μας οδήγησε σε ένα σύνολο από τρεις μόνο προσομοιωτές που να υποστηρίζουν μέρος των προδιαγραφών αυτών [5][6][7]. Η κινητικότητα που είναι μια βασική προδιαγραφή δεν υποστηριζόταν από κανένα από τους προσομοιωτές. Επομένως, οδηγηθήκαμε στην επιλογή του προσομοιωτή ο οποίος παρέχει την πιο εύκολη υποδομή για επέκταση και προσθήκη δικών μας προδιαγραφών. Ο προσομοιωτής της επιλογής μας ήταν ο LoRa-MAB.

6.2 Χρήση προσομοιωτή LoRa-MAB

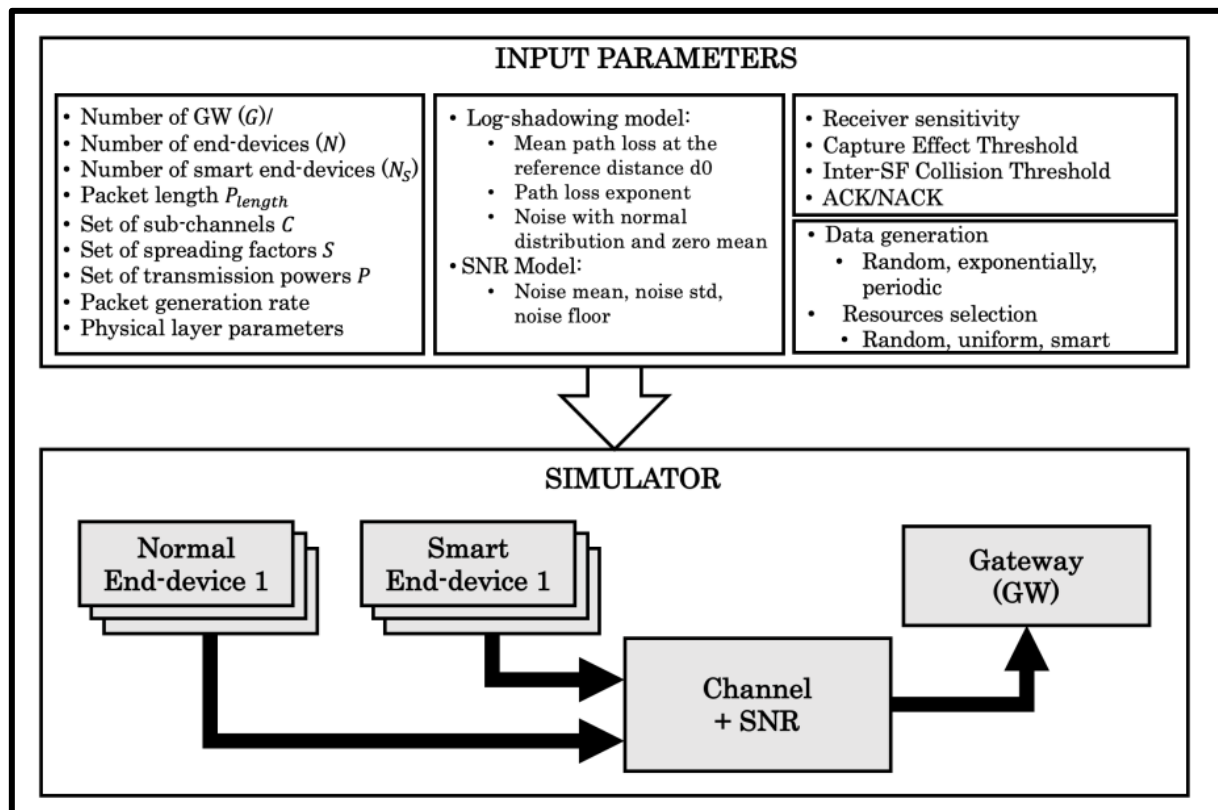
Ο προσομοιωτής LoRa-MAB είναι ένας ανοικτού πηγαίου κώδικα ο οποίος δημιουργήθηκε στα πλαίσια μιας ερευνητικής προσπάθειας [8] από την συνεργασία Γάλλων επιστημών από 4 πανεπιστήμια. Η δημιουργία του προσομοιωτή αυτού έγινε με απώτερο σκοπό να δοκιμαστεί από την ομάδα αυτή ερευνητών ένας αλγόριθμος τεχνητής νοημοσύνης αλλά μη

επιβλεπόμενης μάθησης, ο $\exp 3$. Η προσέγγιση τους ωστόσο ήταν η σύγκριση του συγκεκριμένου αλγόριθμου σε καταστάσεις σταθερού περιβάλλοντος συγκριτικά με ένα αλγόριθμο τυχαίας επιλογής τιμής για το spreading factor.

Ο προσομοιωτής έχει δημιουργηθεί σε γλώσσα προγραμματισμού Python και είναι κτισμένο πάνω στην βιβλιοθήκη SimPy [7]. Ο τρόπος λειτουργίας του προσομοιωτή είναι αρκετά απλός. Θεωρούμε ένα δίκτυο LoRaWAN που αποτελείται από μια ή περισσότερες πύλες, όπου καθεμία βρίσκεται στο κέντρο ενός κυκλικού δικτύου ακτίνας R , με N τελικές συσκευές ομοιόμορφα κατανομημένες στον χώρο. Για απλότητα, οι λειτουργίες του κεντρικού εξυπηρετητή ενσωματώνονται στην πύλη. Έτσι, στην υλοποίηση του προσομοιωτή υπάρχουν 3 συστατικά: τελικές συσκευές, μοντέλο προώθησης μηνυμάτων και πύλες. Η επιλογή των πόρων αποστολής των πακέτων διαχωρίζεται σε 2 είδη με βάση το είδος της τελικής συσκευής:

- Η κανονική τελική συσκευή που επιλέγει τους πόρους ακολουθώντας μια τυχαία επιλογή με χρήση αλγορίθμων παραγωγή τυχαίων αριθμών.
- Η έξυπνη τελική συσκευή που επιλέγει το πόρους καταφεύγοντας σε έναν αλγόριθμο τεχνητής νοημοσύνης.

