



Πανεπιστήμιο Πειραιώς
Σχολή Τεχνολογιών Πληροφορικής και Επικοινωνιών
Τμήμα Πληροφορικής

Πτυχιακή Εργασία

Τίτλος	Εφαρμογές Μηχανικής Μάθησης σε Ηλεκτρικά Οχήματα Machine Learning Applications in Electric Vehicles
Όνοματεπώνυμο Φοιτητή	Νικόλαος Ανδριανόπουλος
Πατρώνυμο	Χαράλαμπος
Αριθμός Μητρώου	Π20018
Επιβλέπων	Ιωάννης Θεοδωρίδης, Καθηγητής

Ημερομηνία Παράδοσης: Σεπτέμβριος 2024

Copyright ©

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν αποκλειστικά τον συγγραφέα και δεν αντιπροσωπεύουν τις επίσημες θέσεις του Πανεπιστημίου Πειραιώς.

Ως συγγραφέας της παρούσας εργασίας δηλώνω πως η παρούσα εργασία δεν αποτελεί προϊόν λογοκλοπής και δεν περιέχει υλικό από μη αναφερόμενες πηγές.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κ. Ιωάννη Θεοδωρίδη, Καθηγητή του Τμήματος Πληροφορικής του Πανεπιστημίου Πειραιώς για την εμπιστοσύνη που μου έδειξε στην εκπόνηση αυτής της πτυχιακής εργασίας.

Επίσης, θα ήθελα να ευχαριστήσω την Δρα. Ευαγγελία Χονδροδήμα και τον υποψήφιο διδάκτορα κ. Ιωάννη Κοντούλη, μέλη του Εργαστηρίου Επιστήμης Δεδομένων του Πανεπιστημίου Πειραιώς, για τη βοήθεια και στήριξη που μου παρείχαν στην εκπόνηση της παρούσας εργασίας.

Τέλος, είμαι ευγνώμων προς όλους όσους με ενθάρρυναν και με στήριξαν καθόλη τη διάρκεια των σπουδών μου και κυρίως την οικογένειά μου, που ήταν και είναι πάντα στο πλευρό μου.

Περίληψη

Η παρούσα εργασία επικεντρώνεται στον τομέα της βαθιάς μάθησης. Πιο συγκεκριμένα, στοχεύει στην αντιμετώπιση του προβλήματος της ακριβούς πρόβλεψης της ενεργειακής ζήτησης των φορτιστών ενός δικτύου φόρτισης ηλεκτρικών οχημάτων. Αρχικά, προτείνεται μια μεθοδολογία επίλυσης του συγκεκριμένου προβλήματος. Η μεθοδολογία αυτή προτείνει τη χρήση τεχνολογιών βαθιάς μάθησης και στόχος της είναι η εξέταση, αξιολόγηση και σύγκριση τριών μοντέλων που βασίζονται στη χρήση συνελκτικών νευρωνικών δικτύων για τη βραχυπρόθεσμη πρόβλεψη της ενεργειακής ζήτησης. Η εφαρμογή αυτής της μεθοδολογίας γίνεται με τη χρήση ενός συνόλου δεδομένων ανοικτής πρόσβασης που περιέχει πληροφορίες για ένα δίκτυο φορτιστών εντός της πόλης του Πάλο Άλτο στην Καλιφόρνια. Μετά τη δημιουργία και την εκπαίδευσή τους, τα μοντέλα αξιολογήθηκαν χρησιμοποιώντας το RMSE στις προβλέψεις του συνολικού συστήματος. Με βάση το χαμηλότερο RMSE, προκύπτει ότι το T-GCN μοντέλο είναι το ιδανικό μοντέλο για τις προβλέψεις ενεργειακής ζήτησης.

Λέξεις Κλειδιά: ενεργειακή ζήτηση, μηχανική μάθηση, πρόβλεψη

Abstract

This thesis focuses on the field of deep learning. More specifically, it aims to address the issue of accurately predicting the energy demand of chargers in an electric vehicle (EV) charging network. Initially, a methodology is proposed in order to solve this particular problem. The methodology suggests the use of deep learning technologies with the ultimate goal of examining, evaluating and comparing three models based on convolutional neural networks for short-term energy demand prediction. The application of this methodology utilizes an open-source dataset containing data about a charger network in the city of Palo Alto, California. After their creation and training, the models were evaluated using RMSE on the forecast for the total system. Based on the lowest RMSE scores, it was conducted that the T-GCN model is the ideal model for energy demand predictions.

Keywords: energy demand, machine learning, prediction

Πίνακας Εικόνων και Διαγραμμάτων

Εικόνες

Εικόνα 1: Μη κατευθυνόμενος γράφος G	2
Εικόνα 2: Πίνακας Γειτνίασης Γράφου G	3
Εικόνα 3: Βασική Αρχιτεκτονική μοντέλου βαθιάς μάθησης	6
Εικόνα 4: Αρχιτεκτονική CNN	7
Εικόνα 5: Αρχιτεκτονική LSTM	8
Εικόνα 6: Αρχιτεκτονική υβριδικού CNN+LSTM	8
Εικόνα 7: Αρχιτεκτονική GCN	9
Εικόνα 8: Αρχιτεκτονική T-GCN	10
Εικόνα 9: Το Δίκτυο Φορτιστών του Palo Alto	24
Εικόνα 10: Αρχικό Σύνολο Δεδομένων	25
Εικόνα 11: Χαρακτηριστικά που θα χρησιμοποιηθούν στην έρευνα	25
Εικόνα 12: Raster Map	26
Εικόνα 13: Ενημερωμένο Σύνολο Δεδομένων	26
Εικόνα 14: Αναπαράσταση του γράφου G στο δίκτυο	28

Διαγράμματα

Διάγραμμα 4.1: Αποτελέσματα για τις τελευταίες 7 ημέρες δεδομένων	30
Διάγραμμα 4.2: Αποτελέσματα για τις τελευταίες 30 ημέρες δεδομένων	31

Πίνακες

Πίνακας 4.1: RMSE προβλέψεων για το συνολικό σύστημα	29
Πίνακας 4.2: RMSE προβλέψεων για κάθε σταθμό	29
Πίνακας 4.3: Σύγκριση RMSE επιδόσεων των 2 μοντέλων T-GCN	31

Πίνακας Περιεχομένων

Copyright ©	ii
Ευχαριστίες	iii
Περίληψη	iv
Abstract	iv
Πίνακας Εικόνων και Διαγραμμάτων	v
Εικόνες	v
Διαγράμματα	v
Πίνακες	v
Εισαγωγή	1
1. Θεωρητικό Υπόβαθρο	2
1.1 Γράφοι	2
1.2 Πίνακας Γειτνίασης	3
1.3 Raster Map	3
1.4 Η φόρμουλα Haversine	4
1.5 Τεχνητή Νοημοσύνη	4
1.5 Μηχανική Μάθηση	5
1.6 Βαθιά Μάθηση	5
1.7 Επιβλεπόμενη μάθηση	6
1.8 Μοντέλα Μηχανικής Μάθησης	6
1.8.1 Convolutional Neural Network - CNN	6
1.8.2 Long Short-Term Memory - LSTM	7
1.8.3 Convolutional LSTM Hybrid	8
1.8.4 Graph Convolutional Network - GCN	9
1.8.5 Temporal Graph Convolutional Network - T-GCN	9
2. Σχετικές Εργασίες	11
2.1 Πρόβλεψη ημερήσιας απαιτούμενης ενέργειας με Deep Neural Networks	11
2.2 Πρόβλεψη μηνιαίας απαιτούμενης ενέργειας με την μεθοδολογία GEORGE	11
2.3 Πρόβλεψη απαιτούμενης ενέργειας με την χρήση Χρονοσειρών	12
2.4 Πρόβλεψη διανομής σταθμών φόρτισης ηλεκτρικών οχημάτων με χρήση μηχανικής μάθησης	12
2.5 Πρόβλεψη ζητούμενης ενέργειας με βάση την αρχιτεκτονική Transformer	13
Εφαρμογές Μηχανικής Μάθησης σε Ηλεκτρικά Οχήματα	vi

2.6 Πρόβλεψη ενεργειακής ζήτησης ηλεκτρικών οχημάτων με Υπολογιστικά Μοντέλα	13
3. Μεθοδολογία	15
3.1 Συλλογή Δεδομένων	15
3.2 Επεξεργασία Δεδομένων	15
3.3 Κανονικοποίηση Δεδομένων	15
3.4 Διάσπαση Συνόλου Δεδομένων Για Τα Μοντέλα CNN και Convolutional LSTM	15
3.4.1 Η Συνάρτηση findcell	16
3.4.2 Η Συνάρτηση getcell	16
3.4.3 Η Συνάρτηση build_rastermap	17
3.4.4 Η Συνάρτηση traintest	18
3.5 Διάσπαση Συνόλου Δεδομένων Για Το Μοντέλο T-GCN	19
3.5.1 Η Συνάρτηση distance	19
3.5.2 Η Συνάρτηση adj_norm	20
3.5.3 Η Συνάρτηση getGraph	20
3.5.4 Η Συνάρτηση setupTGCN	21
3.6 Εκπαίδευση και Αξιολόγηση των Μοντέλων	22
3.6.1 Τα Μοντέλα προς Εκπαίδευση	22
3.6.2 Αξιολόγηση με την Μετρική RMSE	23
4. Εφαρμογή	24
4.1 Σύνολο Δεδομένων	24
4.2 Προεπεξεργασία Δεδομένων για τα Μοντέλα CNN και ConvlSTM	25
4.2.1 Εφαρμογή Raster Map	26
4.2.2 Διαχωρισμός σε Σύνολα Εκπαίδευσης και Δοκιμής των Μοντέλων CNN και ConvlSTM	27
4.3 Προεπεξεργασία Δεδομένων για το Μοντέλο T-GCN	27
4.3.1 Δημιουργία Γράφου και Πίνακα Γεινιάσης	27
4.3.2 Διαχωρισμός σε Σύνολα Εκπαίδευσης και Δοκιμής του Μοντέλου T-GCN	27
4.4 Εκπαίδευση των Μοντέλων	28
4.5 Αξιολόγηση των Μοντέλων	29
4.6 Αποτελέσματα	29
4.7 Σύγκριση με αντίστοιχη έρευνα	31
5. Συμπεράσματα - Μελλοντικές Εργασίες	33

5.1 Συμπεράσματα	33
5.2 Μελλοντικές Εργασίες	33
Πίνακας Ορολογιών	34
Πίνακας Συντμήσεων - Αρκτικόλεξων - Ακρωνύμιων	37
Βιβλιογραφία	39

Εισαγωγή

Ένα σημαντικό μέρος της καθημερινότητας των ανθρώπων, το οποίο παραμένει σταθερό καθ' όλη σχεδόν τη διάρκεια της ζωής τους, αφορά στις μεταφορές και, γενικά, στις μετακινήσεις στο εξωτερικό περιβάλλον. Από τη χρήση ζωήλατων οχημάτων, μέχρι τις αρχές του 20ου αιώνα, στα μέσα μαζικής μεταφοράς, τα πρώτα συμβατικά αυτοκίνητα και, πλέον, στη χρήση ηλεκτρικών οχημάτων, ο άνθρωπος ανέκαθεν είχε τον τρόπο να μετακινηθεί εκτός του σπιτιού του.

Σήμερα, εν έτει 2024, η ραγδαία αύξηση των ηλεκτρικών οχημάτων που κυκλοφορούν στους δρόμους έχει συμβάλει σε σημαντικές αλλαγές στον ενεργειακό τομέα, καθώς και στον τομέα των μετακινήσεων, καθιστώντας αναγκαία την πρόβλεψη της ενεργειακής ζήτησης με τη μεγαλύτερη δυνατή ακρίβεια. Η υιοθέτηση της ηλεκτροκίνησης από ένα μεγάλο και συνεχώς αυξανόμενο ποσοστό του πληθυσμού του πλανήτη έχει δημιουργήσει την ανάγκη ύπαρξης αξιόπιστων μεθόδων πρόβλεψης της ενεργειακής ζήτησης, ειδικά στα δίκτυα σταθμών φόρτισης ηλεκτρικών οχημάτων.

Η παρούσα εργασία, εστιάζει στην πρόβλεψη της ζητούμενης ενέργειας για ένα δίκτυο σταθμών φόρτισης ηλεκτρικών οχημάτων, χρησιμοποιώντας σύγχρονες μεθόδους και μοντέλα πρόβλεψης. Πιο συγκεκριμένα, αυτή η έρευνα στοχεύει στην ανάπτυξη μιας προσέγγισης για την πρόβλεψη της ενεργειακής ζήτησης χρησιμοποιώντας αλγόριθμους μηχανικής μάθησης (Machine Learning) και βαθιάς μάθησης (Deep Learning). Σκοπός είναι η δημιουργία χρήσιμων εργαλείων, που θα επιτρέψουν τη βέλτιστη διαχείριση των σχετικών με την ενέργεια ζητημάτων για αυτούς τους σταθμούς φόρτισης, λαμβάνοντας υπόψη παράγοντες όπως, ενδεικτικά, τις γεωγραφικές συντεταγμένες των σταθμών φόρτισης.

1. Θεωρητικό Υπόβαθρο

Στο συγκεκριμένο κεφάλαιο παρουσιάζεται η θεωρία στην οποία βασίζεται η συγκεκριμένη εργασία, ούτως ώστε να επιτευχθεί ο τελικός στόχος, ήτοι η πρόβλεψη της ενεργειακής ζήτησης.

Συγκεκριμένα, καταγράφονται οι ερμηνείες βασικών εννοιών, η κατανόηση των οποίων αποτελεί αναγκαία προϋπόθεση για την εξοικείωση του αναγνώστη με το αντικείμενο και την παρακολούθηση του κειμένου.

1.1 Γράφοι

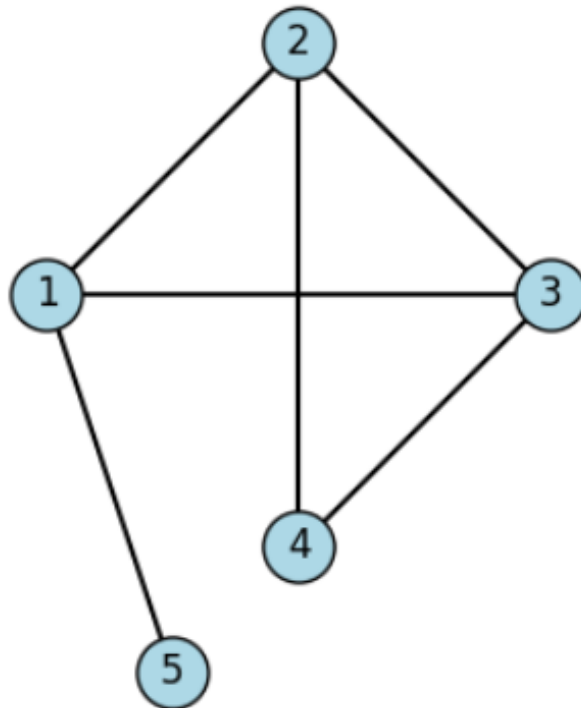
Ένας γράφος (Graph) είναι μια μαθηματική δομή που χρησιμοποιείται για την αναπαράσταση σχέσεων και συνδέσεων μεταξύ οντοτήτων. Η δομή ενός γράφου αποτελείται από δύο κύρια στοιχεία:

- **Κόμβοι (Nodes):** Οι κόμβοι αντιπροσωπεύουν τις οντότητες ή τα σημεία ενός γράφου. Για παράδειγμα, σε ένα κοινωνικό δίκτυο κάθε κόμβος αναπαριστά έναν χρήστη.
- **Ακμές (Edges):** Οι ακμές είναι οι συνδέσεις που συνδέουν δύο κόμβους μεταξύ τους. Μπορούν να έχουν κατεύθυνση (συμμετρικές) σε έναν κατευθυνόμενο γράφο ή να είναι απλές συνδέσεις χωρίς κατεύθυνση (ασύμμετρες) σε έναν μη κατευθυνόμενο γράφο.

Οι γράφοι μπορούν να είναι είτε **κατευθυνόμενοι**, όπου οι ακμές έχουν κατεύθυνση από έναν κόμβο σε έναν άλλον, είτε **μη κατευθυνόμενοι**, όπου οι ακμές δεν έχουν κατεύθυνση και η σύνδεση είναι αμοιβαία. Επίσης, οι ακμές μπορούν να έχουν βάρος που αναπαριστά την απόσταση μεταξύ των κόμβων του γράφου.

Έστω τοπολογία γράφου $G = (V, E)$ όπου $V = \{1, 2, 3, 4, 5\}$ και $E = \{(1, 2), (2, 3), (3, 4), (5, 1), (1, 3), (2, 4)\}$.

Παρακάτω φαίνεται ο σχεδιασμός αυτού του μη κατευθυνόμενου γράφου.



Εικόνα 1 | Μη κατευθυνόμενος γράφος G

1.2 Πίνακας Γειτνίασης

Ο πίνακας γειτνίασης (adjacency matrix) είναι μια μαθηματική αναπαράσταση ενός γράφου που χρησιμοποιείται ευρέως για την ανάλυση και την επεξεργασία γραφημάτων. Ο πίνακας γειτνίασης A ενός γράφου G με N κόμβους είναι ένα πίνακας $N \times N$, όπου το κάθε στοιχείο του πίνακα A_{ij} υποδεικνύει την παρουσία ή όχι μιας ακμής μεταξύ των κόμβων i και j . Οι τιμές του πίνακα εξαρτώνται από τον τύπο του γράφου που καλείται να αναπαραστήσει ο πίνακας γειτνίασης. Συγκεκριμένα:

- **Μη Κατευθυνόμενος Γράφος:** Στον μη κατευθυνόμενο γράφο, ο πίνακας γειτνίασης είναι συμμετρικός. Δηλαδή ισχύει $A_{ij} = A_{ji}$. Εάν υπάρχει ακμή που να συνδέει τους κόμβους, τότε ισχύει $A_{ij} = 1$, αλλιώς παίρνει την τιμή 0.
- **Κατευθυνόμενος Γράφος:** Στον κατευθυνόμενο γράφο, ο πίνακας γειτνίασης δεν είναι απαραίτητα συμμετρικός. Εάν υπάρχει ακμή που να συνδέει τον κόμβο i με τον κόμβο j , τότε ορίζεται $A_{ij} = 1$. Αντίστροφα, εάν υπάρχει ακμή που να συνδέει τον κόμβο j με τον κόμβο i , τότε ορίζεται $A_{ij} = 1$, χωρίς απαραίτητα να είναι ίσο με το A_{ij} .