Στην Εικόνα 8 απεικονίζεται η ροή λειτουργίας του προσομοιωτή με τα δεδομένα εισόδου που δίνονται από τον ίδιο τον χρήστη στην γραμμή εντολών αλλά και τις τιμές που ορίζει ο ίδιος ο προσομοιωτής για τις τιμές αρκετών παραμέτρων. Αρκετές από τις μεταβλητές εισόδου κατά την λειτουργία του προσομοιωτή τις κρατήσαμε σταθερές μεταξύ τους το σύνολο των τιμών για το spreading factor όπου περιλάμβανε όλες τις τιμές συνεχώς. Παράμετροι όπως ο αριθμός των κόμβων και ο αριθμός των έξυπνων κόμβων ήταν τιμές οι οποίες άλλαζαν για τα διάφορα σενάρια.



Εικόνα 8: Διάγραμμα λειτουργίας προσομοιωτή LoRa-MAB [8]

Επομένως, η λειτουργία του προσομοιωτή αυτού είναι ήδη πολύ κοντά σε αυτό που αναζητούσαμε για την αξιολόγηση του μοντέλου μας ωστόσο απουσίαζαν κάποια πολύ σημαντικά σημεία. Αυτά τα σημεία ήταν οι δύο αλγόριθμοι που θέλουμε να συγκρίνουμε, ο blind ADR και το δικό μας μοντέλο, καθώς και η κινητικότητα στις τελικές συσκευές όπου αναγκάζει τον αλγόριθμο ADR να μην ενεργοποιείται. Γι' αυτό τους λόγους προχωρήσαμε στην επέκταση του προσομοιωτή LoRa-MAB προσθέτοντας και αντικαθιστώντας τις λειτουργίες αυτές.

6.3 Επέκταση προσομοιωτή LoRa-MAB

Η επέκταση του προσομοιωτή LoRa-MAB χωρίστηκε σε δύο σκέλη. Πρώτο μέρος ήταν η αντικατάσταση των αλγορίθμων που γινόταν χρήση ανάλογα με το είδος της συσκευής και δεύτερο η προσθήκη της κινητικότητας στις τελικές συσκευές.

Για το πρώτο σκέλος χρειάστηκε να υλοποιήσουμε δύο συνολικά συναρτήσεις. Η πρώτη συνάρτηση ήταν η υλοποίηση του blind ADR η οποία αντικατέστησε την τυχαία επιλογή των πόρων που έκανε η κανονική συσκευή. Για τις έξυπνες τελικές συσκευές έγινε η αντικατάσταση του αλγορίθμου exp3 από την δική μας υλοποίηση. Η συνάρτηση αυτή

φορτώνει στον προσομοιωτή το μοντέλο και για κάθε αποστολή μηνύματος δίνει τα δεδομένα εισόδου και παίρνει ως έξοδο το spreading factor που θα χρησιμοποιήσει.

Στο δεύτερο σκέλος της επέκτασης χρειάστηκε μεγαλύτερη αλλαγή στον κώδικα. Πιο συγκεκριμένα στην βιβλιοθήκη SimPy η λογική λειτουργίας της είναι βασισμένη στην δημιουργία συναρτήσεων οι οποίες δρουν ως γεγονότα που πραγματοποιούνται. Επομένως, για κάθε τελική συσκευή που κινείται μέσα στο χώρο δημιουργήσαμε μια συνάρτηση - γεγονός η οποία αλλάζει την θέση της συσκευής μέσα στα όρια του χώρου. Η αλλαγή αυτή της θέσης γίνεται μέσα σε ένα χρονικό διάστημα το οποίο εμείς θέτουμε όπου ουσιαστικά πραγματοποιεί αυτό το γεγονός. Ως πρώτη υλοποίηση προχωρήσαμε στην τυχαία εναλλαγή συντεταγμένων για κάθε συσκευή όμως αυτό δεν ήταν καθόλου ρεαλιστικό σενάριο.

Για αυτό το λόγο προχωρήσαμε στην εύρεση βιβλιοθηκών όπου θα μας βοηθήσουν στην δημιουργία κίνησης η οποία θα συνάδει σε μία πραγματική κίνηση μια συσκευής μέσα στο χώρο. Η βιβλιοθήκη της επιλογής μας ήταν το Pymobility [8], μια ανοικτού κώδικα βιβλιοθήκη η οποία υλοποιεί τους πλείστους αλγορίθμους προσομοίωσης κινητικότητας. Ο αλγόριθμος της επιλογής μας ήταν ο στοχαστικός τυχαίος περίπατος (random stochastic walk) από και προς το κέντρο του χώρου όπου βρίσκεται η πύλη.

Η βιβλιοθήκη αυτή με βάση τα δεδομένα εισόδου όπου είναι το μέγεθος του χώρου και η ταχύτητα με την οποία κινείται το αντικείμενο προς τον στόχο παράγει ένα σύνολο από συντεταγμένες στο χώρο οι οποίες δημιουργούν ένα ρεαλιστικό σενάριο κίνησης μιας συσκευής μέσα στον χώρο. Επομένως, για κάθε τελική συσκευή δημιουργήσαμε ένα σύνολο κινήσεων μέσα στο χώρο αρχίζοντας από διαφορετική θέση, τις οποίες θα ακολουθούν κατά την διάρκεια εκτέλεσης του προσομοιωτή. Συνολικά, για κάθε συσκευή δημιουργήθηκαν 10 σενάρια κίνησης της μέσα στον χώρο με διαφορετικό αριθμό κινήσεων σε κάθε σενάριο και διαφορετική ταχύτητα κίνησης.

6.4 Χρήση του προσομοιωτή και άντληση δεδομένων

Αφού έγινε η επέκταση του προσομοιωτή προχωρήσαμε στην υλοποίηση των δέκα συνολικά σεναρίων κίνησης. Σε αυτά τα σενάρια έχουμε διαφορετικές διαδρομές κινήσεων για κάθε τελική συσκευή. Κάνοντας μια πρώτη εκτέλεση του αλγόριθμου μας σε αυτά τα σενάρια παρατηρήσαμε ότι τα αποτελέσματα δεν ήταν ικανοποιητικά. Τα ποσοστά επιτυχίας αποστολής μηνυμάτων από τις τελικές συσκευές ήταν υπερβολικά χαμηλό αντίθετα με το

ποσοστό που παρήγαγε προηγουμένως το μοντέλο μας. Το ποσοστό αυτό δεν ήταν ένα αποτέλεσμα που μας βρήκε απροετοίμαστους.

Ο λόγος ήταν ότι τα πραγματικά δεδομένα που χρησιμοποιήθηκαν για την εκπαίδευση του μοντέλου δεν αντικατοπτρίζουν τις συνθήκες λειτουργίας του προσομοιωτή. Οι συνθήκες με τις οποίες λειτουργεί ο προσομοιωτής απέχουν αρκετά από ένα εμπορικό δίκτυο σε πραγματικό περιβάλλον. Αφού, η λειτουργία του υφιστάμενου μοντέλου δεν ήταν ικανοποιητική στην λειτουργία του προσομοιωτή προχωρήσαμε στην δημιουργία νέου μοντέλου. Το νέο αυτό μοντέλο θα τροφοδοτείται από δεδομένα τα οποία αντλήθηκαν από τον ίδιο τον προσομοιωτή για να μπορεί το μοντέλο να γενικοποιήσει στον τρόπο λειτουργίας του. Έτσι, δημιουργήσαμε επιπρόσθετα σενάρια τα οποία χρησιμοποιήθηκαν ως σενάρια άντλησης δεδομένων για την εκπαίδευση του μοντέλου μας.

Μέσα από τα σενάρια αυτά δημιουργήσαμε ένα μεγάλο σύνολο δεδομένων το οποίο περιλάμβανε όλα τα δεδομένα εισόδου που χρειάζονται για την εκπαίδευση του μοντέλου. Το σύνολο όμως των δεδομένων που εξαγάγαμε από τον προσομοιωτή περιλαμβάνει όλα τα πακέτα που στάλθηκαν από τις τελικές συσκευές είτε αυτά σταλθήκαν επιτυχώς είτε όχι. Για την εκπαίδευση του όμως από το σύνολο αυτό χρησιμοποιήθηκαν όλα τα δεδομένα των πακέτων που είχαν επιτυχή αποστολή προς την πύλη. Για την εκπαίδευση του μοντέλου ακολουθήσαμε και πάλι την ίδια διαδικασία με την χρήση των αλγορίθμων μηχανικής μάθησης με τα καλύτερα αποτελέσματα αλλά και την δημιουργία νευρωνικού δικτύου με τις παραμέτρους του υφιστάμενου.

Πίνακας 8: Αποτελέσματα εκπαίδευσης με την χρήση δεδομένων του προσομοιωτή

Αλγόριθμος	Ποσοστό Επιτυχίας
KNN	89%
LDA	78%
RFC	91%
Neural Network	91%

Όπως βλέπουμε στον Πίνακα 8 τα αποτελέσματα που κατάφεραν να παρουσιάσουν οι αλγόριθμοι ήταν και πάλι πάρα πολύ καλά. Αξιοσημείωτο ήταν το αποτέλεσμα που παρουσίασε ο αλγόριθμος RFC ο οποίος κατάφερε να έχει ποσοστό επιτυχίας ίσο με το νευρωνικό μας δίκτυο ύψους 91%. Αυτό δείχνει την δυνατότητα αυτών των απλών αλγορίθμων να λύσουν προβλήματα κατηγοριοποίησης με υψηλό ποσοστό επιτυχίας. Αρκετά

καλό αποτέλεσμα παρουσίασε και ο αλγόριθμος KNN με ποσοστό επιτυχίας της τάξεως του 89%. Έχοντας αντλήσει τόσο καλά ποσοστά επιτυχίας από τα μοντέλα μας προχωρήσαμε στην αξιολόγηση του μοντέλου μέσα στον προσομοιωτή.