Παρακάτω, φαίνεται ως παράδειγμα ο πίνακας γειτνίασης του γράφου G που προβάλλεται στην Εικόνα 2.

	1	2	3	4	5
1	0	1	1	0	0
2	0	0	1	1	0
3	0	0	0	1	0
4	0	0	0	0	0
5	1	0	0	0	0

Εικόνα 2 | Πίνακας Γειτνίασης Γράφου G

1.3 Raster Map

Οι χάρτες raster αποτελούν μια σημαντική τεχνολογία στον τομέα της γεωγραφικών πληροφοριακών συστημάτων, η οποία επιτρέπει την απεικόνιση γεωγραφικών δεδομένων με τη μορφή πλέγματος (grid). Κάθε κελί του πλέγματος αντιπροσωπεύει μια συγκεκριμένη γεωγραφική περιοχή και περιέχει μια τιμή αντίστοιχη με μια ιδιότητα-χαρακτηριστικό εκείνης της περιοχής (π.χ. υψόμετρο, θερμοκρασία).

Η διαδικασία δημιουργίας ενός τέτοιου χάρτη περιλαμβάνει συγκεκριμένα στάδια. Αρχικά, γίνεται η συλλογή των δεδομένων που πρόκειται να χρησιμοποιηθούν για τη δημιουργία του χάρτη. Τα δεδομένα αυτά προέρχονται από διάφορες πηγές, όπως δορυφορικές εικόνες ή -όπως στη συγκεκριμένη εργασία- επίγειες μετρήσεις. Συγκεκριμένα, περνούν από το στάδιο επεξεργασίας και μετατρέπονται στην κατάλληλη μορφή για να δημιουργηθεί το raster.

Αφού ολοκληρωθεί η μετατροπή των δεδομένων, ακολουθεί η δημιουργία του πλέγματος. Το πλέγμα, αποτελείται από έναν προκαθορισμένο αριθμό κελιών, καθένα από τα οποία έχει συγκεκριμένο μέγεθος. Το μέγεθος του κάθε κελιού εξαρτάται από την ακρίβεια που απαιτείται για την εφαρμογή του και, κατ' επέκταση, από το είδος του προβλήματος. Για παράδειγμα, ένας χάρτης που χρησιμοποιείται για την ανάλυση κλιματικών αλλαγών ενδεχομένως να χρειάζεται μεγαλύτερα κελιά από έναν χάρτη που χρησιμοποιείται για τον

πολεοδομικό σχεδιασμό μιας περιοχής [15], περίπτωση κατά την οποία απαιτούνται μικρότερα κελιά για την επίτευξη μεγαλύτερης ακρίβειας.

Μετά τη δημιουργία του πλέγματος, σειρά έχει η ανάθεση τιμών σε κάθε κελί. Η διαδικασία αυτή γίνεται με διάφορους τρόπους που εξαρτώνται από τη φύση των δεδομένων και τις απαιτήσεις της εφαρμογής.

Μετά τη δημιουργία του πλέγματος, τελική σειρά έχει η ανάθεση τιμών σε κάθε κελί του. Η διαδικασία αυτή γίνεται με διάφορους τρόπους, που εξαρτώνται από τη φύση των δεδομένων και τις απαιτήσεις της εκάστοτε εφαρμογής.

1.4 Η φόρμουλα Haversine

Η φόρμουλα Haversine (Haversine formula) είναι ένας μαθηματικός τύπος που χρησιμοποιείται για τον υπολογισμό της απόστασης μεταξύ δύο σημείων πάνω στην επιφάνεια μιας σφαίρας [17]. Η απόσταση αυτή είναι γνωστή ως “μεγάλη κυκλική απόσταση” και είναι η μικρότερη απόσταση μεταξύ δύο σημείων πάνω στη σφαίρα.

Ο τύπος αυτός βασίζεται σε τριγωνομετρικές συναρτήσεις και χρησιμοποιεί τις γεωγραφικές συντεταγμένες (γεωγραφικό μήκος και πλάτος) των 2 σημείων που θέλουμε να υπολογίσουμε την μεταξύ τους απόσταση. Τα σημεία αυτά εκφράζονται σε μοίρες και για την εφαρμογή του τύπου πρέπει να μετατραπούν σε ακτίνια. Ο τύπος υπολογίζει την απόσταση με βάση τη διαφορά στο γεωγραφικό μήκος και πλάτος των 2 σημείων και χρησιμοποιεί την τιμή της γωνίας για να βρει τη μεταξύ τους απόσταση. Ο τύπος έχει ως εξής:

$$a = \sin^2\left(\frac{\Delta\varphi}{2}\right) + \cos(\varphi_1) * \cos(\varphi_2) * \sin^2\left(\frac{\Delta\lambda}{2}\right), (1.1)$$

$$c = 2 * \operatorname{atan2}(\sqrt{a}, \sqrt{1-a}), (1.2)$$

$$d = R * c \quad (1.3)$$

όπου:

- φ_1, φ_2 , τα γεωγραφικά πλάτη των 2 σημείων σε ακτίνια
- λ_1, λ_2 , τα γεωγραφικά μήκη των 2 σημείων σε ακτίνια
- $\Delta\varphi = \varphi_2 - \varphi_1$ και $\Delta\lambda = \lambda_2 - \lambda_1$, οι διαφορές γεωγραφικού πλάτους και μήκους, αντίστοιχα
- R , η ακτίνα της Γης σε χιλιόμετρα που κυμαίνεται στο διάστημα [6.357, 6.378] km.

Ο συγκεκριμένος τύπος, δίνει τη χιλιομετρική απόσταση μεταξύ δύο σημείων. Χρησιμοποιείται συχνά σε εφαρμογές όπως το παγκόσμιο σύστημα εντοπισμού (Global Positioning System - GPS) και άλλα γεωγραφικά πληροφοριακά συστήματα (Geographic Information System - GIS).

1.5 Τεχνητή Νοημοσύνη

Όπως προαναφέρθηκε, ο τομέας στον οποίο εντάσσεται το πρόβλημα της συγκεκριμένης έρευνας είναι η μηχανική μάθηση, η οποία με τη σειρά της ανήκει σε ένα ευρύτερο πεδίο, την τεχνητή νοημοσύνη.

Η τεχνητή νοημοσύνη (Artificial Intelligence - AI) αποτελεί κλάδο της επιστήμης των υπολογιστών και εστιάζει στην ανάπτυξη αλγορίθμων και συστημάτων που είναι ικανά να εκτελούν εργασίες οι οποίες απαιτούν ανθρώπινη νοημοσύνη. Στον ευρύτερο αυτό τομέα που ονομάζεται τεχνητή νοημοσύνη υπάγονται η μηχανική μάθηση, η βαθιά μάθηση, η αναγνώριση προτύπων (Pattern Recognition), η αναγνώριση εικόνας (Image Recognition), οι ευφυείς πράκτορες (Intelligent Agents) κ.ά.

Η τεχνητή νοημοσύνη βασίζεται στην αξιοποίηση της υπολογιστικής ισχύος των συστημάτων, ούτως ώστε να γίνει η ανάλυση και επεξεργασία δεδομένων με απώτερο στόχο την εξαγωγή λογικών αποτελεσμάτων.

1.5 Μηχανική Μάθηση

Η μηχανική μάθηση (machine learning) είναι ένας σημαντικός τομέας της τεχνητής νοημοσύνης (artificial intelligence), που επικεντρώνεται στην ανάπτυξη αλγορίθμων και μοντέλων που επιτρέπουν στους υπολογιστές να μαθαίνουν από ένα σύνολο δεδομένων και να βελτιώνουν την απόδοσή τους αυτόνομα με την πάροδο του χρόνου. Με τη βοήθεια της μηχανικής μάθησης οι υπολογιστές μπορούν να εντοπίσουν πρότυπα-μοτίβα (patterns), να κάνουν προβλέψεις και να οδηγηθούν στη λήψη αποφάσεων έπειτα από ανάλυση των δεδομένων.

Η διαδικασία υλοποίησης και εφαρμογής ενός μοντέλου μηχανικής μάθησης αποτελείται από τέσσερα βασικά βήματα. Αρχικά, γίνεται η συλλογή των δεδομένων που θα χρησιμοποιηθούν. Τα δεδομένα αυτά έπειτα περνούν από το στάδιο της προεπεξεργασίας και του καθαρισμού, ούτως ώστε να είναι έτοιμα για μελλοντική ανάλυση. Αφού γίνει η κατάλληλη επιλογή του αλγορίθμου μηχανικής μάθησης, το αντίστοιχο μοντέλο εκπαιδεύεται με τα δεδομένα. Τέλος, η απόδοση του μοντέλου αξιολογείται με συγκεκριμένες μετρικές προκειμένου να ελεγχθεί η ακρίβεια με την οποία κάνει τις προβλέψεις.

Τα βασικά είδη μηχανικής μάθησης διακρίνονται σε (i) μη-επιβλεπόμενη μάθηση (unsupervised learning), (ii) ημι-επιβλεπόμενη μάθηση (semi-supervised learning), (iii) επιβλεπόμενη μάθηση (supervised learning), (iv) ενισχυτική μάθηση (reinforcement learning) και (v) μάθηση από μεταφορά (transfer learning).

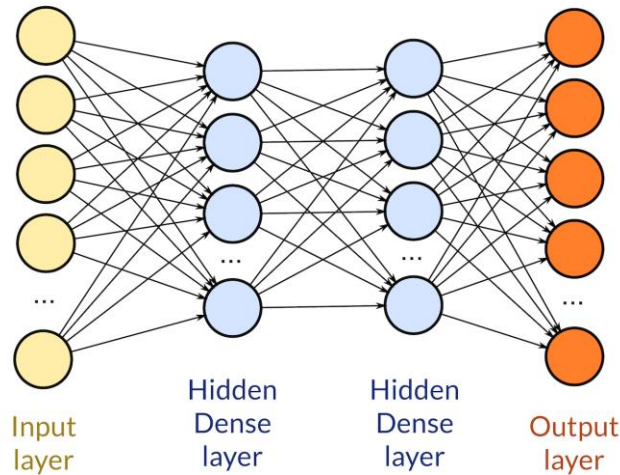
1.6 Βαθιά Μάθηση

Η βαθιά μάθηση (deep learning) είναι ένας κλάδος της μηχανικής μάθησης που εστιάζει στη χρήση πολυεπίπεδων νευρωνικών δικτύων (multi-layer neural networks) για την ανάλυση και επεξεργασία μεγάλων όγκων δεδομένων.

Τα βαθιά νευρωνικά δίκτυα αποτελούνται από πολλές στοιβάδες τεχνητών νευρώνων, που προσπαθούν να μιμηθούν τη λειτουργία του ανθρώπινου εγκεφάλου. Κάθε στοιβάδα μαθαίνει να αναγνωρίζει συγκεκριμένα χαρακτηριστικά των δεδομένων, τα οποία μεταφέρονται στην επόμενη στοιβάδα. Η διαδικασία αυτή επαναλαμβάνεται μέχρι να φτάσει σε ένα τελικό επίπεδο, στο οποίο πραγματοποιείται η τελική πρόβλεψη.

Χαρακτηριστικό παράδειγμα χρήσης βαθιάς μάθησης είναι η ανάλυση εικόνας. Τα βαθιά νευρωνικά δίκτυα είναι ικανά να αναγνωρίσουν πρόσωπα, αντικείμενα και σκηνές με μεγάλη ακρίβεια και χρησιμοποιούνται σε τομείς όπως είναι η ιατρική διάγνωση, όπου μπορούν να εντοπιστούν μοτίβα ή/και ανωμαλίες, για παράδειγμα, σε εικόνες που έχουν προκύψει με τη χρήση νευροαπεικονιστικών τεχνικών.

Ένας ακόμα τομέας στον οποίο έχει μεγάλη συνεισφορά η βαθιά μάθηση είναι η επεξεργασία φυσικής γλώσσας. Οι υπολογιστές με τη βοήθεια των βαθιών νευρωνικών δικτύων μπορούν τόσο να κατανοήσουν όσο και να δημιουργήσουν κείμενα σε ανθρώπινη-φυσική γλώσσα. Επιπλέον, γίνεται έτσι επιτρεπτή η δημιουργία ψηφιακών βοηθών όπως, ενδεικτικά, η Siri της Apple και η Alexa της Amazon, καθώς και η δημιουργία συστημάτων αυτόματης μετάφρασης, όπως το Google Translate. Τα τελευταία χρόνια έχει σημειωθεί σημαντική εξέλιξη στον τομέα της ανάπτυξης μεγάλων γλωσσικών μοντέλων (large language models), όπως είναι το GPT-4 και το Google Bard. Αυτά τα μοντέλα έχουν την ικανότητα να κατανοούν και να παράγουν κείμενα με μεγάλη φυσικότητα παρέχοντας ισχυρές δυνατότητες για την εξαγωγή πληροφορίας.



Εικόνα 3 | Βασική αρχιτεκτονική μοντέλου βαθιάς μάθησης

1.7 Επιβλεπόμενη μάθηση

Η επιβλεπόμενη μάθηση είναι μια μέθοδος στη μηχανική μάθηση, στην οποία τα μοντέλα εκπαιδεύονται με τη χρήση δεδομένων που περιέχουν τόσο τις εισόδους όσο και τις αντίστοιχες ετικέτες (labels). Με αυτόν τον τρόπο, το μοντέλο μαθαίνει να συνδέει τα δεδομένα εισόδου με τις σωστές ετικέτες-εξόδους. Η διαδικασία αυτή αποτελείται από τέσσερα στάδια. Αρχικά, γίνεται η συλλογή των δεδομένων, ακολουθεί η εκπαίδευση (training) του μοντέλου, η αξιολόγησή του (validation) και, τέλος, γίνεται η εφαρμογή του μοντέλου σε νέα δεδομένα με στόχο τη βελτίωση της ακρίβειας (testing).

Η επιβλεπόμενη μάθηση εφαρμόζεται σε διάφορα πεδία, όπως, για παράδειγμα, η πρόβλεψη οικονομικών τιμών και η πρόβλεψη ενέργειας. Πιο συγκεκριμένα, στην πρόβλεψη ενέργειας, τα μοντέλα επιβλεπόμενης μάθησης αποσκοπούν στην πρόβλεψη της ενεργειακής κατανάλωσης ή της ενεργειακής τιμής βασισμένα σε ιστορικά δεδομένα και άλλους παράγοντες που επηρεάζουν την ενεργειακή ζήτηση.

Όσον αφορά την παρούσα εργασία, τα μοντέλα επιβλεπόμενης μάθησης που χρησιμοποιούνται για τη μοντελοποίηση των γεωχωρικών δεδομένων (geospatial data) είναι τα συνελκτικά νευρωνικά δίκτυα (Convolutional Neural Networks - CNNs) [1], τα δίκτυα μακράς βραχυπρόθεσμης μνήμης (Long Short-Term Memory -LSTM) [2] και τα χρονικά συνελκτικά δίκτυα γράφων (Temporal Graph Convolutional Network - T-GCN) [3].

1.8 Μοντέλα Μηχανικής Μάθησης

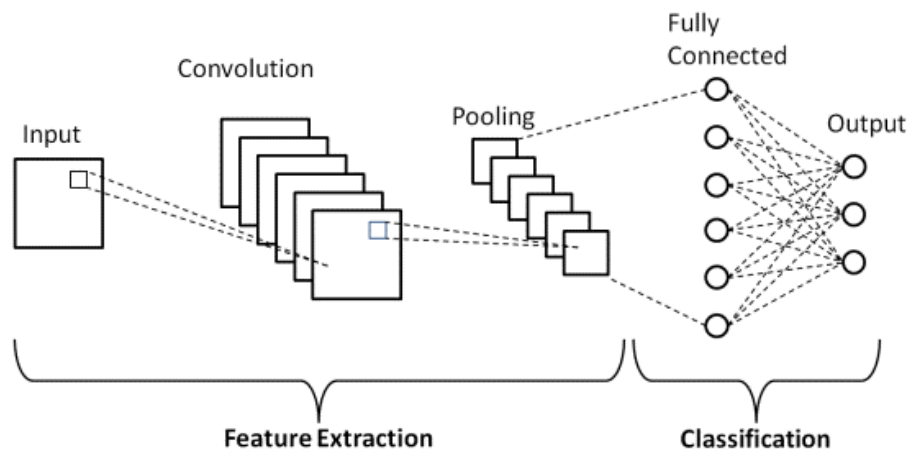
Σε αυτή την ενότητα του κεφαλαίου, παρατίθενται θεωρητικές πληροφορίες σχετικά με τα μοντέλα μηχανικής μάθησης που χρησιμοποιήθηκαν στην παρούσα εργασία. Συγκεκριμένα, αναλύεται η λειτουργία των μοντέλων: Convolutional Neural Networks, Convolutional LSTM hybrid και Temporal Graph Convolutional Networks.

1.8.1 Convolutional Neural Network - CNN

Τα Συνελκτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks - CNNs) αποτελούν ένα είδος αλγορίθμων επιβλεπόμενης μάθησης που χρησιμοποιούνται για την ταξινόμηση των εισερχόμενων δεδομένων (κυρίως εικόνων). Η ταξινόμηση αυτή γίνεται με τη βοήθεια των επιπέδων συνέλιξης που απαρτίζουν το CNN, τα οποία αναγνωρίζουν πρότυπα (pattern recognition) από τα εισερχόμενα δεδομένα. Χρησιμοποιούνται λειτουργίες όπως η ενεργοποίηση (Activation), η υποδειγματοληψία (Pooling) κ.ά. για την τελική πρόβλεψη, που καθιστούν εφικτή την αναγνώριση και την κατηγοριοποίηση οπτικών μοτίβων.

Παράλληλα με την αναγνώριση προτύπων και την ανάλυση εικόνας, τα συνελκτικικά νευρωνικά δίκτυα έχουν βρει εφαρμογή και σε χρονοσειρές, καθώς μπορούν να εξάγουν χαρακτηριστικά από δεδομένα πολλών διαστάσεων, καθιστώντας τα δίκτυα αυτά ιδανικά για την ανάλυση πολύπλοκων μοτίβων. Τα βασικά επίπεδα ενός τέτοιου δικτύου είναι:

- Συνελκτικικά Επίπεδα (Convolutional Layers): Υπεύθυνα για την εξαγωγή χαρακτηριστικών από την είσοδο, χρησιμοποιώντας φίλτρα για την εφαρμογή συνελκτικών πράξεων σε αυτήν, εντοπίζοντας χαρακτηριστικά όπως ακμές και πρότυπα.
- Επίπεδα Υποδειγματοληψίας (Pooling Layers): Μειώνουν τη διαστατικότητα των χαρακτηριστικών που εξάγονται από τα συνελκτικά επίπεδα και λειτουργούν ως ένα παράθυρο, εφαρμόζοντας μια λειτουργία λήψης μέγιστου ή μέσου στα δεδομένα που δέχονται ως είσοδο.
- Πλήρως Συνδεδεμένα Επίπεδα (Fully Connected Layers): Είναι υπεύθυνα για την ταξινόμηση των χαρακτηριστικών που εξήχθησαν από τα παραπάνω επίπεδα. Κάθε νευρώνας ενός τέτοιου επιπέδου συνδέεται με κάθε νευρώνα του προηγούμενου επιπέδου, ενεργοποιώντας μια γραμμική συνάρτηση ανάμεσα στα επίπεδα.



Εικόνα 4 | Αρχιτεκτονική CNN

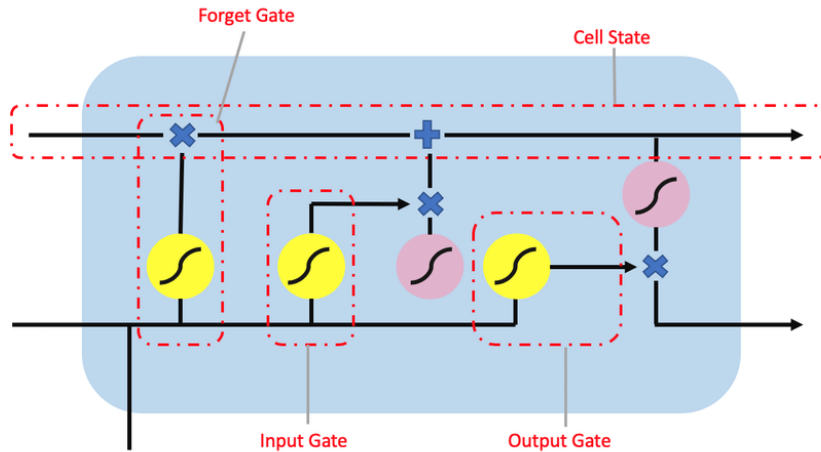
1.8.2 Long Short-Term Memory - LSTM

Τα Δίκτυα Μακράς Βραχυπρόθεσμης Μνήμης (LSTM) είναι μια ειδική κατηγορία επαναληπτικών νευρωνικών δικτύων (Recurrent Neural Networks - RNNs). Έχουν σχεδιαστεί για να διαχειρίζονται ακολουθιακά δεδομένα (sequential data) και να χρησιμοποιούνται σε αντίστοιχες εργασίες, όπως, ενδεικτικά, η πρόβλεψη χρονοσειρών (time series), η κατανάλωση ενέργειας, που αποτελεί το πρόβλημα στο οποίο εστιάζει η παρούσα εργασία, η ανάλυση χρόνου και η επεξεργασία φυσικής γλώσσας. Τα παραπάνω είναι εφικτά λόγω της ικανότητας των εν λόγω δικτύων να αποθηκεύουν πληροφορίες για μεγάλα χρονικά διαστήματα. Τα παραδοσιακά RNN συχνά εμφανίζουν δυσκολίες στη διατήρηση πληροφοριών για μεγάλες ακολουθίες. Αυτό συμβαίνει επειδή παρατηρείται το φαινόμενο της εξαφάνισης της κλίμακας των συναρτήσεων σφάλματος κατά τη διάρκεια της εκπαίδευσης. Τα LSTM επιλύουν αυτό το πρόβλημα χρησιμοποιώντας μια πιο πολύπλοκη δομή που βασίζεται σε τρεις πύλες. Αυτές οι πύλες αποτελούν τους ρυθμιστές ροής των πληροφοριών μέσα στη μνήμη, βοηθώντας το δίκτυο να μάθει ποιες πληροφορίες είναι χρήσιμες και ποιες πρέπει να απορριφθούν. Οι πύλες του LSTM λειτουργούν ως εξής:

- Πύλες εισόδου (Input Gates): Καθορίζουν ποιες νέες πληροφορίες αποθηκεύονται στη μνήμη.
- Πύλες λήθης (Forget Gates): Καθορίζουν ποιες πληροφορίες απορρίπτονται από τη μνήμη.

- Πύλες εξόδου (Output Gates): Καθορίζουν ποιες πληροφορίες χρησιμοποιούνται για την παραγωγή της εξόδου.

Αυτή η δομή επιτρέπει στα LSTM να διατηρούν σημαντικές πληροφορίες για μεγαλύτερα χρονικά διαστήματα, κάτι που τα καθιστά ιδανικά για εφαρμογές όπως η ανάλυση χρηματοοικονομικών δεδομένων και η πρόβλεψη τάσεων.



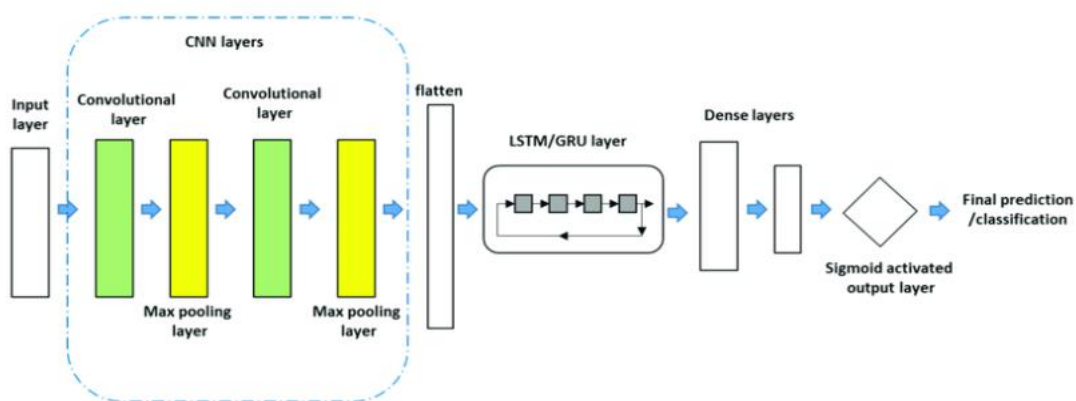
Εικόνα 5 | Αρχιτεκτονική LSTM

1.8.3 Convolutional LSTM Hybrid

Τα συνελκτικά νευρωνικά δίκτυα και τα δίκτυα LSTM αποτελούν δύο ισχυρά εργαλεία βαθιάς μάθησης. Ο συνδυασμός των δύο επιτρέπει την επεξεργασία των δεδομένων που περιέχουν τόσο χωρικές όσο και χρονικές διαστάσεις.

Τα CNN, όπως προαναφέρθηκε, είναι ιδιαίτερα αποτελεσματικά στην επεξεργασία δεδομένων πλέγματος, επειδή μπορούν να εξάγουν σημαντικά χωρικά χαρακτηριστικά μέσω των συνελκτικών στοιβάδων τους. Από την άλλη, τα LSTM είναι σχεδιασμένα για τη διαχείριση ακολουθιών και χρονοσειρών, διατηρώντας, παράλληλα, σημαντικές πληροφορίες για μεγάλα χρονικά διαστήματα.

Ένα υβριδικό μοντέλο που συνδυάζει τα CNN και τα LSTM είναι ιδιαίτερα αποτελεσματικό σε προβλήματα που περιέχουν χωρικά και χρονικά δεδομένα, όπως τα προβλήματα ενεργειακής κατανάλωσης [4][5]. Τα CNN μπορούν να εξάγουν χαρακτηριστικά από πλέγματα που απεικονίζουν την κατανομή της ζήτησης σε μια περιοχή, ενώ τα LSTM να αναλύσουν τις χρονικές αλληλουχίες αυτών των χαρακτηριστικών και να προβλέψουν μελλοντικές ενεργειακές τάσεις ζήτησης.



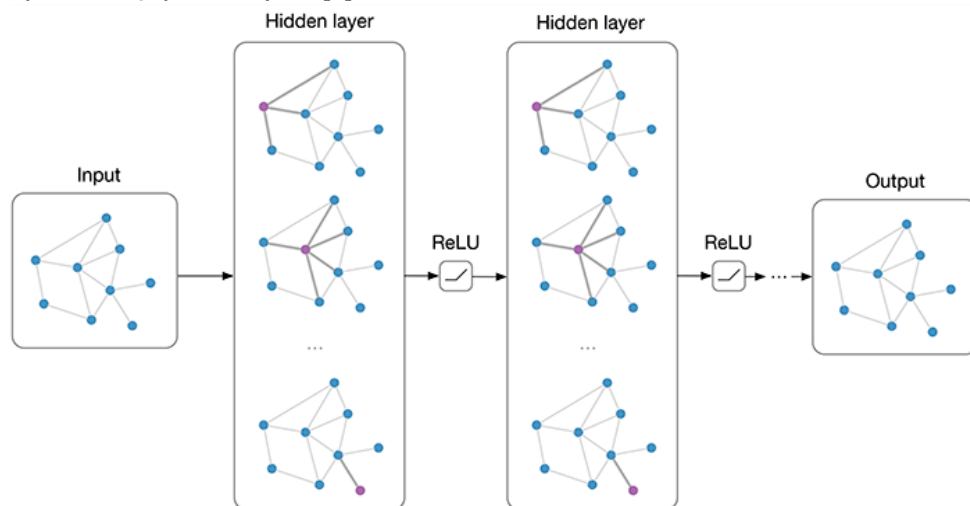
Εικόνα 6 | Αρχιτεκτονική υβριδικού CNN+LSTM

1.8.4 Graph Convolutional Network - GCN

Τα Γραφήματα Συνελκτικών Νευρωνικών Δικτύων (Graph Convolutional Networks) είναι ένα είδος νευρωνικών δικτύων που έχουν σχεδιαστεί για να επεξεργάζονται δεδομένα που έχουν τη μορφή γραφημάτων. Τα δεδομένα γραφημάτων περιλαμβάνουν κόμβους και ακμές που μπορούν να αναπαραστήσουν διάφορα συστήματα, όπως για παράδειγμα, κοινωνικά δίκτυα και μοριακές δομές [6][7].

Τα δίκτυα αυτά λειτουργούν εφαρμόζοντας συνελκτικές λειτουργίες σε γραφήματα, παρόμοιες με αυτές που εφαρμόζονται στα συνελκτικά νευρωνικά δίκτυα (CNN) για την επεξεργασία εικόνας. Σε αντίθεση με τα CNN, δεν δουλεύουν με δεδομένα πλέγματος αλλά επεξεργάζονται ανώμαλες δομές γραφημάτων. Κάθε κόμβος ξεχωριστά ενημερώνει την κατάστασή του συνδυάζοντας πληροφορίες από γειτονικούς κόμβους, καθιστώντας έτσι εφικτή τη ροή πληροφοριών σε όλο το γράφημα.

Τα δίκτυα GCN χρησιμοποιούνται επίσης για την ανάλυση δεδομένων γραφημάτων από δίκτυα διανομής ενέργειας. Μπορούν να εντοπίσουν μοτίβα κατανάλωσης σε διαφορετικές τοποθεσίες, όπως και να βελτιστοποιήσουν τη ροή ενέργειας στο δίκτυο, βελτιώνοντας έτσι, συνολικά, την κατανομή των πόρων [8].



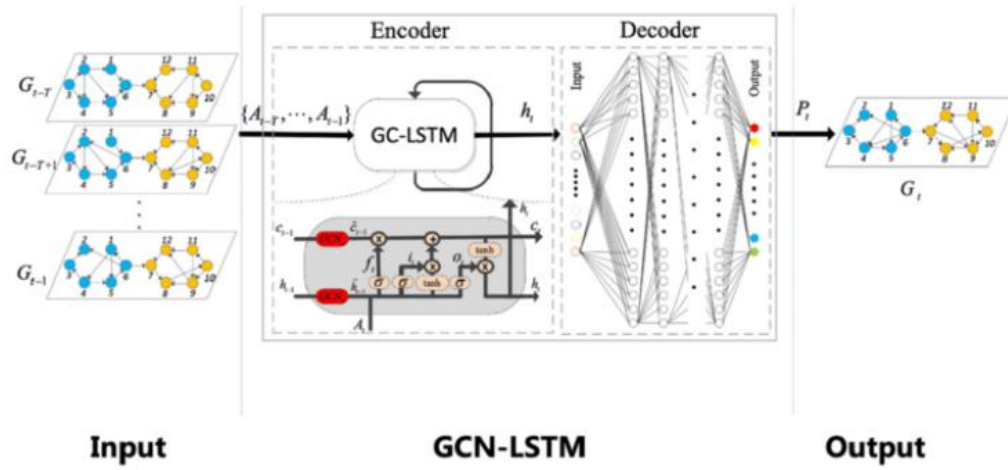
Εικόνα 7 | Αρχιτεκτονική GCN

1.8.5 Temporal Graph Convolutional Network - T-GCN

Τα Γραφήματα Συνελκτικών Νευρωνικών Δικτύων με Χρονικές Διαστάσεις (Temporal Graph Convolutional Networks - T-GCN) είναι μια τεχνική βαθιάς μάθησης που συνδυάζει τις δυνατότητες των γραφημάτων συνελκτικών νευρωνικών δικτύων (GCN) με την ανάλυση χρονοσειρών. Αυτό κάνει εφικτή την επεξεργασία και την ανάλυση δεδομένων που έχουν τη μορφή γραφήματος και εμφανίζουν μεταβολές στον χρόνο.

Τα T-GCN εφαρμόζονται στα δεδομένα γραφημάτων που με την πάροδο του χρόνου εμφανίζουν αλλαγές. Συνδυάζουν τις λειτουργίες του GCN με χρονικά νευρωνικά δίκτυα, όπως είναι τα LSTM, προκειμένου να ληφθούν υπόψη οι χρονικές σχέσεις και αλληλεπιδράσεις των δεδομένων.

Η λειτουργία των T-GCN περιλαμβάνει δύο βασικά βήματα. Αρχικά, εφαρμόζονται συνελκτικές λειτουργίες για την εξαγωγή χαρακτηριστικών από το γράφημα σε κάθε χρονική στιγμή. Έπειτα, οι χρονικές αλληλεπιδράσεις μεταξύ των χρονικών σημείων μοντελοποιούνται με τη χρήση LSTM δικτύων. Με αυτόν τον τρόπο, ενσωματώνονται τόσο οι χωρικές σχέσεις του γραφήματος όσο και οι χρονικές δυναμικές των δεδομένων.



Εικόνα 8 | Αρχιτεκτονική T-GCN με χρήση LSTM

2. Σχετικές Εργασίες

Σε αυτό το κεφάλαιο παρατίθενται οι σχετικές state-of-the-art εργασίες που χρησιμοποιούνται μοντέλα-μέθοδοι μηχανικής και βαθιάς μάθησης. Το πρόβλημα που προσπαθούν να επιλύουν οι έρευνες αυτές είναι παρόμοιο με αυτό που προσεγγίζουμε σε αυτήν την εργασία.

2.1 Πρόβλεψη ημερήσιας απαιτούμενης ενέργειας με Deep Neural Networks

Η έρευνα των Krielkinge, G.V., Cauwer, C.D, Sapountzoglou, N., Coosemans, T., Messagie, M. (2021), αναδεικνύει τη δυνατότητα της ημερήσιας πρόβλεψης της ενεργειακής ζήτησης για τη φόρτιση ηλεκτρικών οχημάτων. Οι συγγραφείς του άρθρου αναφέρουν πως πρόκειται για μια καινοτόμο προσέγγιση στο ζήτημα της πρόβλεψης της ενεργειακής ζήτησης, καθώς με την αύξηση του αριθμού των ηλεκτρικών οχημάτων που κινούνται στα οδικά δίκτυα και κατα συνέπεια της ζήτησης για φόρτιση, υπάρχουν ανησυχίες για τις επιπτώσεις που θα υπάρξουν στο ηλεκτρικό δίκτυο [9]. Για αυτόν τον λόγο, όπως αναφέρουν, είναι σημαντική η ανάπτυξη μοντέλων μάθησης που θα παρέχουν ακριβείς προβλέψεις.