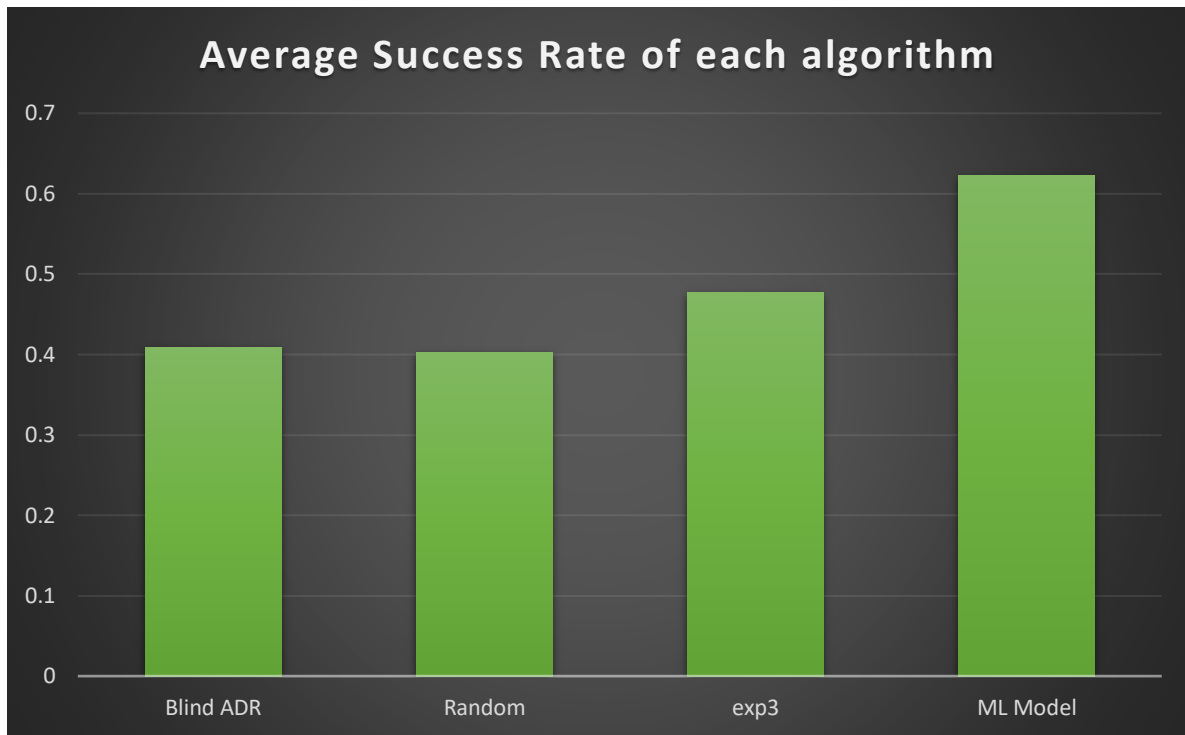
6.5 Αποτελέσματα Αλγορίθμων

Αφού έγινε εκπαίδευση του μοντέλου προχωρήσαμε στην εκτέλεση των 10 σεναρίων για κάθε αλγόριθμο ξεχωριστά για διάφορες παραμέτρους που θα αναλυθούν πιο κάτω στα αποτελέσματα. Για σκοπούς περεταίρω αξιολόγησης δοκιμάστηκαν και οι αλγόριθμοι τυχαίας επιλογής και `expr3` πάνω στα σενάρια μας για να έχουμε ένα πιο ευρύ φάσμα αποτελεσμάτων.

Όπως προαναφέρθηκε οι αλγόριθμοι όλοι έτρεξαν συνολικά πάνω σε 10 σενάρια για διάφορες παραμέτρους. Οι παράμετροι οι οποίοι εναλλάσσονταν ήταν ο αριθμός των τελικών συσκευών (25,60 και 100) και το ποσοστό των συσκευών με κίνηση μέσα στον χώρο (25,50 και 100). Το σύνολο του αριθμού των τελικών συσκευών συμπεριλαμβάνει τις τρεις αυτές τιμές αφού κατά τους δοκιμαστικούς ελέγχους του προσομοιωτή παρατηρήσαμε ότι σε γενικές γραμμές 20 κόμβοι είναι ένας αριθμός με τον οποίο το δίκτυο δεν παρουσιάζει κάποια συμφόρηση, ενώ αντίθετα στους 100 κόμβους το δίκτυο παρουσίαζε πολλές συγκρούσεις μηνυμάτων και μεγάλη συμφόρηση. Ο αριθμός 60 ήταν μια μέση λύση στην οποία το δίκτυο δεν παρουσίαζε ακραίες περιπτώσεις. Με αυτό τον τρόπο θέλαμε να ελέγξουμε την διαχείριση του κάθε αλγορίθμου σε διάφορες συνθήκες και κατά πόσο η λειτουργία του επηρεάζεται από αυτό.

Στην γραφική παράσταση 5 έχουμε τον μέσο όρο επιτυχίας για κάθε αλγόριθμο. Όπως παρατηρούμε το μοντέλο μηχανικής μάθησης που δημιουργήσαμε παρουσιάζει τον καλύτερο μέσο όρο επιτυχίας με ποσοστό γύρω στο 63%, με τον δεύτερο `expr3` να παρουσιάζει ποσοστό επιτυχίας στα 49%. Τέλος έχουμε τους δύο τελευταίους αλγόριθμους `blind ADR` και `random` με ποσοστό επιτυχίας στα 41% και 40% αντίστοιχα.

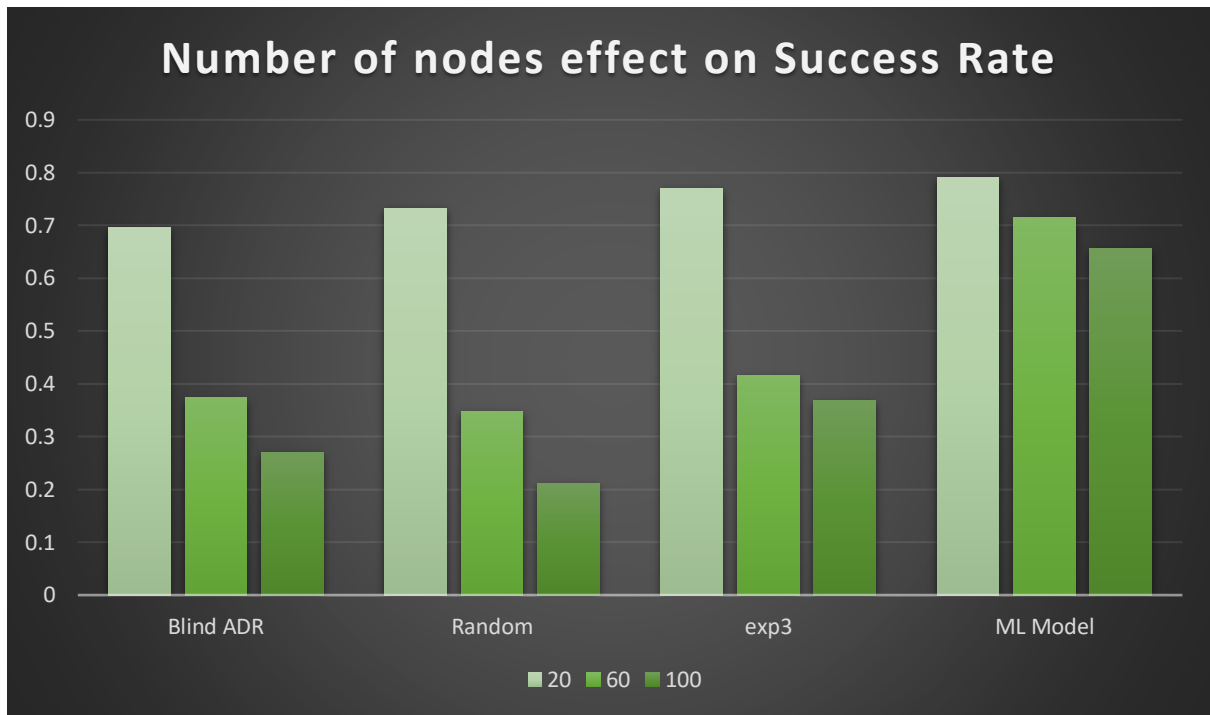
Να σημειωθεί ότι τα ποσοστά αυτά όπως αναφέρθηκε αποτελούν τον μέσο όρο από όλα τα σενάρια και παραμέτρους που εκτελέστηκαν. Κάποια από τα σενάρια αυτά περιέχουν μεγάλο αριθμό από τελικές συσκευές (πχ. 100) όπου σε αυτές τις περιπτώσεις λόγω της μεγάλης συμφόρησης που δημιουργείται στο δίκτυο είχαμε πολλές συγκρούσεις πακέτων στην πύλη.