Το άρθρο κάνει αναφορά για τη χρήση ενός εξελιγμένου νευρωνικού δικτύου βαθιάς μάθησης για την πρόβλεψη της ενεργειακής ζήτησης των ηλεκτρικών οχημάτων με χρονική ανάλυση δεκαπέντε (15) λεπτών.

Το εξελιγμένο μοντέλο αυτό αποτελείται από τεχνητά νευρωνικά δίκτυα (artificial neural networks - ANN), αναδρομικά νευρωνικά δίκτυα (recurrent neural networks - RNN) και δίκτυα μακράς βραχυπρόθεσμης μνήμης (Long Short-Term Memory - LSTM). Στο νευρωνικό δίκτυο γίνεται προσθήκη χαρακτηριστικών/δεδομένων καιρού και ημερολογίου, τα οποία συμβάλλουν στη βελτίωση της τελικής πρόβλεψης.

Το εν λόγω μοντέλο εφαρμόστηκε στον χώρο φόρτισης ηλεκτρικών οχημάτων ενός νοσοκομείου, σημειώνοντας μείωση του μέσου απόλυτου σφάλματος (mean absolute error - MAE) κατά περίπου 29 τοις εκατό (29 %) και 19,2 (19,2 %) τοις εκατό στο ριζικό μέσο τετραγωνικό σφάλμα (root mean squared error - RMSE), αντίστοιχα, αναδεικνύοντας έτσι τη σημαντική βελτίωση που σημειώθηκε στην ακρίβεια των προβλέψεων του μοντέλου.

Πιο συγκεκριμένα, η έρευνα αυτή προβάλλει τη δυνατότητα πρόβλεψης της ενεργειακής ζήτησης για τη φόρτιση ηλεκτρικών οχημάτων για την επόμενη ημέρα, σημειώνοντας MAE μικρότερο από 1 kiloWatt (kW). Αυτού του είδους οι προβλέψεις αποτελούν σημαντικό παράγοντα για τη διαχείριση της αυξανόμενης πίεσης που αντιμετωπίζουν τα δίκτυα φόρτισης λόγω της αυξανόμενης χρήσης ηλεκτρικών οχημάτων.

2.2 Πρόβλεψη μηνιαίας απαιτούμενης ενέργειας με την μεθοδολογία GEORGE

Η δεύτερη εργασία που παραθέτουμε είναι των Meddi, S., Cavaglioni, S., Cerquitelli, T., Manfredi, E., Regalia, A., Menolascino, R., Zardo, G. (2023) και προτείνει τη χρήση μιας μεθοδολογίας την οποία οι συγγραφείς ονομάζουν GEORGE (enerGy dEmand fOrecasting foR charGing infrastruCTurE), μια καινοτόμο προσέγγιση για την πρόβλεψη της μηνιαίας ενεργειακής ζήτησης σε υποδομές φόρτισης ηλεκτρικών οχημάτων [10].

Στην προσέγγιση αυτή χρησιμοποιούνται ιστορικά δεδομένα για την παρακολούθηση των σταθμών φόρτισης και, στη συνέχεια, την ανάπτυξη των μοντέλων πρόβλεψης Random Forest και XGBoost.

Η μεθοδολογία αυτή αποτελείται από μια διαδικασία ανακάλυψης γνώσης από δεδομένα (Knowledge Discovery process from Data - KDD), η οποία είναι προσαρμοσμένη στις επιχειρησιακές ανάγκες, δηλαδή την υποστήριξη της πιο αποδοτικής ανάπτυξης και αξιοποίησης των υπάρχουσών υποδομών φόρτισης. Πιο συγκεκριμένα, η μεθοδολογία GEORGE αποτελείται τα εξής τέσσερα στάδια:

- *Προεπεξεργασία δεδομένων*, για τη συγχώνευση δεδομένων από διαφορετικές πηγές και τον καθορισμό τους.
- *Μετασχηματισμός δεδομένων*, για την προσαρμογή της δομής των εισερχόμενων δεδομένων.

- *Προγνωστική ανάλυση*, για την εξαγωγή του κατάλληλου μοντέλου για την ακριβή πρόβλεψη.
- *Αξιολόγηση και ερμηνεία των αποτελεσμάτων*, για την εκτίμηση της ποιότητας του κάθε μοντέλου μέσω συγκεκριμένων μετρήσεων.

Κεντρικό στοιχείο της προτεινόμενης μεθοδολογίας είναι η ταξινόμηση των στοιχείων με βάση την ενεργειακή κατανάλωση σε kWh. Η ταξινόμηση αυτή προσφέρει σημαντικές πληροφορίες για τη διαχείριση και την ανάπτυξη των υποδομών φόρτισης.

Τα μοντέλα που αναπτύχθηκαν σε αυτήν την έρευνα, εμφανίζουν ισχυρές επιδόσεις, κυρίως στην πρόβλεψη σημείων υψηλής χρήσης, γεγονός που καθιστά το εργαλείο αυτό χρήσιμο για την αποφυγή της υπερφόρτωσης του δικτύου και την ταυτόχρονη διασφάλιση της βέλτιστης λειτουργίας της υποδομής φόρτισης.

Για κάθε αλγόριθμο ταξινόμησης που χρησιμοποιήθηκε έγινε η κατάλληλη αξιολόγηση, με το πιο αποδοτικό μοντέλο να είναι το μοντέλο Random Forest που χρησιμοποιεί τρεις (3) κλάσεις, τριανταεπτά (37) χαρακτηριστικά πρόβλεψης και δύο (2) μήνες ιστορικών δεδομένων.

2.3 Πρόβλεψη απαιτούμενης ενέργειας με την χρήση Χρονοσειρών

Η τρίτη μελέτη που παρουσιάζουμε των Kim, Y.; Kim, S. (2021), αναλύει την πρόβλεψη της ενεργειακής ζήτησης κατά τη διάρκεια της φόρτισης ηλεκτρικών οχημάτων χρησιμοποιώντας μοντέλα χρονοσειρών [11]. Στόχος αυτής της έρευνας είναι η ανάπτυξη μεθόδων που παρέχουν ακριβείς προβλέψεις, που θα συμβάλουν στη βελτίωση της διαχείρισης και σχεδίασης υποδομών φόρτισης ηλεκτρικών οχημάτων.

Όπως προαναφέρθηκε, σε αυτή την εργασία γίνεται χρήση μοντέλων χρονοσειρών για την πρόβλεψη της ζήτησης φόρτισης. Συγκεκριμένα υλοποιούνται, αξιολογούνται και συγκρίνονται τέσσερα τέτοια μοντέλα: TBATS, ARIMA, ANN και LSTM. Τα μοντέλα αυτά αναλύουν ιστορικά δεδομένα με στόχο τον εντοπισμό και την εξόρυξη προτύπων που μπορούν να επεκταθούν στο μέλλον, παρέχοντας ακριβείς προβλέψεις.

Τα αποτελέσματα της έρευνας παρουσιάστηκαν σε γεωγραφικές κλίμακες μιας χώρας, μιας πόλης και ενός σταθμού φόρτισης, χρησιμοποιώντας παράλληλα πραγματικά δεδομένα για την εφαρμογή και σε άλλες περιοχές. Η ανάλυση των τεσσάρων μοντέλων έδειξε μεγάλη ακρίβεια στις μεγαλύτερες κλίμακες. Από την άλλη, στην κλίμακα του ενός σταθμού οι εξωγενείς μεταβλητές δεν επηρέασαν σε σημαντικό βαθμό την ακρίβεια. Αυτό συνέβη καθώς η ατομική συμπεριφορά είναι σημαντικός παράγοντας στον καθορισμό της κατανάλωσης.

Έπειτα, η μελέτη εμβαθύνει στον έλεγχο της ανθεκτικότητας για τη σταθερή παροχή ενέργειας. Συγκρίθηκαν τρία σενάρια: τρεις έξι και δώδεκα μήνες, αντίστοιχα. Οι ερευνητές διαπιστώνουν ότι οι παρελθοντικές τιμές ήταν ανεξαρτήτως σταθερές στις βραχυπρόθεσμες προβλέψεις, όμως έπαιξαν σημαντικό ρόλο τόσο στις μεσοπρόθεσμες όσο και στις μακροπρόθεσμες προβλέψεις.

Τέλος, σε δεδομένα μακρο-κλίμακας σχετικά με άλλα πρότυπα, το μοντέλο ARIMA με παλινδρομητές έδειξε τα καλύτερα αποτελέσματα, ακολουθούμενο από το TBATS, ANN και LSTM. Το μοντέλο TBATS αναμένεται να φανεί πιο χρήσιμο όταν είναι διαθέσιμες μόνο μονοδιάστατες τιμές, ενώ το μοντέλο LSTM σημειώνει την καλύτερη απόδοση για μικροδεδομένα.

2.4 Πρόβλεψη διανομής σταθμών φόρτισης ηλεκτρικών οχημάτων με χρήση μηχανικής μάθησης

Η έρευνα των M. C. El Rai, S. A. Hadi, H. A. Damis, A. Gawanmeh, (2022), εστιάζει σε ένα ζήτημα σχετικό με αυτό που προσπαθούμε να επιλύσουμε στην παρούσα εργασία. Συγκεκριμένα, στόχος αυτής της έρευνας είναι να προτείνει μια μεθοδολογία, όπου μέσα από τη χρήση τεχνικών μηχανικής μάθησης θα μπορεί να προβλέψει τις βέλτιστες τοποθεσίες για την εγκατάσταση των σταθμών φόρτισης ηλεκτρικών οχημάτων, συμβάλλοντας έτσι στην υποστήριξη των υποδομών [12].

Η μεθοδολογία που ακολουθεί το άρθρο περιλαμβάνει τη χρήση διαφόρων μοντέλων μηχανικής μάθησης, όπως K-κοντινότεροι γείτονες (K-Nearest Neighbors - KNN), λογιστική παλινδρόμηση (logistic regression), νευρωνικά δίκτυα (neural networks) και υποστηρικτικές

μηχανές διανυσμάτων (Support Vector Machines - SVM) για την πρόβλεψη των βέλτιστων τοποθεσιών για την εγκατάσταση σταθμών φόρτισης ηλεκτρικών οχημάτων (Electric Vehicle Charging Stations - EVCS).

Πιο συγκεκριμένα, επικεντρώνεται στην πόλη του Ντουμπάι στα Ηνωμένα Αραβικά Εμιράτα, όπου για την εκπαίδευση των μοντέλων, γίνεται αξιοποίηση δεδομένων, όπως η πυκνότητα του πληθυσμού, τα σημεία ενδιαφέροντος (Points Of Interest - POI) και η παρουσία καμερών ασφαλείας. Τα χαρακτηριστικά αυτά επιλέγονται από τους συγγραφείς λόγω της σημαντικότητας για την τοποθέτηση των EVCS, καθώς προβάλλουν περιοχές με υψηλή ζήτηση, ασφάλεια και προσβασιμότητα.

Τα αποτελέσματα της μελέτης δείχνουν ότι το μοντέλο KNN σημείωσε την υψηλότερη ακρίβεια στην επικύρωση των προβλέψεων με ποσοστό 89 τοις εκατό (89 %). Από αυτό συμπεραίνουμε ότι το συγκεκριμένο μοντέλο είναι πιο αποτελεσματικό συγκριτικά με άλλα μοντέλα όταν πρόκειται για την πρόβλεψη κατάλληλων τοποθεσιών για την εγκατάσταση σταθμών φόρτισης σε αστικά κέντρα όπως το Ντουμπάι. Η υψηλή ακρίβεια που παρέχει το μοντέλο αυτό το καθιστά πολύτιμο εργαλείο για τους αστικούς σχεδιαστές, όπως χαρακτηριστικά αναφέρουν οι συγγραφείς.

Η χρήση διαφόρων χαρακτηριστικών όπως η πυκνότητα του πληθυσμού και τα POI's καταδεικνύουν τη σημασία της ένταξης πολυδιάστατων δεδομένων στα μοντέλα πρόβλεψης, ενώ, παράλληλα, οι ερευνητές τονίζουν τη σημασία της επιλογής των κατάλληλων μοντέλων μηχανικής μάθησης, αφού το KNN υπερέιχε όλων των υπολοίπων σε αυτή την έρευνα.

2.5 Πρόβλεψη ζητούμενης ενέργειας με βάση την αρχιτεκτονική Transformer

Η πέμπτη κατά σειρά μελέτη των X. Huang, D. Wu and B. Boulet, (2023) που παρουσιάζουμε στο πλαίσιο της βιβλιογραφικής ανασκόπησης, εξετάζει την επίλυση του προβλήματος της ενεργειακής ζήτησης κατά τη φόρτιση ηλεκτρικών οχημάτων σε σταθμούς φόρτισης. Πιο συγκεκριμένα, τα μοντέλα που χρησιμοποιούνται στη συγκεκριμένη έρευνα βασίζονται στην αρχιτεκτονική των μοντέλων Transformer [13].

Οι ερευνητές της συγκεκριμένης μελέτης επισημαίνουν πως στις περισσότερες ήδη υπάρχουσες μεθόδους επίλυσης του συγκεκριμένου προβλήματος είναι απαραίτητη η ύπαρξη μεγάλου όγκου δεδομένων εκπαίδευσης, κάτι που δεν είναι πάντα εφικτό, κυρίως για νέα δίκτυα φορτιστών ηλεκτρικών οχημάτων. Με σκοπό την αντιμετώπιση των προβλημάτων που εμφανίζουν οι παραδοσιακές μέθοδοι πρόβλεψης, οι συγγραφείς προτείνουν τα μοντέλα Probformer και MetaProbformer.

Το πρώτο μοντέλο, είναι βασισμένο όπως προαναφέρθηκε, στην αρχιτεκτονική των Transformer. Οι transformer είναι γνωστοί για την ικανότητά τους να καταγράφουν μακροχρόνιες εξαρτήσεις σε σειριακά δεδομένα, γεγονός που τους κάνει ικανούς για την επίλυση του συγκεκριμένου προβλήματος. Το μοντέλο MetaProbformer αποτελεί επέκταση του αρχικού Probformer και βασίζεται στη μετα-μάθηση, γεγονός που επιτρέπει στο μοντέλο να προσαρμοστεί γρήγορα σε νέες και άγνωστες καταστάσεις. Αυτό είναι ιδιαίτερα σημαντικό για τους νέους σταθμούς φόρτισης, για τους οποίους υπάρχουν περιορισμένα ιστορικά δεδομένα.

Τα αποτελέσματα των δοκιμών των μοντέλων Probformer και MetaProbformer υπερέχουν με διαφορά σε σχέση με αυτά των βασικών μεθόδων πρόβλεψης, καθώς είναι πιο αποτελεσματικά στην αντιμετώπιση αβέβαιων και μη συνηθισμένων καταστάσεων που σχετίζονται με την πρόβλεψη της ενεργειακής κατανάλωσης κατά τη φόρτιση ενός ηλεκτρικού οχήματος.

2.6 Πρόβλεψη ενεργειακής ζήτησης ηλεκτρικών οχημάτων με Υπολογιστικά Μοντέλα

Η τελευταία έρευνα που παραθέτουμε σε αυτό το κεφάλαιο, είναι των Oyedeji, Mojeed O., AIDhaifallah, Mujahed, Rezk, Hegazy, Mohamed, Ahmed Ali A., (2023) και η οποία επικεντρώνεται στην ανάπτυξη και αξιολόγηση μοντέλων μηχανικής μάθησης για την πρόβλεψη κατανάλωσης ηλεκτρικών οχημάτων στη Βαρκελώνη [14].

Στόχος αυτής της μελέτης είναι η πρόβλεψη της ημερήσιας ενεργειακής ζήτησης. Οι συγγραφείς κάνουν χρήση πέντε (5) μοντέλων μηχανικής μάθησης. Συγκεκριμένα,

χρησιμοποιούνται η Υποστήριξη Διανυσματικής Παλινδρόμησης (Support Vector Regression - SVR), η Παλινδρόμηση με Γκαουσιανή Διαδικασία (Gaussian Process Regression - GPR), τα τεχνητά νευρωνικά δίκτυα ANN, τα Δέντρα Αποφάσεων (Decision Tree - DT) και η μάθηση συνόλου (Ensemble Learning).

Η συγκεκριμένη μελέτη εξετάζει δύο δομές μοντέλων. Το μοντέλο Τύπου I χρησιμοποιεί δεδομένα κατανάλωσης των προηγούμενων επτά ημερών ως είσοδο, ενώ το μοντέλο Τύπου II χρησιμοποιεί δεδομένα τόσο της προηγούμενης όσο και της ίδιας ημέρας, καθώς και της προηγούμενης εβδομάδας.

Τα μοντέλα Τύπου II, αν και χρησιμοποιούν λιγότερα χαρακτηριστικά εισόδου, είχαν συγκρίσιμη απόδοση με αυτά του Τύπου I. Η αξιολόγηση των μοντέλων έγινε με τη μετρική R^2 , στην οποία και τα πέντε μοντέλα εμφάνισαν ακρίβεια πρόβλεψης της τάξεως του 75-80%.

3. Μεθοδολογία

Σε αυτήν την ενότητα της εργασίας παρουσιάζεται η μεθοδολογία η οποία προτείνεται για την αποτελεσματική αντιμετώπιση του προβλήματος της πρόβλεψης της ενεργειακής ζήτησης σε ένα συγκεκριμένο δίκτυο φορτιστών ηλεκτρικών οχημάτων. Στη μεθοδολογία αυτή γίνεται χρήση αλγορίθμων βαθιάς μάθησης, οπότε η ύπαρξη ενός ικανού συνόλου δεδομένων είναι απαραίτητη.

3.1 Συλλογή Δεδομένων

Εναρκτήριο βήμα της επίλυσης του προβλήματος είναι η συγκέντρωση δεδομένων, κατάλληλα για το προς επίλυση πρόβλημα. Τα δεδομένα θα πρέπει να περιέχουν τις καταγραφές για την ενέργεια που καταναλώθηκε σε κάθε περίοδο φόρτισης, τη χρονοσφραγίδα (timestamp) της έναρξης της φόρτισης, το γεωγραφικό μήκος (Longitude) και γεωγραφικό πλάτος (Latitude) κάθε σταθμού, καθώς και την ονομασία/κωδικό κάθε σταθμού φόρτισης του δικτύου.

3.2 Επεξεργασία Δεδομένων

Πρωτεύον βήμα της επεξεργασίας του συνόλου δεδομένων που διαθέτουμε είναι η αναδιάταξή του σε ημερήσιες καταγραφές. Αυτό το βήμα προηγείται των υπολοίπων, επειδή η χρονοσφραγίδα που περιέχει η κάθε εγγραφή είναι της μορφής ΜΗΝΑΣ/ΗΜΕΡΑ/ΧΡΟΝΟΣ ΩΡΑ:ΛΕΠΤΟ.

Αποτέλεσμα αυτού του αρχικού βήματος είναι η μετατροπή της χρονοσφραγίδας και, κατ' επέκταση, του δείκτη του συνόλου δεδομένων στη μορφή ΧΡΟΝΟΣ-ΜΗΝΑΣ-ΗΜΕΡΑ. Αυτή η μετατροπή του συνόλου δεδομένων σε ημερήσιες καταγραφές θα μας βοηθήσει στη συνέχεια της εργασίας και, συγκεκριμένα, κατά τη διάρκεια διάσπασης του αρχικού συνόλου σε υποσύνολα εκπαίδευσης (training set) και δοκιμής (testing set).

3.3 Κανονικοποίηση Δεδομένων

Όσον αφορά την κανονικοποίηση των δεδομένων, χαρακτηριστικά όπως η χρονοσφραγίδα της περιόδου φόρτισης και το όνομα του σταθμού φόρτισης είναι αδύνατο να κανονικοποιηθούν, ενώ οι συντεταγμένες της θέσης του σταθμού απαγορεύεται να κανονικοποιηθούν. Έτσι, όταν αναφερόμαστε στην κανονικοποίηση των δεδομένων, εννοούμε αποκλειστικά το ποσό της ενεργειακής κατανάλωσης σε kWh που καταγράφεται.

Στη μεθοδολογία που παρουσιάζουμε, η κανονικοποίηση των δεδομένων γίνεται κατά τη διάρκεια διάσπασης του συνόλου σε υποσύνολα εκπαίδευσης και δοκιμής. **Σημείωση:** Τα κανονικοποιημένα δεδομένα χρησιμοποιούνται μόνο κατά την εκπαίδευση των αναπτυσσόμενων μοντέλων μηχανικής μάθησης, ενώ για τη δοκιμή των μοντέλων αυτών χρησιμοποιούνται οι μη-κανονικοποιημένες τιμές ενέργειας.

3.4 Διάσπαση Συνόλου Δεδομένων Για Τα Μοντέλα CNN και Convolutional LSTM

Όπως έχει προαναφερθεί, στην παρούσα εργασία γίνεται χρήση τριών μοντέλων βαθιάς μάθησης:

1. Συνελκτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks - CNNs)
2. Υβριδικό μοντέλο Συνελκτικών Νευρωνικών Δικτύων με στοιβάδες LSTM (Convolutional LSTM Hybrid).
3. Γραφήματα Συνελκτικών Νευρωνικών Δικτύων με Χρονικές Διαστάσεις (Temporal Graph Convolutional Networks - T-GCN).

Για τα πρώτα δύο μοντέλα, η διάσπαση του αρχικού συνόλου γίνεται με τη χρήση raster map. Η υλοποίηση του raster map γίνεται με τη χρήση των τριών συναρτήσεων: `findcell`, `getcell` και `build_rastermap`.

3.4.1 Η Συνάρτηση findcell

Η συνάρτηση findcell, εντοπίζει σε ποιο κελί ενός πλέγματος ανήκει ένα συγκεκριμένο γεωγραφικό σημείο (geographical point). Δέχεται ως είσοδο μια εγγραφή του συνόλου δεδομένων και δύο λίστες. Κάθε μια από αυτές τις λίστες περιέχει τα εκάστοτε όρια για το γεωγραφικό μήκος και το γεωγραφικό πλάτος, αντίστοιχα.

Η συνάρτηση συγκρίνει τις συντεταγμένες της γραμμής με τα όρια του πλέγματος και καθορίζει τους δείκτες των κελιών x και y , στα οποία αντιστοιχούν το γεωγραφικό μήκος και πλάτος, αντίστοιχα. Στο τέλος, η συνάρτηση επιστρέφει τους δείκτες που έχουν καθοριστεί. Ακολουθεί ο ψευδοκώδικας της συνάρτησης findcell:

ΕΝΑΡΞΗ ΣΥΝΑΡΤΗΣΗΣ findcell

Είσοδοι Συνάρτησης: μία γραμμή row του συνόλου δεδομένων, μια λίστα x_cell με τα όρια του γεωγραφικού μήκους, μια λίστα y_cell με τα όρια του γεωγραφικού πλάτους.

Έξοδοι Συνάρτησης: ο δείκτης $x_gridcell$ που δείχνει το όριο του γεωγραφικού μήκους, ο δείκτης $y_gridcell$ που δείχνει το όριο του γεωγραφικού πλάτους.

0: Για κάθε στοιχείο x στη λίστα $list_x$ με δείκτη i :

0.1: Έλεγξε εάν η τιμή του *Longitude* της γραμμής row είναι μικρότερη ή ίση του στοιχείου x .

0.2: Εάν ισχύει η συνθήκη της γραμμής 0.1, τότε όρισε τον δείκτη $x_gridcell$ ίσο με $i - 1$ ΚΑΙ σπάσε τον βρόγχο.

1: Τέλος επανάληψης.

2: Για κάθε στοιχείο y στη λίστα $list_y$ με δείκτη j :

2.1: Έλεγξε εάν η τιμή του *Latitude* της γραμμής row είναι μικρότερη ή ίση του στοιχείου y

2.2: Εάν ισχύει η συνθήκη της γραμμής 2.1, τότε όρισε τον δείκτη $y_gridcell$ ίσο με $j - 1$ ΚΑΙ σπάσε τον βρόγχο.

3: Τέλος επανάληψης.

4: Επέστρεψε τους δείκτες $x_gridcell$ και $y_gridcell$.

ΤΕΛΟΣ ΣΥΝΑΡΤΗΣΗΣ findcell

3.4.2 Η Συνάρτηση getcell

Η συνάρτηση getcell αναθέτει σε κάθε γραμμή row του συνόλου δεδομένων το κελί του πλέγματος στο οποίο ανήκει με βάση τις συντεταγμένες (*Longitude* και *Latitude*). Ορίζει τα όρια του πλέγματος, το οποίο έχει διαστάσεις 5×5 . Στη συνέχεια, για κάθε γραμμή καλεί τη συνάρτηση findcell για να προσδιορίσει σε ποιο κελί του πλέγματος ανήκει η γραμμή αυτή. Τα αποτελέσματα αποθηκεύονται σε δύο νέες στήλες του συνόλου δεδομένων $xcell$ και $ycell$. Τα νέα χαρακτηριστικά αυτά αντιστοιχούν στους δείκτες των κελιών x και y , αντίστοιχα. Ακολουθεί ο ψευδοκώδικας της συνάρτησης getcell:

ΕΝΑΡΞΗ ΣΥΝΑΡΤΗΣΗΣ getcell

Είσοδοι Συνάρτησης: το σύνολο δεδομένων *data*.

Έξοδοι Συνάρτησης: το ενημερωμένο σύνολο δεδομένων *data*.

0: Όρισε τη μεταβλητή $nx = 5$.

1: Όρισε τη μεταβλητή $ny = 5$.

2: Όρισε τη λίστα x με τα όρια του πλέγματος για το γεωγραφικό μήκος.

3: Όρισε τη λίστα y με τα όρια του πλέγματος για το γεωγραφικό πλάτος.

4: Αρχικοποίησε τη στήλη $xcell$ του συνόλου δεδομένων *data* με την τιμή -1.

5: Αρχικοποίησε τη στήλη $ycell$ του συνόλου δεδομένων *data* με την τιμή -1.

6: Για κάθε γραμμή row στο σύνολο δεδομένων *data* με δείκτη idx επανέλαβε:

6.1: Όρισε τις τιμές των $xcell$ και $ycell$ χαρακτηριστικών κάθε γραμμής με την κλήση της συνάρτησης findcell χρησιμοποιώντας ως εισόδους τη γραμμή row στην οποία βρίσκεται η επανάληψη.

7: Τέλος επανάληψης.

8: Επίστρεψε το ενημερωμένο σύνολο δεδομένων *data*.

ΤΕΛΟΣ ΣΥΝΑΡΤΗΣΗΣ *getcell*

3.4.3 Η Συνάρτηση *build_rastermap*

Η συνάρτηση *build_rastermap* δημιουργεί έναν τρισδιάστατο χάρτη που αποτυπώνει την κατανάλωση ενέργειας σε ένα πλέγμα χωρικών κελιών κατά τη διάρκεια του χρόνου, βασιζόμενο σε δεδομένα από σταθμούς φόρτισης.

Αρχικά, ορίζει ένα πλέγμα διαστάσεων 5x5 και υπολογίζει τον συνολικό αριθμό πλήρων ημερών και τον καταχωρεί στη μεταβλητή *num_h* μεταξύ της πρώτης και της τελευταίας ημερήσιας καταγραφής του συνόλου δεδομένων. Στη συνέχεια, αρχικοποιεί με μηδενικές τιμές έναν τρισδιάστατο πίνακα *raster_map* ο οποίος έχει διαστάσεις [*num_h + 1, 5, 5*].

Έπειτα, για κάθε ημέρα αυτού του διαστήματος, η συνάρτηση επιλέγει τα δεδομένα που αντιστοιχούν σε εκείνη την ημέρα και υπολογίζει για κάθε κελί του πλέγματος τον αριθμό των **μοναδικών σταθμών στις εγγραφές** (δηλαδή το πλήθος των σταθμών του δικτύου), χρησιμοποιώντας τις τιμές *xcell* και *ycell*, που έχουν προστεθεί στο *dataset* κατά την κλήση της συνάρτησης *getcell*. Εφόσον υπάρχουν σταθμοί, γίνεται ο υπολογισμός του συνολικού ποσού ενέργειας που καταναλώθηκε και αποθηκεύεται στο αντίστοιχο σημείο του πίνακα.

Τέλος, η συνάρτηση επιστρέφει τον ολοκληρωμένο τρισδιάστατο πίνακα *raster_map*, που παρέχει μια χωρική και χρονική κατανομή της ενεργειακής κατανάλωσης στα διαφορετικά κελιά του πλέγματος. Ακολουθεί ο ψευδοκώδικας της συνάρτησης *build_rastermap*:

ΕΝΑΡΞΗ ΣΥΝΑΡΤΗΣΗΣ *build_rastermap*

Είσοδοι Συνάρτησης: το σύνολο δεδομένων *data*.

Έξοδοι Συνάρτησης: τον τρισδιάστατο πίνακα *raster_map* που αποτυπώνει την ενεργειακή κατανάλωση.

0: Όρισε τη μεταβλητή *nx = 5*.

1: Όρισε τη μεταβλητή *ny = 5*.

2: Υπολόγισε τον αριθμό πλήρων ημερών *num_h* ως το ακέραιο μέρος της διαφοράς ανάμεσα στην τελευταία και την πρώτη χρονική στιγμή σε δευτερόλεπτα, διαιρεμένο με το πλήθος των δευτερολέπτων σε μια ημέρα.

3: Δημιούργησε έναν τρισδιάστατο πίνακα *raster_map* με διαστάσεις [*num_h + 1, nx, ny*], αρχικοποιώντας τον με μηδενικές τιμές.

4: Όρισε τη μεταβλητή *timestart* ως την πρώτη χρονική στιγμή των δεδομένων.

5: Για κάθε ημέρα *x* από 0 έως *num_h* επανάλαβε:

5.1: Όρισε τη μεταβλητή *window* ως το άθροισμα *timestart + x*.

5.2: Φίλτραρε τα δεδομένα για την τρέχουσα ημέρα και αποθήκευσέ τα στη μεταβλητή *current*.

5.3: Για κάθε κελί *y* στον οριζόντιο άξονα από 0 έως *nx - 1* επανάλαβε:

5.3.1: Για κάθε κελί *z* στον κάθετο άξονα από 0 έως *ny - 1* επανάλαβε:

5.3.1.1: Υπολόγισε τον αριθμό μοναδικών σταθμών φόρτισης *chargers* στο τρέχον κελί.

5.3.1.2: Έλεγξε εάν ισχύει η συνθήκη *chargers == 0*.

5.3.1.3: Εάν ισχύει η συνθήκη, τότε προχώρα στο επόμενο κελί.

5.3.1.4: Εάν δεν ισχύει η συνθήκη, τότε υπολόγισε το άθροισμα ενέργειας *Energy* για το τρέχον κελί και αποθήκευσέ το στον πίνακα *raster_map* στην θέση [*x, y, z*].

5.3.2: Τέλος εσωτερικού βρόγχου.

6: Τέλος εξωτερικού βρόγχου.

7: Επέστρεψε τον τρισδιάστατο πίνακα *raster_map*.

ΤΕΛΟΣ ΣΥΝΑΡΤΗΣΗΣ *build_rastermap*

3.4.4 Η Συνάρτηση `traintest`

Η συνάρτηση `traintest` είναι αυτή που με την κλήση της διασπά το σύνολο δεδομένων σε αντίστοιχα υποσύνολα εκπαίδευσης και δοκιμής. Δέχεται τέσσερις εισόδους:

- Το σύνολο δεδομένων `data`.
- Το όνομα του μοντέλου `model_name`, όπου πρόκειται να χρησιμοποιηθούν τα υποσύνολα για την εκπαίδευση και την αξιολόγησή του. **Σημείωση:** Το όνομα του μοντέλου λαμβάνει μια από τις τιμές `conv` και `convlstm`.
- Τον ορίζοντα πρόβλεψης `forecast_h`. **Σημείωση:** Ο ορίζοντας πρόβλεψης μπορεί να είναι μια από τις τιμές `1`, `7` και `30`.
- Τον τρισδιάστατο πίνακα `raster_map`.

Αρχικά, η συνάρτηση κάνει τον κατάλληλο έλεγχο εγκυρότητας για τις τιμές που έχουν ο ορίζοντας πρόβλεψης `forecast_h` και το όνομα του μοντέλου `model_name`. Συγκεκριμένα, ο ορίζοντας πρόβλεψης πρέπει να είναι είτε `1` είτε `7` είτε `30` ημέρες, ενώ το όνομα του μοντέλου είναι είτε `conv` είτε `convlstm`. Επιπλέον, ορίζει παραμέτρους, όπως ο αριθμός των χρονικών καθυστερήσεων `numlags` και οι διαστάσεις του πλέγματος `nx` και `ny`. Στη συνέχεια, γίνεται κανονικοποίηση του πίνακα `raster_map` βάσει της μέγιστης τιμής του, ενώ, παράλληλα, οι κενές NaN και οι μηδενικές τιμές του πίνακα αντικαθίστανται από μικρές τυχαίες τιμές.

Η συνάρτηση δημιουργεί έναν νέο πίνακα `matrixlags` που περιέχει τις καθυστερημένες τιμές των δεδομένων που πρόκειται να χρησιμοποιηθούν για την εκπαίδευση και δοκιμή του μοντέλου. Έπειτα, γίνεται αρχικοποίηση των υποσυνόλων εκπαίδευσης `X_train` και `y_train`, αλλά και των αντίστοιχων υποσυνόλων δοκιμής `X_test` και `y_test`. Ακολουθεί η συμπλήρωση των υποσυνόλων αυτών με τιμές από τον πίνακα `matrixlags`, ενώ εάν το μοντέλο είναι του τύπου `ConvLSTM` (δηλαδή εάν το `model_name` είναι `'convlstm'`), τότε γίνεται η κατάλληλη προσαρμογή των διαστάσεων των υποσυνόλων, προσθέτοντας μια επιπλέον διάσταση ούτως ώστε να ανταποκριθούν στις απαιτήσεις του μοντέλου. Για την εκπαίδευση των μοντέλων, γίνεται χρήση των δεδομένων από το 2012 έως και το 2020. Η δοκιμή των τριών μοντέλων στους δύο ορίζοντες πρόβλεψης `1` και `7` ημερών γίνεται με τη χρήση των καταγραφών από τις τελευταίες `30` ημέρες δεδομένων. Αντίστοιχα, για την μακροπρόθεσμη πρόβλεψη των `30` ημερών, χρησιμοποιήθηκε μεγαλύτερο σύνολο δοκιμής δεδομένων, των `120` τελευταίων ημερών. Τέλος, επιστρέφονται τα τέσσερα απαιτούμενα σύνολα εκπαίδευσης και δοκιμής, καθώς και η μέγιστη τιμή του `raster_map`, που θα χρησιμοποιηθεί στην αποκανονικοποίηση των προβλέψεων πριν την αξιολόγησή τους. Ακολουθεί ο ψευδοκώδικας της συνάρτησης.

ΕΝΑΡΞΗ ΣΥΝΑΡΤΗΣΗΣ `traintest`

Είσοδοι Συνάρτησης: το σύνολο δεδομένων `data`, το όνομα του μοντέλου `model_name`, ο ορίζοντας πρόβλεψης `forecast_h` και ο τρισδιάστατος πίνακας `raster_map`.

Έξοδοι Συνάρτησης: τα σύνολα εκπαίδευσης και δοκιμής που θα χρησιμοποιηθούν στα μοντέλα πρόβλεψης.