Γραφική Παράσταση 5 : Μέσος όρος ποσοστού επιτυχίας για κάθε αλγόριθμο

Έπειτα, βλέπουμε την γραφική παράσταση 6 που παρουσιάζει το μέσο ποσοστό επιτυχίας του κάθε αλγόριθμου ανά αριθμό κινητών συσκευών στο χώρο. Ο αριθμός των τελικών συσκευών κινήθηκε στα πλαίσια τριών τιμών: 20, 60 και 100. Στόχος αυτών των σεναρίων ήταν η διερεύνηση της ανταπόκρισης που έχουν οι τελικές συσκευές σε μεταβαλλόμενο αριθμό τελικών συσκευών. Μπορούμε εύκολα να διακρίνουμε ότι το μοντέλο μας μπορεί να είχε κάποια πτώση στο ποσοστό επιτυχίας του με την αύξηση των τελικών συσκευών όμως ήταν μια μείωση εξαιρετικά μικρή σχετικά με την αντιμετώπιση των υπόλοιπων αλγορίθμων.

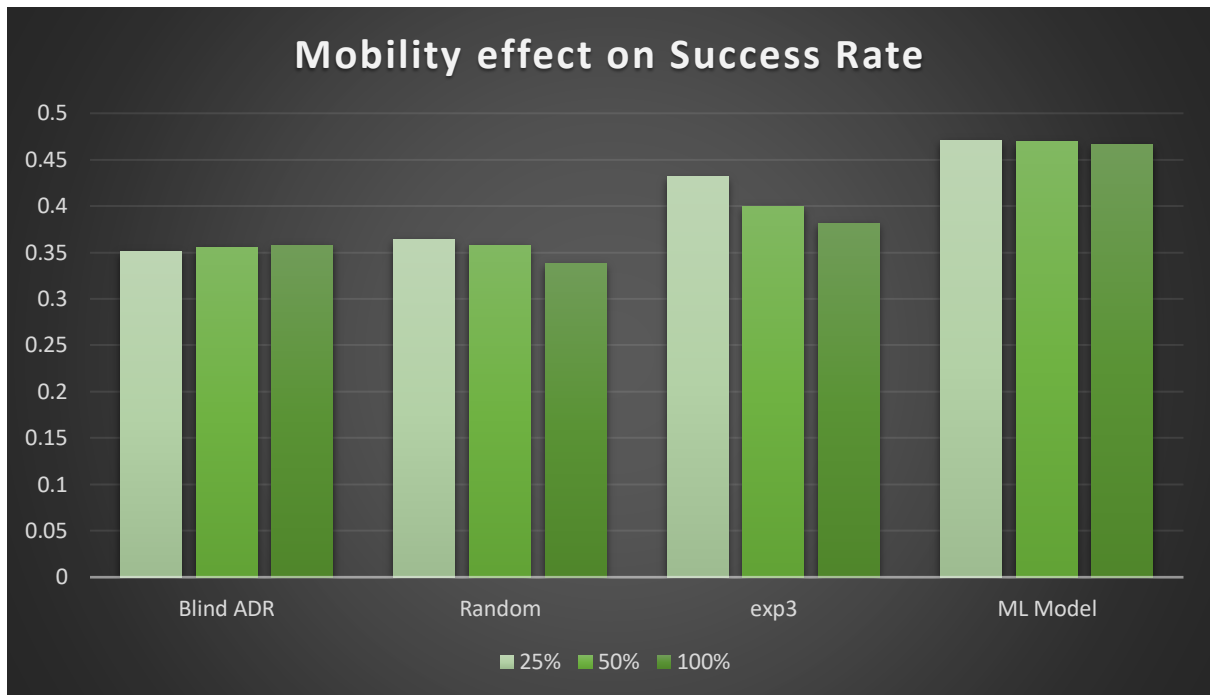
Η μείωση του ποσοστού επιτυχίας ήταν κάτι αναμενόμενο αφού όσο περισσότερες τελικές τόσο περισσότερα μηνύματα αποστέλλονται έτσι έχουμε αύξηση των συγκρούσεων πακέτων στην πύλη. Ωστόσο, οι άλλοι τρεις αλγόριθμοι μετά την μεταφορά από 20 τελικές συσκευές, το ποσοστό επιτυχίας και τον τριών είχε μειωθεί κατά 50% ενώ το μοντέλο μας είχε μια μικρή μείωση της τάξεως του 5%.



Γραφική Παράσταση 6: Ποσοστό επιτυχίας κάθε αλγορίθμου συνάρτηση του αριθμού των τελικών συσκευών στο χώρο.

Στην συνέχεια έχουμε την γραφική παράσταση 7 που παρουσιάζει το μέσο ποσοστό επιτυχίας του κάθε αλγορίθμου ανά ποσοστό κινητών συσκευών στο χώρο. Συγκεκριμένα, τα σενάρια αυτά αποτελούνται συνολικά από 60 τελικές συσκευές εκ των οποίων ένα μέρος ή (25% & 50%,) όλες οι συσκευές κινούνται μέσα στο χώρο. Από αυτά τα αποτελέσματα θέλαμε να παρατηρήσουμε πως κάθε αλγόριθμος ανταποκρίνεται σε μη σταθερές συνθήκες από διαφορετικό αριθμό τελικών συσκευών. Όπως μπορούμε να δούμε, στον ίδιο αριθμό κόμβων όμως με διαφορετικό αριθμό συσκευών να κινούνται το μοντέλο μας κατάφερε να έχει μια σταθερή ανταπόκριση. Το μοντέλο μας μετά από την αύξηση της κινητικότητας δεν επέφερε σχεδόν καμία αλλαγή στα αποτελέσματα.