0: Εάν η τιμή της μεταβλητής `forecast_h` δεν είναι `1`, `7` ή `30`, τότε:

0.1: Εμφάνισε κατάλληλο μήνυμα λάθους και διάκοψε την εκτέλεση του κώδικα.

1: Εάν η τιμή της μεταβλητής `model_name` δεν είναι `conv` ή `convlstm`, τότε:

1.1: Εμφάνισε κατάλληλο μήνυμα λάθους και διάκοψε την εκτέλεση του κώδικα.

2: Όρισε τις παρακάτω παραμέτρους:

2.1: `numlags = 30`

2.2: `nx = 5`

2.3: `ny = 5`

3: Υπολόγισε τη μέγιστη τιμή του πίνακα `raster_map` κατά μήκος του χρονικού άξονα + `0.01` και αποθήκευσέ τη στη μεταβλητή `maxval`.

4: Κανονικοποίησε τον πίνακα `raster_map` διαιρώντας τον με την `maxval` και αποθήκευσέ τον σε έναν νέο πίνακα `norm`, κάνοντας παράλληλα τις παρακάτω αντικαταστάσεις:

4.1: Αντικατάστησε τις τιμές NaN με `0`.

4.2: Αντικατάστησε τις μηδενικές τιμές με τυχαίες τιμές από την κανονική κατανομή με μέσο όρο `0` και τυπική απόκλιση `0.01`.

- 5: Αρχικοποίησε με μηδενικές τιμές τον πίνακα *matrixlags* και όρισε τις διαστάσεις του ως $[raster_map.shape[0] - (numlags + forecast_h), numlags + forecast_h, nx, ny]$.
 - 6: Όρισε τη μεταβλητή $rng = matrixlags.shape[0]$.
 - 7: Εάν *forecast_h* in [1,7]: όρισε $itrain = rng - 30$, αλλιώς $itrain = rng - 120$
 - 8: Όρισε $itest = rng$.
 - 9: Για κάθε χρονικό σημείο *i* από 0 έως $rng - 1$ επανάλαβε:
 - 9.1: Αποθήκευσε στο *matrixlags[i]* τα δεδομένα από $norm[i:i+numlags+forecast_h, :, :]$.
 - 10: Τέλος επανάληψης
 - 11: Αρχικοποίησε με μηδενικές τιμές τους τετραδιάστατους πίνακες *X_train*, *y_train*, *X_test* και *y_test* και όρισε τις παρακάτω διαστάσεις:
 - 11.1: Για τον *X_train*, (*itrain*, *numlags*, *nx*, *ny*).
 - 11.2: Για τον *y_train*, (*itrain*, *forecast_h*, *nx*, *ny*).
 - 11.2: Για τον *X_test*, (*itest - itrain*, *numlags*, *nx*, *ny*).
 - 11.3: Για τον *y_test*, (*itest - itrain*, *forecast_h*, *nx*, *ny*).
 - 12: Για κάθε κελί στον οριζόντιο άξονα *x* από 0 έως $nx - 1$, επανάλαβε:
 - 12.1: Για κάθε κελί στον κάθετο άξονα *y* από 0 έως $ny - 1$, επανάλαβε:
 - 12.1.1: Γέμισε τα *X_train*, *y_train*, *X_test*, *y_test* με τις αντίστοιχες καθυστερημένες τιμές από τον *matrixlags*.
 - 12.2: Τέλος εσωτερικής επανάληψης.
 - 13: Τέλος επανάληψης.
 - 14: Εάν το *model_name* είναι 'convlstm', τότε:
 - 14.1: Προσάρμοσε τα υποσύνολα, επεκτείνοντας τα υποσύνολα κατά μία διάσταση.
 - 14.2: Επίστρεψε τα υποσύνολα.
 - 15: Αποκατανομοποίησε το *y_test*, πολλαπλασιάζοντας τα με το *maxval*.
 - 16: Αναδιάταξε τους άξονες των *X_train*, *y_train*, *X_test*, *y_test*.
 - 17: Επίστρεψε τα *X_train*, *y_train*, *X_test*, *y_test*, *maxval*.
- ΤΕΛΟΣ ΣΥΝΑΡΤΗΣΗΣ *traintest*

3.5 Διάσπαση Συνόλου Δεδομένων Για Το Μοντέλο T-GCN

Για τη διάσπαση του συνόλου δεδομένων με σκοπό την εκπαίδευση και αξιολόγηση του μοντέλου T-GCN δεν γίνεται χρήση του *raster map*. Αντιθέτως, χρησιμοποιείται ένας γράφος, του οποίου το πλήθος των κόμβων είναι ίσο με το πλήθος των σταθμών φόρτισης του δικτύου.

Η υλοποίηση αυτού του γράφου γίνεται με τη χρήση τριών συναρτήσεων: **distance**, **adj_norm**, **getGraph**. Για τη διάσπαση του συνόλου δεδομένων σε υποσύνολα εκπαίδευσης και δοκιμής με τη χρήση του γράφου χρησιμοποιείται η συνάρτηση **setupTGCN**. Ακολουθεί η περιγραφή της λειτουργίας και του ψευδοκώδικα για την κάθε συνάρτηση.

3.5.1 Η Συνάρτηση **distance**

Η συνάρτηση **distance** υπολογίζει την απόσταση μεταξύ δύο γεωγραφικών σημείων *x* και *y* σε χιλιόμετρα, χρησιμοποιώντας τη φόρμουλα Haversine. Τα σημεία *x* και *y* που δέχεται ως είσοδο η συνάρτηση είναι λεξικά (dictionaries), που περιέχουν τις συντεταγμένες γεωγραφικού μήκους και γεωγραφικού πλάτους, αντίστοιχα. Η απόσταση που υπολογίζεται επιστρέφεται σε χιλιόμετρα.

ΕΝΑΡΞΗ ΣΥΝΑΡΤΗΣΗΣ **distance**

Είσοδοι Συνάρτησης: Σημεία *x* και *y*.

Έξοδοι Συνάρτησης: Η απόσταση *dist* των σημείων σε χιλιόμετρα.

0: Όρισε την ακτίνα της γης στη μεταβλητή $R = 6373$.

1: Μετάτρεψε το γεωγραφικό πλάτος και μήκος των σημείων *x* και *y* σε ακτίνα.

- 2: Υπολόγισε τις διαφορές των γεωγραφικών πλατών και μηκών μεταξύ των δύο σημείων.
- 3: Εφάρμοσε τον τύπο Haversine για τον υπολογισμό της απόστασης.
- 4: Επέστρεψε την απόσταση σε χιλιόμετρα.

ΤΕΛΟΣ ΣΥΝΑΡΤΗΣΗΣ distance

3.5.2 Η Συνάρτηση `adj_norm`

Η συνάρτηση `adj_norm` δέχεται ως είσοδο έναν πίνακα γειτνίασης `adj`. Στη συνέχεια, υπολογίζει έναν πίνακα βαθμών (degree matrix) `d` και κανονικοποιεί τον πίνακα `adj` με τη βοήθειά του. Τέλος, επιστρέφει τον κανονικοποιημένο πλέον πίνακα `adjnorm`.

ΕΝΑΡΞΗ ΣΥΝΑΡΤΗΣΗΣ `adj_norm`

Είσοδοι Συνάρτησης: Πίνακας γειτνίασης `adj`.

Έξοδοι Συνάρτησης: Κανονικοποιημένος πίνακας γειτνίασης `adjnorm`.

0: Υπολόγισε τον πίνακα βαθμών `d` ως η δύναμη -0.5 των τιμών του πίνακα `adj`.

1: Υπολόγισε τον κανονικοποιημένο πίνακα `adjnorm`.

2: Επέστρεψε τον κανονικοποιημένο πίνακα `adjnorm`.

ΤΕΛΟΣ ΣΥΝΑΡΤΗΣΗΣ `adj_norm`

3.5.3 Η Συνάρτηση `getGraph`

Η συνάρτηση `getGraph` δημιουργεί έναν γράφο `G` χρησιμοποιώντας δεδομένα σταθμών. Για κάθε σταθμό του δικτύου, δημιουργείται ένας κόμβος στον `G` και αποθηκεύονται ως ιδιότητες του κόμβου οι γεωγραφικές πληροφορίες του σταθμού (όνομα, γεωγραφικό μήκος, γεωγραφικό πλάτος).

Έπειτα, υπολογίζεται η απόσταση μεταξύ κάθε ζεύγους κόμβων χρησιμοποιώντας τη συνάρτηση `distance`. Εάν αυτή η απόσταση είναι μικρότερη από 2.5 χιλιόμετρα, τότε προστίθεται μια ακμή που ενώνει τους δύο κόμβους. Το βάρος της ακμής είναι ίσο με το εκθετικό αρνητικό της απόστασης. Τέλος, δημιουργείται και, στη συνέχεια, κανονικοποιείται με τη συνάρτηση `adj_norm` ο πίνακας γειτνίασης, ο οποίος επιστρέφεται μαζί με τον γράφο.

ΕΝΑΡΞΗ ΣΥΝΑΡΤΗΣΗΣ `getGraph`

Είσοδοι Συνάρτησης: Σύνολο δεδομένων `data`.

Έξοδος Συνάρτησης: Ο γράφος `G` και ο κανονικοποιημένος πίνακας γειτνίασης `adj`.

0: Δημιούργησε έναν κενό γράφο `G`.

1: Για κάθε μοναδικό σταθμό στο σύνολο δεδομένων `data` επανάλαβε:

1.1: Πρόσθεσε τον σταθμό ως κόμβο στον γράφο `G`.

1.2: Αποθήκευσε το όνομα του σταθμού `Station Name`, το γεωγραφικό μήκος και πλάτος `Longitude` και `Latitude` και την θέση `Position` ως ιδιότητες του κόμβου.

2: Τέλος επανάληψης

3: Για κάθε ζεύγος κόμβων `x, y` του γράφου `G` επανάλαβε:

3.1: Υπολόγισε την απόσταση `dist` μεταξύ των κόμβων, με τη συνάρτηση `distance`.

3.2: Εάν ισχύει $dist > 2.5$, τότε τερμάτισε την συγκεκριμένη επανάληψη.

3.3: Πρόσθεσε στον γράφο, την ακμή μεταξύ των δύο κόμβων με βάρος $\exp(-dist)$.

4: Τέλος επανάληψης.

5: Δημιούργησε τον πίνακα γειτνίασης `adj` από τον γράφο `G`.

6: Κανονικοποίησε τον πίνακα `adj` με τη συνάρτηση `adj_norm`.

7: Επέστρεψε τον γράφο `G` και τον κανονικοποιημένο πίνακα γειτνίασης `adjnorm`.

ΤΕΛΟΣ ΣΥΝΑΡΤΗΣΗΣ `getGraph`

3.5.4 Η Συνάρτηση `setupTGCN`

Η συγκεκριμένη συνάρτηση, προετοιμάζει τα δεδομένα για την εκπαίδευση και τη δοκιμή ενός μοντέλου T-GCN. Αρχικά, γίνεται ο κατάλληλος έλεγχος σχετικά με τον ορίζοντα πρόβλεψης `forecast_h`, ούτως ώστε αυτός να είναι 1, 7 ή 30 ημέρες. Στη συνέχεια, ορίζεται ο αριθμός των καθυστερημένων μεταβλητών `numlags` και με την κλήση της συνάρτησης `getGraph`, δημιουργείται ο γράφος `G` και ο πίνακας γειτνίασης `adj`. Όπως και στη συνάρτηση `build_rastermap` που αναλύθηκε νωρίτερα, έτσι και εδώ γίνεται ο υπολογισμός των πλήρων ημερών μεταξύ της πρώτης και της τελευταίας εγγραφής του συνόλου δεδομένων.

Έπειτα, αρχικοποιείται ο πίνακας χρονοσειρών `ts` στον οποίο θα αποθηκευτεί η ενεργειακή κατανάλωση του **κάθε σταθμού ανά ημέρα**. Τα δεδομένα κανονικοποιούνται διαιρώντας την κάθε τιμή με το μέγιστο της αντίστοιχης σειράς. Δημιουργείται ένας πίνακας `matrixlags`, ο οποίος περιέχει για κάθε κόμβο τις προηγούμενες τριάντα (30) ημέρες δεδομένων αλλά και τις επόμενες ημέρες που πρέπει να προβλέπουν. Όπως και τα δύο προηγούμενα μοντέλα, το μοντέλο T-GCN εκπαιδεύεται χρησιμοποιώντας δεδομένα από το 2012 έως το 2020, ενώ για την δοκιμή των μοντέλων στους ορίζοντες πρόβλεψης 1 και 7 ημερών γίνεται χρήση των τελευταίων 30 ημερών καταγραφών. Αντίστοιχα με τα προηγούμενα μοντέλα, η δοκιμή του T-GCN στη μακροπρόθεσμη πρόβλεψη 30 ημερών γίνεται στις τελευταίες 120 ημέρες δεδομένων. Τέλος, το σύνολο δοκιμής `y_test` αποκανονικοποιείται, πολλαπλασιάζοντας με τις αρχικές μέγιστες τιμές και επιστρέφονται ο γράφος `G`, ο (κανονικοποιημένος) πίνακας γειτνίασης `adj` και τα σύνολα εκπαίδευσης και δοκιμής.

ΕΝΑΡΞΗ ΣΥΝΑΡΤΗΣΗΣ `setupTGCN`

Είσοδοι Συνάρτησης: Σύνολο δεδομένων `data`, ορίζοντας πρόβλεψης `forecast_h`.

Έξοδοι Συνάρτησης: γράφος `G`, κανονικοποιημένος πίνακας γειτνίασης `adj`, σύνολα `X_train`, `y_train`, `X_test`, `y_test`.

0: Εάν η τιμή της μεταβλητής `forecast_h` δεν είναι 1, 7 ή 30, τότε:

0.1: Εμφάνισε κατάλληλο μήνυμα λάθους και διάκοψε την εκτέλεση του κώδικα.

1: Όρισε τη μεταβλητή `numlags = 30`.

2: Κάλεσε τη συνάρτηση `getGraph(data)` και αποθήκευσε τα αποτελέσματα στις μεταβλητές `G` και `adj`.

3: Υπολόγισε τον αριθμό πλήρων ημερών `num_h` ως το ακέραιο μέρος της διαφοράς ανάμεσα στην τελευταία και την πρώτη χρονική στιγμή σε δευτερόλεπτα, διαιρεμένο με το πλήθος των δευτερολέπτων σε μια ημέρα.

4: Αρχικοποίησε με μηδενικές τιμές έναν πίνακα χρονοσειρών `ts` με διαστάσεις:

$[len(G.nodes), num_h + 1]$.

5: Όρισε τη μεταβλητή `starttime` ως την ελάχιστη ημερομηνία του `data`.

6: Για κάθε ημέρα i από 0 έως `num_h` επανάλαβε:

6.1: Όρισε `window = starttime + timedelta(days=i)`.

6.2: Αποθήκευσε τα δεδομένα της τρέχουσας ημέρας στο dataframe `current`.

6.3: Για κάθε κόμβο n του `G` επανάλαβε (βρόγχος με μεταβλητή j):

6.3.1: Αποθήκευσε στη μεταβλητή `temp` τα δεδομένα του `current` για τον σταθμό n .

6.3.2: Αποθήκευσε το άθροισμα της στήλης `temp['Energy']` στον πίνακα `ts[j][i]`.

6.4: Τέλος εσωτερικής επανάληψης.

7: Τέλος επανάληψης.

8: Αντικατάστησε τις μηδενικές τιμές του `ts` με τυχαίες τιμές από κανονική κατανομή.

9: Κανονικοποίησε τον `ts` διαιρώντας κάθε στοιχείο του με το μέγιστο της αντίστοιχης χρονοσειράς.

10: Όρισε `num_nodes = len(G.nodes())` και αρχικοποίησε με μηδενικές τιμές τον πίνακα `matrixlags` με διαστάσεις:

$[ts.shape[-1]-(numlags+forecast_h), ts.shape[0], numlags+forecast_h]$.

11: Όρισε τη μεταβλητή `rng = matrixlags.shape[0]`.

- 12: Εάν `forecast_h` in `[1,7]`, όρισε `itrain = rng - 30`, αλλιώς όρισε `itrain = rng - 120`
- 13: Όρισε `itest = rng`.
- 14: Για κάθε i από 1 έως `rng` επανάλαβε:
- 14.1: Αποθήκευσε στο `matrixlags[i]`, τα δεδομένα `norm[:, i:i+numlags+forecast_h]`.
- 15: Τέλος επανάληψης
- 16: Αρχικοποίησε με μηδενικές τιμές τους τετραδιάστατους πίνακες `X_train`, `y_train`, `X_test` και `y_test` και όρισε τις παρακάτω διαστάσεις:
- 16.1: Για τον `X_train`, (`itrain`, `num_nodes`, `numlags`).
- 16.2: Για τον `y_train`, (`itrain`, `num_nodes`, `forecast_h`).
- 16.2: Για τον `X_test`, (`itest - itrain`, `num_nodes`, `numlags`).
- 16.3: Για τον `y_test`, (`itest - itrain`, `num_nodes`, `forecast_h`).
- 17: Για κάθε κόμβο n στον γράφο G (βρόγχος με μεταβλητή i) επανάλαβε:
- 17.1: Αποθήκευσε τα δεδομένα των τελευταίων `itrain` εγγραφών του πίνακα `matrixlags` στο `X_train` και `y_train`.
- 17.2: Αποθήκευσε τα δεδομένα των τελευταίων `itest` εγγραφών του πίνακα `matrixlags` στο `X_test` και `y_test`.
- 18: Τέλος επανάληψης.
- 19: Αποκανονικοποίησε το `y_test` πολλαπλασιάζοντας με τις αρχικές μέγιστες τιμές και αποθήκευσέ το στη μεταβλητή `y_test_denorm`.
- 20: Εκτύπωσε τις διαστάσεις των `X_train`, `y_train`, `X_test`, `y_test_denorm`.
- 21: Επίστρεψε τα `G`, `adj`, `X_train`, `y_train`, `X_test`, `y_test_denorm`, `max_`.
- ΤΕΛΟΣ ΣΥΝΑΡΤΗΣΗΣ `setupTGCN`**

3.6 Εκπαίδευση και Αξιολόγηση των Μοντέλων

Στην παρούσα μεθοδολογία, αφού έχει γίνει η προαπαιτούμενη διάσπαση του συνόλου δεδομένων σε σύνολα εκπαίδευσης και δοκιμής, προχωράμε στη δημιουργία των μοντέλων που προαναφέρθηκαν (CNN, ConvLSTM, T-GCN).

Στη συνέχεια αυτού του κεφαλαίου περιγράφονται αναλυτικά οι υλοποιήσεις για κάθε ένα ξεχωριστό μοντέλο. Όλα τα μοντέλα εκπαιδεύτηκαν χρησιμοποιώντας εκατό (100) εποχές εκπαίδευσης (epochs), η βελτιστοποίηση του μοντέλου γίνεται με τη χρήση του ADAM βελτιστοποιητή (optimizer), ενώ ως μετρική υπολογισμού σφάλματος χρησιμοποιούν το μέσο απόλυτο σφάλμα (Mean Absolute Error - MAE) που περιγράφεται στην παρακάτω σχέση 3.1:

$$\frac{1}{n} * \sum_{i=1}^n |y_i - k| \quad (3.1)$$

όπου:

- n , ο αριθμός των παρατηρήσεων
- y_i , η πραγματική τιμή
- k , η πρόβλεψη του μοντέλου για την i -οστή παρατήρηση

3.6.1 Τα Μοντέλα προς Εκπαίδευση

Το μοντέλο CNN που εκπαιδεύουμε είναι ένα μοντέλο συνελκτικών νευρωνικών δικτύων δύο διαστάσεων. Χρησιμοποιεί, 16 φίλτρα (filters) για την ανίχνευση των χαρακτηριστικών στην είσοδο, όπου το κάθε φίλτρο σαρώνει περιοχές 3x3 της εικόνας. Τέλος, χρησιμοποιείται η συνάρτηση ενεργοποίησης ReLU (Rectified Linear Unit), η οποία εισάγει μη-γραμμικότητα στο μοντέλο.

Όσον αφορά το μοντέλο ConvLSTM, γίνεται χρήση στρωμάτων TimeDistributed, που επιτρέπουν την εφαρμογή των συνελκτικών στρωμάτων σε κάθε χρονική στιγμή ξεχωριστά σε μια ακολουθία δεδομένων. Τα συνελκτικά στρώματα είναι ίδια με αυτά του CNN (16 φίλτρα με μέγεθος 3x3), ενώ παράλληλα προστίθενται ένα στρώμα LSTM με πενήντα (50) νευρώνες που χρησιμοποιείται για την επεξεργασία χρονοσειρών.

Τέλος, το μοντέλο T-GCN κάνει χρήση δύο στρωμάτων GraphConv με δεκαέξι (16) και δέκα (10) φίλτρα, αντίστοιχα. Και στα δύο στρώματα γίνεται χρήση του κανονικοποιημένου πίνακα γειτνίασης adj που έχει δημιουργηθεί κατά τη διάσπαση του συνόλου δεδομένων σε σύνολα εκπαίδευσης και δοκιμής. Επίσης, γίνεται χρήση ενός στρώματος LSTM με πενήντα (50) νευρώνες, όπως και στο μοντέλο ConvLSTM.

3.6.2 Αξιολόγηση με την Μετρική RMSE

Η αξιολόγηση των μοντέλων γίνεται πάνω σε **ολόκληρο** το σύστημα, δηλαδή πάνω στο άθροισμα ολόκληρης της απαιτούμενης ενέργειας για τη συγκεκριμένη περιοχή.

Για τα μοντέλα CNN και ConvLSTM, όταν γίνεται αναφορά σε “ολόκληρο σύστημα”, εννοούμε το άθροισμα για κάθε κελί του raster map. Για το μοντέλο T-GCN, η αξιολόγηση πάνω σε ολόκληρο το σύστημα αναφέρεται στο άθροισμα των προβλέψεων για κάθε σταθμό φόρτισης.

Η αξιολόγηση γίνεται χρησιμοποιώντας τη μετρική του ριζικού μέσου τετραγωνικού λάθους (Root Mean Squared Error - RMSE), ο τύπος του οποίου αποτυπώνεται στην παρακάτω σχέση 3.2:

$$\sqrt{\sum_{i=1}^n (y_i - k)^2} \quad (3.2)$$

όπου:

- n , ο αριθμός των παρατηρήσεων
- y_i , η πραγματική τιμή
- k , η πρόβλεψη του μοντέλου για την i -οστή παρατήρηση

4. Εφαρμογή

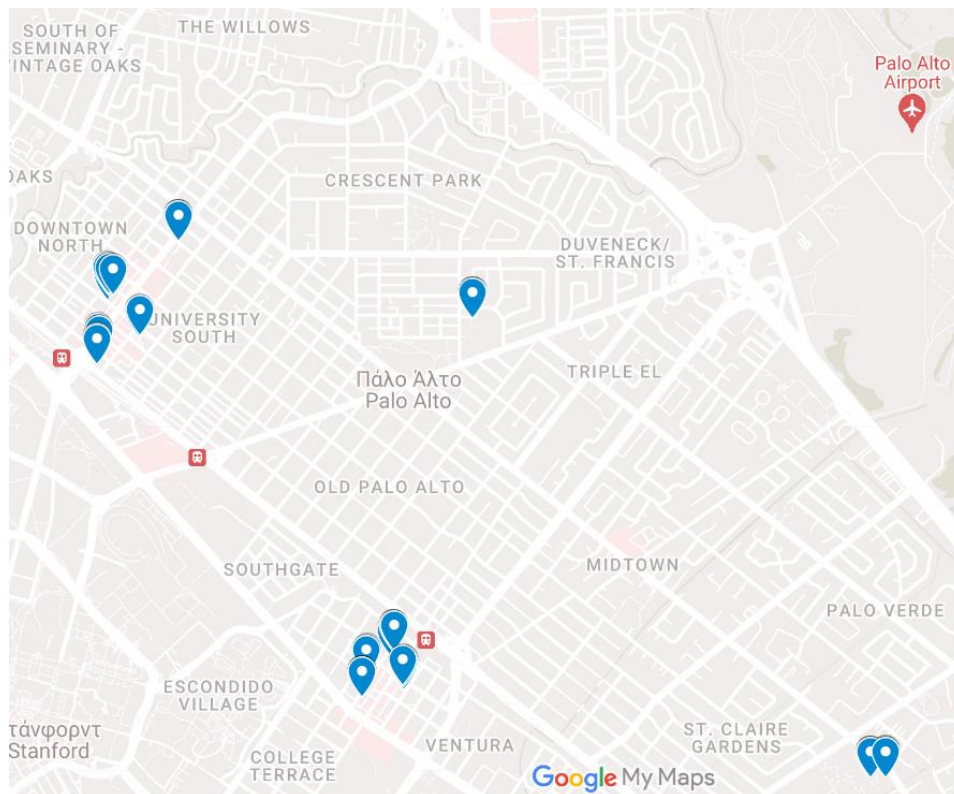
Στο συγκεκριμένο κεφάλαιο της εργασίας θα γίνει η επεξήγηση του κώδικα που αναπτύχθηκε για την εφαρμογή της μεθοδολογίας, όπως παρουσιάστηκε στο προηγούμενο κεφάλαιο. Ο κώδικας είναι χωρισμένος σε δύο αρχεία Juryter Notebook [18] με τους τίτλους “0-main-conv” και “0-main-tgcn”.

Το πρώτο αρχείο περιέχει ολόκληρη την υλοποίηση των μοντέλων CNN και ConvLSTM, από το στάδιο της προεπεξεργασίας των δεδομένων μέχρι και την εκπαίδευση και αξιολόγηση των μοντέλων.

Αντίστοιχα, στο δεύτερο αρχείο, πραγματοποιείται η ανάλογη διαδικασία για το μοντέλο T-GCN.

4.1 Σύνολο Δεδομένων

Πρόκειται για ένα ανοικτό σύνολο δεδομένων, που αφορά στην ενεργειακή κατανάλωση από το δίκτυο των φορτιστών της πόλης του Palo Alto, CA για το χρονικό διάστημα Ιούλιος 2011 έως Δεκέμβριος 2020 [16].



Εικόνα 9 | Το Δίκτυο Φορτιστών του Palo Alto. Πηγή: Google Maps

Το αρχείο δεδομένων αποτελείται από τις καταγεγραμμένες περιόδους φόρτισης σε ένα δίκτυο 47 σταθμών φόρτισης. Υπάρχουν πληροφορίες, όπως το όνομα του σταθμού φόρτισης, οι χρονοσφραγίδες (timestamp) για την έναρξη και τη λήξη της φόρτισης (και οι αντίστοιχες ζώνες ώρας), ο χρόνος που ήταν συνδεδεμένο το όχημα στον φορτιστή, καθώς και ο χρόνος που φόρτιζε η μπαταρία. Επίσης, υπάρχουν πληροφορίες για τον κάθε σταθμό φόρτισης, όπως το είδος της θύρας (Port Type), ο αριθμός της θύρας (Port Number), το είδος του φορτιστή (Plug Type) και οι γεωγραφικές συντεταγμένες (γεωγραφικό μήκος και πλάτος).

Επίσης, υπάρχουν αποθηκευμένες πληροφορίες που αφορούν τον χρήστη, όπως η κατανάλωση της ενέργειας που έγινε κατά τη διάρκεια της φόρτισης (σε kWh), το κόστος (Fee), σε τι νόμισμα έγινε η συναλλαγή (Currency), καθώς και πληροφορίες σχετικά με τον χρήστη (στη συγκεκριμένη περίπτωση οδηγό του οχήματος), όπως ο αριθμός χρήστη (User ID) και ο ταχυδρομικός κωδικός που έχει δηλώσει ο χρήστης (Driver Postal Code).

	Station Name	MAC Address	Org Name	Start Date	Start Time Zone	End Date	End Time Zone	Transaction Date (Pacific Time)	Total Duration (hh:mm:ss)	Charging Time (hh:mm:ss)	...	Longitude
0	PALO ALTO CA / HAMILTON #1	000D:6F00:015A:9D76	City of Palo Alto	7/29/2011 20:17	PDT	7/29/2011 23:20	PDT	7/29/2011 23:20	3:03:32	1:54:03	...	-122.160309
1	PALO ALTO CA / HAMILTON #1	000D:6F00:015A:9D76	City of Palo Alto	7/30/2011 0:00	PDT	7/30/2011 0:02	PDT	7/30/2011 0:02	0:02:06	0:01:54	...	-122.160309
2	PALO ALTO CA / HAMILTON #1	000D:6F00:015A:9D76	City of Palo Alto	7/30/2011 8:16	PDT	7/30/2011 12:34	PDT	7/30/2011 12:34	4:17:32	4:17:28	...	-122.160309
3	PALO ALTO CA / HAMILTON #1	000D:6F00:015A:9D76	City of Palo Alto	7/30/2011 14:51	PDT	7/30/2011 16:55	PDT	7/30/2011 16:55	2:03:24	2:02:58	...	-122.160309
4	PALO ALTO CA / HAMILTON #1	000D:6F00:015A:9D76	City of Palo Alto	7/30/2011 18:51	PDT	7/30/2011 20:03	PDT	7/30/2011 20:03	1:11:24	0:43:54	...	-122.160309

5 rows x 33 columns

Εικόνα 10 | Αρχικό σύνολο δεδομένων

4.2 Προεπεξεργασία Δεδομένων για τα Μοντέλα CNN και ConVLSTM

Για την εκπαίδευση των μοντέλων μηχανικής μάθησης θα γίνει χρήση μόλις πέντε από το σύνολο των χαρακτηριστικών που περιέχει το αρχικό σύνολο δεδομένων. Συγκεκριμένα:

- **Start Date:** Η χρονοσφραγίδα της έναρξης της κάθε περιόδου φόρτισης.
- **Station Name:** Το όνομα του σταθμού φόρτισης για κάθε περίοδο φόρτισης.
- **Energy:** Η κατανάλωση ενέργειας (σε kWh) για κάθε περίοδο φόρτισης.
- **Latitude:** Το γεωγραφικό πλάτος (Latitude) κάθε φορτιστή.
- **Longitude:** Το γεωγραφικό μήκος (Longitude) κάθε φορτιστή.

Υπενθυμίζεται ότι στόχος της εργασίας είναι η εκπαίδευση των μοντέλων για την πρόβλεψη απαιτούμενης ενέργειας, οπότε χαρακτηριστικά μεταδεδομένων (metadata) που αφορούν τους σταθμούς φόρτισης, όπως το είδος και ο αριθμός της θύρας ή το κόστος φόρτισης, δεν είναι χρήσιμα στην εφαρμογή της συγκεκριμένης μεθοδολογίας.

	Station Name	Energy	Longitude	Latitude
Start Date				
2012-01-01	PALO ALTO CA / HAMILTON #1	7.215172	-122.160309	37.444572
2012-01-01	PALO ALTO CA / BRYANT #2	4.670387	-122.162140	37.446510
2012-01-01	PALO ALTO CA / HAMILTON #2	6.042525	-122.160263	37.444607
2012-01-01	PALO ALTO CA / HAMILTON #1	11.716562	-122.160309	37.444572
2012-01-02	PALO ALTO CA / HIGH #4	3.984439	-122.162880	37.443676
...
2020-12-31	PALO ALTO CA / TED THOMPSON #3	1.841000	-122.144043	37.428623
2020-12-31	PALO ALTO CA / WEBSTER #1	12.744000	-122.158272	37.448284
2020-12-31	PALO ALTO CA / MPL #4	29.665000	-122.113510	37.422104
2020-12-31	PALO ALTO CA / WEBSTER #1	11.976000	-122.158272	37.448284
2020-12-31	PALO ALTO CA / CAMBRIDGE #4	1.571000	-122.146034	37.426178

Εικόνα 11 | Χαρακτηριστικά που θα χρησιμοποιηθούν στην έρευνα

4.2.1 Εφαρμογή Raster Map

Για την υλοποίηση των μοντέλων CNN και ConVLSTM που αναλύθηκαν στο προηγούμενο κεφάλαιο, είναι απαραίτητη η υλοποίηση και η εφαρμογή του raster map. Αυτό επιτυγχάνεται με την κλήση της συνάρτησης `getcell`, η οποία επιστρέφει το ενημερωμένο σύνολο δεδομένων, έχοντας προσθέσει δυο νέες στήλες.



Εικόνα 12 | Raster Map

	Station Name	Energy	Longitude	Latitude	xcell	ycell
Start Date						
2012-01-01	PALO ALTO CA / HAMILTON #1	7.215172	-122.160309	37.444572	0	4
2012-01-01	PALO ALTO CA / BRYANT #2	4.670387	-122.162140	37.446510	0	4
2012-01-01	PALO ALTO CA / HAMILTON #2	6.042525	-122.160263	37.444607	0	4
2012-01-01	PALO ALTO CA / HAMILTON #1	11.716562	-122.160309	37.444572	0	4
2012-01-02	PALO ALTO CA / HIGH #4	3.984439	-122.162880	37.443676	0	3
...
2020-12-31	PALO ALTO CA / TED THOMPSON #3	1.841000	-122.144043	37.428623	1	1
2020-12-31	PALO ALTO CA / WEBSTER #1	12.744000	-122.158272	37.448284	0	4
2020-12-31	PALO ALTO CA / MPL #4	29.665000	-122.113510	37.422104	4	0
2020-12-31	PALO ALTO CA / WEBSTER #1	11.976000	-122.158272	37.448284	0	4
2020-12-31	PALO ALTO CA / CAMBRIDGE #4	1.571000	-122.146034	37.426178	1	0

Εικόνα 13 | Ενημερωμένο Σύνολο Δεδομένων

Οι στήλες `xcell` και `ycell`, αντικατοπτρίζουν τη θέση του κάθε σταθμού πάνω στον raster map. Για παράδειγμα, ο σταθμός με όνομα **PALO ALTO CA / HAMILTON #1**, τοποθετείται στο τελευταίο κελί της πρώτης στήλης του πλέγματος (αφού `xcell=0` και `ycell=4`).

Για τη δημιουργία των raster map, χρησιμοποιείται η συνάρτηση `build_rastermap`, η ανάλυση της οποίας έγινε στο προηγούμενο κεφάλαιο. Κατά την κλήση της συγκεκριμένης συνάρτησης, δημιουργείται ένας πίνακας διαστάσεων 5 x 5 για κάθε ημέρα καταγραφών του συνόλου δεδομένων. Σε κάθε κελί του πίνακα, αποθηκεύεται το συνολικό ποσό ενέργειας που καταναλώνεται από τους σταθμούς που αντιστοιχούν γεωγραφικά στο ίδιο κελί

(χαρακτηριστικά x_{cell} και y_{cell}). Η συνάρτηση επιστρέφει έναν πίνακα τριών διαστάσεων, $3288 \times 5 \times 5$, όπου το 3288 αντιστοιχεί στο πλήθος των ημερών του συνόλου δεδομένων για τις οποίες έχει δημιουργηθεί ο αντίστοιχος raster map διαστάσεων 5×5 .

4.2.2 Διαχωρισμός σε Σύνολα Εκπαίδευσης και Δοκιμής των Μοντέλων CNN και ConvLSTM

Σε αυτό το βήμα γίνεται η διάσπαση του συνόλου δεδομένων σε δύο σύνολα: το σύνολο εκπαίδευσης και το σύνολο δοκιμής. Το πρώτο αποτελείται από τα δεδομένα που θα χρησιμοποιηθούν για την εκπαίδευση των δύο μοντέλων, ενώ το δεύτερο για την αξιολόγηση των μοντέλων αυτών.