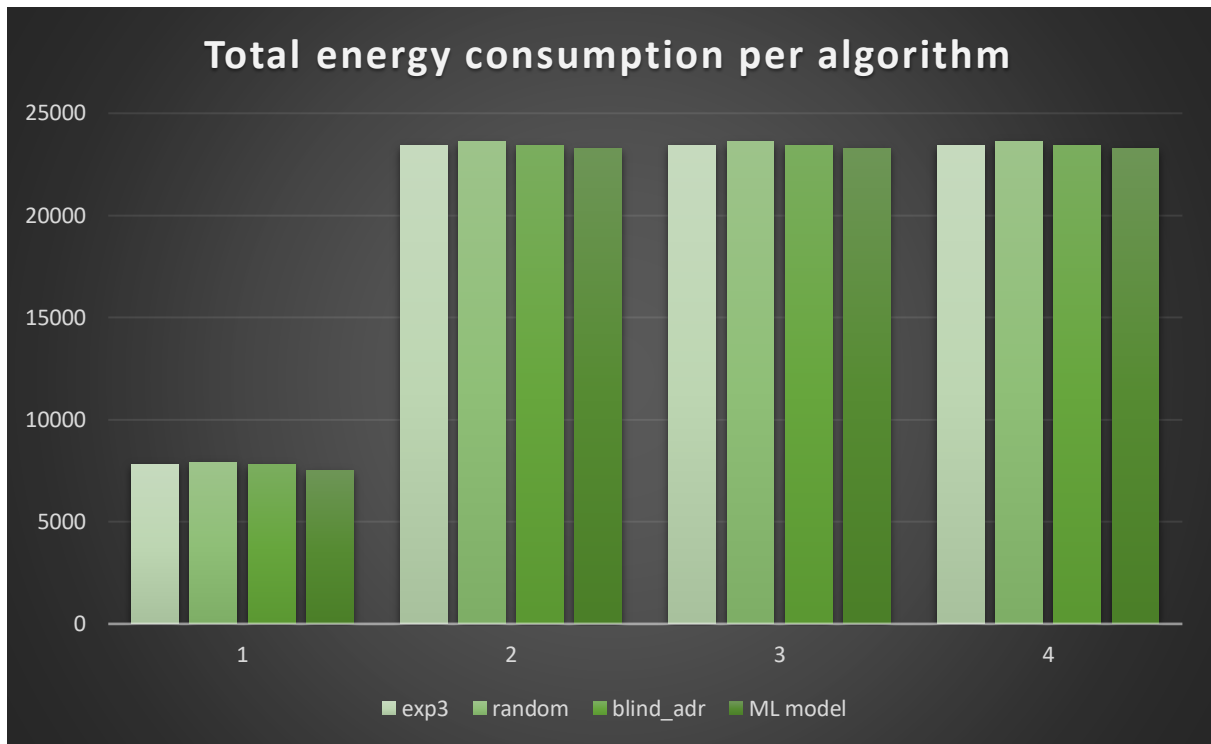
Το ίδιο συμβαίνει και στους αλγόριθμους random και blind ADR κάτι το οποίο ήταν αναμενόμενο λόγω του τρόπου λειτουργίας των δύο αλγορίθμων. Η τυχαία επιλογή πόρων δεν αναμένεται να αλλάξει ποσοστό επιτυχίας αλλά ούτε και η εναλλαγή ανάμεσα σε σταθερές τιμές του blind ADR. Ο exp3 όμως είναι ένας αλγόριθμος ο οποίος παρουσίασε ποσοστό μείωσης ύψους 5% στην πρώτη αύξηση (από 20 σε 60) και 3% στην δεύτερη αύξηση (από 60 σε 100) κάτι που δείχνει την μη σταθερότητα του σε συνθήκες κινητικότητας αντίθετα με το δικό μας μοντέλο.



Γραφική Παράσταση 7: Ποσοστό επιτυχίας κάθε αλγορίθμου συνάρτηση του ποσοστού των κινητών τελικών συσκευών.

Στην γραφική παράσταση 8 έχουμε τα αποτελέσματα του προσομοιωτή για την μέση κατανάλωση ενέργειας κάθε αλγορίθμου σε τέσσερις διαφορετικές περιπτώσεις. Οι διαφορές όπως μπορούμε να δούμε για κάθε αλγόριθμο είναι πολύ μικρές. Ωστόσο αν παρατηρήσουμε τον Πίνακα 8 όπου έχουμε τους αριθμούς για τα αποτελέσματα, βλέπουμε ότι η ενεργειακή κατανάλωση του δικού μας αλγόριθμου βρίσκεται σε πιο χαμηλά επίπεδα ενώ παράλληλα οι υπόλοιποι αλγόριθμοι κινούνται στάδια επίπεδα κατανάλωσης.

Μέσα από αυτή την γραφική παράσταση φαίνεται ότι ο αλγόριθμος μας εκτός από την παραγωγή καλύτερων αποτελεσμάτων σε όλες τις συνθήκες, έχει και την καλύτερη διαχείριση στην κατανάλωση ενέργειας. Η χρήση των βέλτιστων τιμών για το spreading factor οδήγησε το δίκτυο και τις συσκευές να ελαχιστοποιήσουν την κατανάλωση ενέργειας.



Γραφική Παράσταση 8: Μέση κατανάλωση ενέργειας ανά αλγόριθμο.

Πίνακας 8: Μέση κατανάλωση ενέργειας ανά αλγόριθμο.

Μέση κατανάλωση ενέργειας				
exp3	7810.613	23432.37	23419.38	23436.95
random	7911.012	23612.32	23617.23	23632.77
blind_ADR	7799.706	23439.21	23445.53	23445.36
ML model	7510.238	23271.99	23289.48	23301.66

Κεφάλαιο 7

Συμπεράσματα

7.1 Συμπεράσματα	54
7.2 Μελλοντική Δουλειά	56

7.1 Συμπεράσματα

Σε αυτή την εργασία προτείνεται και αναπτύσσεται μια υλοποίηση για διαμόρφωση των παραμέτρων σε δίκτυα LoRa με την χρήση μηχανικής μάθησης στις τελικές συσκευές. Στόχος μας ήταν να παντρέψουμε δύο τεχνολογίες στον τομέα της πληροφορικής οι οποίες είναι και θα είναι από τις σημαντικότερες του τομέα. Η χρήση `tinyML` δεν ήταν κάτι εύκολο καθώς είναι μια τεχνολογία η οποία ακόμη βρίσκεται σε πολύ πρώιμο στάδιο έτσι η κοινότητα και το ερευνητικό πλαίσιο γύρω από αυτό το κομμάτι είναι αρκετά μικρό. Από την άλλη τα δίκτυα LoRa είναι μια ώριμη τεχνολογία η οποία χρησιμοποιείται και σε εμπορικό επίπεδο κάτι που έκανε την εύρεση πληροφοριών εφικτή και με μεγάλη ευκολία.

Η ανάπτυξη ενός τέτοιου μοντέλου και η επιτυχία του ήταν κάτι που δεν ήταν βέβαιο αφού μετά από έρευνα παρατηρήσαμε ότι δεν υπάρχει κάποια αντίστοιχη ερευνητική προσπάθεια. Η αβεβαιότητα αυτή έκανε το εγχείρημα μας ακόμη πιο δύσκολο. Ωστόσο, η σωστή διαχείριση και προγραμματισμός των βημάτων που ακολουθήσαμε ήταν πολύ καθοριστικό για την ολοκλήρωση της διπλωματικής αυτής εργασίας. Πιο συγκεκριμένα, η σταδιακή ανάπτυξη μοντέλων και η αξιολόγηση τους βήμα προς βήμα ήταν κάτι που επιβεβαίωνε την θεωρία μας και την επιτυχία ενός τέτοιου μοντέλου.

Η διαδικασία λοιπόν που ακολουθήσαμε όπως είδαμε και στα προηγούμενα κεφάλαια σε κάθε στάδιο είχε τα επιθυμητά αποτελέσματα. Αρχίζοντας από την απόδειξη λειτουργίας και ανάπτυξης μοντέλων `tinyML` με το μοντέλο κατηγοριοποίησης κίνησης, καταφέραμε να δείξουμε ότι η ανάπτυξη ενός μοντέλου μηχανικής μάθησης σε τελικές συσκευές είναι κάτι εφικτό και μπορεί να λειτουργήσει. Παράλληλα, η διαδικασία αυτή απέδειξε και την επιτυχία της συνύπαρξης των δύο αυτών τεχνολογιών (LoRa & `tinyML`) στην ίδια τελική συσκευή.

Στην συνέχεια είχαμε, την δημιουργία ενός μοντέλου μηχανικής μάθησης σε υψηλό επίπεδο με την χρήση γνωστών αλγορίθμων κατηγοριοποίησης σε ένα ανοικτά διαθέσιμο σύνολο δεδομένων. Το εγχείρημα αυτό αν και δεν είχε τα βέλτιστα αποτελέσματα, το ποσοστό επιτυχίας του 78% δεν ήταν κάτι που ήταν ικανό να δώσει παύση στην συνέχιση της εργασίας. Κάνοντας, όλες τις απαραίτητες διαφοροποιήσεις και προσθήκες στα δεδομένα μας οδηγηθήκαμε στην δημιουργία ενός συνόλου δεδομένων που ανταποκρίνεται στις απαιτήσεις μας το οποίο είχε τα αναμενόμενα αποτελέσματα με ποσοστό επιτυχίας 97%. Καταφέροντας, λοιπόν να παρουσιάσουμε ένα τόσο υψηλό ποσοστό επιτυχίας επόμενο βήμα ήταν να δούμε πως ανταποκρίνεται ένα τέτοιο μοντέλο σε σενάρια λειτουργίας ενός δικτύου LoRa.

Η αξιολόγηση του μοντέλου λοιπόν ήταν το τελευταίο βήμα για την ολοκλήρωση της διπλωματικής εργασίας. Η αξιολόγηση μπορεί να γίνει σε γενικά πλαίσια με δύο τρόπους την αξιολόγηση σε πραγματικό δίκτυο ή με την χρήση κάποιου προσομοιωτή. Προχωρήσαμε με την επιλογή του προσομοιωτή αφού ήταν μια πιο διαχειρίσιμη λύση μέσα στο χρονικό πλαίσιο που είχαμε. Η επιλογή και επέκταση του προσομοιωτή LoRa-MAB για την αξιολόγηση μας ήταν μια άλλη διαδικασία η οποία αποδείχτηκε πιο πολύπλοκη και χρονοβόρα. Ωστόσο, με την χρήση του προσομοιωτή καταφέραμε να αποδείξουμε την υπεροχή της πρότασης απέναντι σε άλλους αλγόριθμους. Μέσα από τα αποτελέσματα αυτά δείξαμε την καλύτερη διαχείριση της πρότασης μας σε καταστάσεις μη σταθερών συνθηκών ραδιοκυμάτων όπως η κινητικότητα. Τα αποτελέσματα απέδειξαν ότι το μοντέλο μας εκτός από γενικά καλύτερα αποτελέσματα επιτυχίας καταφέρνει να διαχειριστεί καλύτερα καταστάσεις όπου έχουμε μεγάλο αριθμό τελικών συσκευών με η και χωρίς κινητικότητα λόγω της γενικοποίησης των καταστάσεων που μπορεί να παράξει ένα μοντέλο μηχανικής μάθησης.

Καταλήγοντας, η πρόταση μας αποδείχτηκε τόσο ότι είναι εφικτή αλλά και ότι μπορεί να παράγει καλύτερα αποτελέσματα από υφιστάμενες λύσεις που έχουν προταθεί και βρίσκονται υπό λειτουργία στην παρούσα φάση. Με αυτή την δουλειά καταφέραμε να δείξουμε μία μέθοδο επίλυσης 2 εκ των προκλήσεων που αντιμετωπίζουν το διαδίκτυο των πραγμάτων, την βέλτιστη χρήση του εύρους ζώνης και του ρυθμού μεταφοράς δεδομένων. Με την βελτιστοποίηση των τιμών αυτών των δύο παραμέτρων καταφέραμε να έχουμε καλύτερη διαχείριση στην κατανάλωση ενέργειας από τις τελικές συσκευές. Ωστόσο, η δουλειά για την εργασία αυτή δεν έχει ολοκληρωθεί όπως θα δούμε πιο κάτω με περισσότερη λεπτομέρεια.

7.2 Μελλοντική Δουλειά

Το ερευνητικό κομμάτι για την εργασία αυτή δεν έχει ολοκληρωθεί πλήρως αφού έχουμε μόλις αποδείξει την επιτυχία ενός μοντέλου μηχανικής μάθησης σε ένα πιο υψηλό επίπεδο. Στόχος μας είναι έχοντας όλα τα παραπάνω ως βάση να προχωρήσουμε στην υλοποίηση και αξιολόγηση του αλγόριθμου σε πραγματικό δίκτυο. Η διαδικασία υλοποίησης του μοντέλου σε πραγματικά δεδομένα είναι μία αρκετά χρονοβόρα διαδικασία. Η άντληση του κατάλληλου αριθμού δεδομένων για την εκπαίδευση ενός τέτοιου μοντέλου μπορεί να διαρκέσει για μεγάλο χρονικό διάστημα ανάλογα με το μέγεθος του δικτύου. Επιπρόσθετα, ο αριθμός των συσκευών που μπορούν να υποστηρίξουν τόσο την βιβλιοθήκη tensorflow lite αλλά και την αναπαραγωγή πακέτων LoRa δεν είναι μεγάλος κάτι που επιβαρύνει την διαδικασία.

Έτσι η συνέχεια της εργασίας είναι μια διαδικασία η οποία είναι εξίσου σημαντική όμως παράλληλα χρειάζεται εξίσου αρκετή προσπάθεια και έρευνα για να καταφέρουμε να δημιουργήσουμε ένα μοντέλο το οποίο να μπορεί να ανταποκριθεί στις απαιτήσεις ενός πραγματικού δικτύου. Επιπλέον, τα περιθώρια αποτυχίας ή μη ολοκληρωμένης εκπαίδευσης ενός μοντέλου μπορεί να επιφέρει γενική συμφόρηση στο δίκτυο και σε συσκευές που δεν συμμετέχουν στις δοκιμασίες. Έτσι η μεταφορά του θεωρητικού πλαισίου από ένα ελεγχόμενο περιβάλλον σε ένα μη ελεγχόμενο όπως ένας εμπορικό δίκτυο απαιτεί πιο λεπτή διαχείριση για την ελαχιστοποίηση του περιθωρίου σφάλματος.

Βιβλιογραφία

- [1] P. J. Basford, F. M. Bulot, M. Apetroaie-Cristea, S. J. Cox, and S. J. Ossont, “LoRaWAN for Smart City IOT deployments: A long term evaluation,” *Sensors*, vol. 20, no. 3, p. 648, 2020.
- [2] L. Bhatia, M. Breza, R. Marfievici, and J. A. McCann, “Loed: The LoRaWAN at the edge dataset,” *Proceedings of the Third Workshop on Data: Acquisition To Analysis*, 2020
- [3] J. Jongboom, “Adding machine learning to your LoRaWAN® device,” *The Things Network*. [Online]. Available: <https://www.thethingsnetwork.org/article/adding-machine-learning-to-your-LoRaWAN-device>. [Accessed: 10-May-2022].
- [4] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne, “Understanding the limits of LoRaWAN,” *IEEE Communications Magazine*, vol. 55, no. 9, pp. 34–40, 2017..
- [5] F. Cuomo, M. Campo, A. Caponi, G. Bianchi, G. Rossini, and P. Pisani, “ExpLoRa: Extending the performance of LoRa by suitable spreading factor allocations,” *2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2017.
- [6] F. Van den Abeele, J. Haxhibeqiri, I. Moerman, and J. Hoebeke, “Scalability analysis of large-scale LORAWAN networks in NS-3,” *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 2186–2198, 2017.
- [7] K. Q. Abdelfadeel, D. Zorbas, V. Cionca, and D. Pesch, “FREE —fine-grained scheduling for reliable and energy-efficient data collection in LoRaWAN,” *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 669–683, 2020.
- [8] D.-T. Ta, K. Khawam, S. Lahoud, C. Adjih, and S. Martin, “Lora-MAB: A flexible simulator for decentralized learning resource allocation in IOT networks,” *2019 12th IFIP Wireless and Mobile Networking Conference (WMNC)*, 2019.

- [9] M. Alenezi, K. K. Chai, A. S. Alam, Y. Chen, and S. Jimaa, "Unsupervised learning clustering and dynamic transmission scheduling for efficient dense LoRaWAN Networks," *IEEE Access*, vol. 8, pp. 191495–191509, 2020.
- [10] M. Capuzzo, D. Magrin, and A. Zanella, "Confirmed traffic in LoRaWAN: Pitfalls and countermeasures," *2018 17th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, 2018.
- [11] P. Warden and D. Situnayake, *TinyML: Machine learning with tensorflow lite on Arduino and ultra-low-power Microcontrollers*. Sebastopol, California: O'Reilly, 2020.
- [12] R. Kufakunesu, G. P. Hancke, and A. M. Abu-Mahfouz, "A survey on adaptive data rate optimization in LoRaWAN: Recent solutions and major challenges," *Sensors*, vol. 20, no. 18, p. 5044, 2020.
- [13] R. M. Sandoval, A.-J. Garcia-Sanchez, and J. Garcia-Haro, "Optimizing and updating LoRa communication parameters: A machine learning approach," *IEEE Transactions on Network and Service Management*, vol. 16, no. 3, pp. 884–895, 2019.
- [14] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low power wide area networks: An overview," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 2, pp. 855–873, 2017.
- [15] L. Bhatia, M. Breza, R. Marfievici, and J. A. McCann, "Loed: The LoRaWAN at the edge dataset," *Proceedings of the Third Workshop on Data: Acquisition To Analysis*, 2020.
- [16] V. M. Suresh, R. Sidhu, P. Karkare, A. Patil, Z. Lei, and A. Basu, "Powering the IOT through embedded machine learning and LoRa," *2018 IEEE 4th World Forum on Internet of Things (WF-IoT)*, 2018.
- [17] W. Ayoub, A. E. Samhat, F. Nouvel, M. Mroue, and J.-C. Prevotet, "Internet of mobile things: Overview of LoRaWAN, DASH7, and Nb-IOT in LPWANS standards and supported mobility," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1561–1581, 2019.

- [18] V. Gazis, M. Goertz, M. Huber, A. Leonardi, K. Mathioudakis, A. Wiesmaier and F. Zeigerh, "Short Paper: IoT: Challenges, projects, architectures," 18th International Conference on Intelligence in Next Generation Networks (ICIN), 17-19 February 2015
- [19] W. Shyr, C. Lin and H. Feng, "Development of Energy Management System Based on Internet of Things Technique," International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering vol:11, no:3, 2017