Όπως αναλύθηκε στο κεφάλαιο της μεθοδολογίας, για τη δοκιμή των μοντέλων χρησιμοποιούνται οι τελευταίες 30 ημέρες εγγραφών για την πρόβλεψη 1 και 7 ημερών, ενώ για την πρόβλεψη 30 ημερών χρησιμοποιούνται οι τελευταίες 120 ημέρες εγγραφών. Η εκπαίδευση των μοντέλων για όλους τους χρονικούς ορίζοντες γίνεται με τη χρήση δεδομένων από το 2012 έως το 2020.

Ο διαχωρισμός γίνεται με τη χρήση των raster map που έχουν δημιουργηθεί, που, πλέον, περιέχουν κανονικοποιημένες τιμές ενέργειας. Τα σύνολα εκπαίδευσης X_{train} και y_{train} που έχουμε πλέον στη διάθεσή μας αποτελούνται από τις κανονικοποιημένες αυτές τιμές, ενώ το y_{test} αποκανονικοποιείται με σκοπό τη δοκιμή των μοντέλων σε πραγματικές τιμές. Τα τελικά σύνολα εκπαίδευσης και δοκιμής που επιστρέφονται έχουν τη μορφή πλειάδων.

Συγκεκριμένα, για το CNN, η πρώτη διάσταση των train και test set συμβολίζει τα ημερήσια δείγματα δεδομένων που χρησιμοποιούνται για την εκπαίδευση και δοκιμή. Αντίστοιχα, η δεύτερη και η τρίτη διάσταση αντιπροσωπεύουν τις διαστάσεις του raster map, υποδεικνύοντας πως τα δεδομένα είναι κατανομημένα σε ένα πλέγμα διαστάσεων 5×5 , ενώ η τέταρτη και τελευταία διάσταση αναφέρεται στον αριθμό των lagged values που χρησιμοποιούνται για τα σύνολα εκπαίδευσης και στον οριζοντα πρόβλεψης στον οποίο εφαρμόζεται το μοντέλο για τα σύνολα δοκιμής.

Ακολούθως, για το μοντέλο ConvLSTM, τα σύνολα δοκιμής X_{test} και y_{test} ακολουθούν ακριβώς την ίδια δομή με αυτή του μοντέλου CNN. Τα μοντέλα εκπαίδευσης X_{train} , y_{train} είναι πλειάδες 5 διαστάσεων. Οι πρώτες 4 διαστάσεις ακολουθούν παρόμοια δομή με τα ημερήσια δείγματα δεδομένων. Στην πρώτη, τον αριθμό lagged values για τα σύνολα εκπαίδευσης και τον οριζοντα πρόβλεψης για τα σύνολα δοκιμής στη δεύτερη και τις διαστάσεις του raster map στην τρίτη και τέταρτη διάσταση, ενώ στο τέλος υπάρχει μια πέμπτη διάσταση, η οποία είναι ένα κανάλι χαρακτηριστικών (feature channel) που αντιπροσωπεύει ένα μοναδικό χαρακτηριστικό για κάθε κελί του πλέγματος.

4.3 Προεπεξεργασία Δεδομένων για το Μοντέλο T-GCN

4.3.1 Δημιουργία Γράφου και Πίνακα Γειτνίασης

Στο συγκεκριμένο μοντέλο, χρησιμοποιείται ένας γράφος 47 κόμβων (όσοι και οι σταθμοί φόρτισης), μερικοί από τους οποίους συνδέονται μεταξύ τους με τις αντίστοιχες ακμές συγκεκριμένου βάρους. Πιο συγκεκριμένα, οι κόμβοι συνδέονται μεταξύ τους σε περίπτωση που η μεταξύ τους απόσταση δεν ξεπερνάει τα 2.5 χιλιόμετρα. Η απόσταση αυτή υπολογίζεται από τη φόρμουλα Haversine, και ως βάρος ακμής ορίζουμε το αρνητικό εκθετικό της απόστασης αυτής.

Μαζί με τον γράφο δημιουργείται και ο πίνακας γειτνίασης, που μετέπειτα χρησιμοποιείται για τον διαχωρισμό του συνόλου δεδομένων σε υποσύνολα εκπαίδευσης και δοκιμής.

4.3.2 Διαχωρισμός σε Σύνολα Εκπαίδευσης και Δοκιμής του Μοντέλου T-GCN

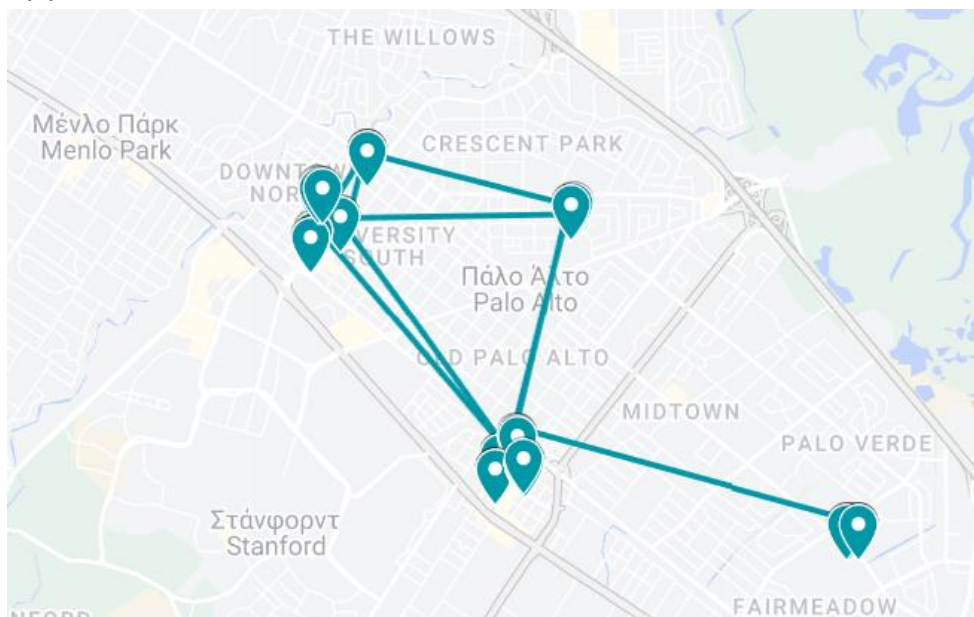
Ο διαχωρισμός των δεδομένων σε σύνολα εκπαίδευσης και δοκιμής γίνεται, όπως προαναφέρθηκε, με τη χρήση του γράφου G, που επιστρέφεται από την κλήση της συνάρτησης getGraph. Γίνεται η αποθήκευση της συνολικής ενεργειακής κατανάλωσης κάθε σταθμού σε

έναν πίνακα, ο οποίος στη συνέχεια κανονικοποιείται για να γίνει ο διαχωρισμός. Το τελικό γράφημα που δημιουργείται αποτελείται από 47 κόμβους. Στην Εικόνα 14, αναπαρίσταται ο γράφος.

Όπως και στα προηγούμενα μοντέλα μηχανικής μάθησης, έτσι και εδώ για τη δοκιμή των μοντέλων χρησιμοποιούνται οι τελευταίες 30 ημέρες εγγραφών για την πρόβλεψη 1 και 7 ημερών, ενώ για την πρόβλεψη 30 ημερών χρησιμοποιούνται οι τελευταίες 120 ημέρες εγγραφών. Η εκπαίδευση των μοντέλων για όλους τους χρονικούς ορίζοντες γίνεται με τη χρήση δεδομένων από το 2012 έως το 2020.

Και σε αυτή τη διάσπαση γίνεται αποκανονικοποίηση του συνόλου δοκιμής πριν αυτό επιστραφεί από τη συνάρτηση, ούτως ώστε η αξιολόγηση να γίνει πάνω σε πραγματικές τιμές καταναλωμένης ενέργειας.

Όπως και στα άλλα μοντέλα μηχανικής μάθησης, έτσι και στο T-GCN τα σύνολα εκπαίδευσης και δοκιμής επιστρέφονται με τη μορφή πλειάδων. Πρόκειται για πλειάδες τριών διαστάσεων, με την πρώτη να συμβολίζει τα ημερήσια δείγματα δεδομένων που χρησιμοποιούνται για εκπαίδευση και δοκιμή αντίστοιχα, τη δεύτερη να αντιστοιχεί στον αριθμό κόμβων του γράφου και, τέλος, την τρίτη διάσταση που στα σύνολα εκπαίδευσης αντιστοιχεί στον αριθμό lagged values που χρησιμοποιούνται και στα σύνολα δοκιμής στον οριζοντα πρόβλεψης.



Εικόνα 14 | Αναπαράσταση του γράφου G στο δίκτυο.

4.4 Εκπαίδευση των Μοντέλων

Και τα τρία μοντέλα εκπαιδεύονται στο ίδιο σύνολο δεδομένων χρησιμοποιώντας τον ίδιο αριθμό καταγραφών για εκπαίδευση, καθώς επίσης και τα ίδια χαρακτηριστικά. Σε κάθε περίπτωση, το προς πρόβλεψη χαρακτηριστικό (feature) είναι η απαιτούμενη ενέργεια.

Πιο αναλυτικά, όσον αφορά τις παραμέτρους για όλα τα μοντέλα:

- Τα μοντέλα CNN και ConvLSTM χρησιμοποιούν τον ίδιο αριθμό φίλτρων (16), με το κάθε φίλτρο να είναι μεγέθους 3x3.
- Τα μοντέλα ConvLSTM και T-GCN χρησιμοποιούν τον ίδιο αριθμό νευρώνων LSTM (50).
- Και τα τρία μοντέλα χρησιμοποιούν τον βελτιστοποιητή ADAM, τη συνάρτηση ενεργοποίησης ReLU και τη μετρική υπολογισμού σφάλματος MAE.
- Στην εκπαίδευση των τριών μοντέλων υπάρχει πλήθος καθυστερημένων μεταβλητών, που ισούται με 30.
- Η εκπαίδευση των μοντέλων έγινε σε εκατό εποχές (epochs).

4.5 Αξιολόγηση των Μοντέλων

Μετά την εκπαίδευση των μοντέλων ακολουθεί η δοκιμή τους πάνω στα κατάλληλα σύνολα δοκιμής. Οι προβλέψεις που επιστρέφονται από όλα τα μοντέλα, είναι στη μορφή πλειάδας. Η πλειάδα αυτή είναι 4 διαστάσεων για τα μοντέλα CNN και ConvLSTM, και 3 διαστάσεων για το T-GCN. Η δομή της πλειάδας για κάθε μοντέλο είναι ίδια με αυτή του αντίστοιχου συνόλου δοκιμής για το κάθε μοντέλο, όπως αναλύθηκε σε προηγούμενη ενότητα.

Η αξιολόγηση των μοντέλων έγινε στο **συνολικό σύστημα**. Συγκεκριμένα, για τα μοντέλα CNN και ConvLSTM, συνολικό σύστημα είναι το άθροισμα των προβλέψεων όλων των κελιών του raster map. Για το μοντέλο T-GCN, ως συνολικό σύστημα αναφερόμαστε στο άθροισμα των προβλέψεων όλων των σταθμών του δικτύου.

Η μετρική που χρησιμοποιήθηκε για την αξιολόγηση των μοντέλων είναι το RMSE. Συγκεκριμένα, για κάθε μοντέλο σε κάθε ορίζοντα πρόβλεψης τα μοντέλα είχαν τις παρακάτω επιδόσεις (Πίνακας 4.1). Υπενθυμίζεται ότι όσο πιο μικρή είναι η τιμή μιας μετρικής, τόσο καλύτερη είναι η επίδοση του μοντέλου.

Μοντέλο	Πρόβλεψη 1 Ημέρας	Πρόβλεψη 7 Ημερών	Πρόβλεψη 30 ημερών
CNN	122	143	205
ConvLSTM	120	140	180
T-GCN	116	136	108

Πίνακας 4.1: RMSE προβλέψεων για το συνολικό σύστημα

Η αξιολόγηση των μοντέλων έγινε και σε επίπεδο σταθμών, συγκρίνοντας πάλι τις επιδόσεις RMSE των μοντέλων (Πίνακας 4.2). Αυτή τη φορά, χρησιμοποιήθηκε το άθροισμα των προβλέψεων για κάθε κελί του raster map που χρησιμοποιείται στα μοντέλα CNN και ConvLSTM. Για το μοντέλο T-GCN, χρησιμοποιήθηκε το άθροισμα των προβλέψεων για κάθε κόμβο του γράφου.

Μοντέλο	Πρόβλεψη 1 Ημέρας	Πρόβλεψη 7 Ημερών	Πρόβλεψη 30 ημερών
CNN	19.2	70.2	282.1
ConvLSTM	20.1	141.5	478.9
T-GCN	18.9	106.2	384.2

Πίνακας 4.2: RMSE προβλέψεων για κάθε grid cell / σταθμό

4.6 Αποτελέσματα

Από τον πίνακα 4.1 συμπεραίνουμε ότι το μοντέλο T-GCN αποδίδει καλύτερα και στους 3 χρονικούς ορίζοντες πρόβλεψης. Για τη βραχυπρόθεσμη πρόβλεψη 1 ημέρας, οι αποκλίσεις των μοντέλων ως προς την απόδοσή τους είναι η μικρότερη συγκριτικά με τη μεσοπρόθεσμη και μακροπρόθεσμη πρόβλεψη, με το μοντέλο T-GCN να εμφανίζει το χαμηλότερο RMSE. Ένα τέτοιο αποτέλεσμα είναι αναμενόμενο, καθώς το μοντέλο CNN χρησιμοποιείται κατεξοχήν στην ανάλυση χωρικών δεδομένων χωρίς χρονικές πληροφορίες, ενώ το T-GCN, όπως αναφέρθηκε προηγουμένως, είναι ιδανικό για τις προβλέψεις δεδομένων που περιέχουν χωρικές και χρονικές πληροφορίες, όπως, εν προκειμένω, τα δεδομένα που χρησιμοποιούμε.

Ομοίως, παρατηρώντας τα αποτελέσματα για τη μεσοπρόθεσμη πρόβλεψη 7 ημερών, βλέπουμε πως το T-GCN εμφανίζει πάλι την καλύτερη επίδοση σε σχέση με τα άλλα δύο

μοντέλα. Και τα τρία μοντέλα εμφανίζουν μικρές αποκλίσεις στις επιδόσεις τους, όμως, βλέποντας το παρακάτω διάγραμμα που περιγράφει την εβδομαδιαία πρόβλεψη (Διάγραμμα 4.1), μπορούμε να συμπεράνουμε ότι το μοντέλο T-GCN λειτουργεί καλύτερα από τα άλλα δύο, αφού το CNN εμφανίζει μια υπερτίμηση στις προβλέψεις του, ενώ το ConvlSTM υποεκτιμά σταθερά τις πραγματικές τιμές.

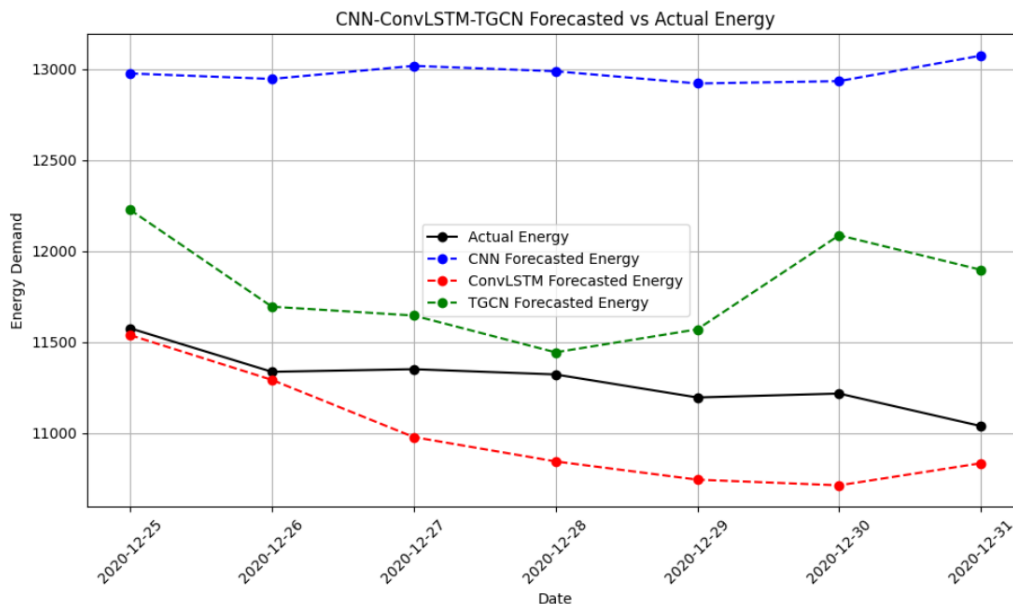
Όσον αφορά τη μακροπρόθεσμη πρόβλεψη των 30 ημερών, το μοντέλο T-GCN εξακολουθεί να έχει την καλύτερη επίδοση, όμως αυτή τη φορά οι επιδόσεις των τριών μοντέλων δεν έχουν την ίδια μικρή απόκλιση που είχαν στην πρόβλεψη 7 ημερών.

Επιπροσθέτως, παρατηρώντας το παρακάτω διάγραμμα που απεικονίζει τις προβλέψεις των μοντέλων μακροπρόθεσμα (Διάγραμμα 4.2), επιβεβαιώνεται ότι το μοντέλο T-GCN αποδίδει καλύτερα, εμφανίζοντας μια σταθερή πορεία προβλέψεων κοντά στις πραγματικές τιμές. Αντιθέτως, τα μοντέλα CNN και ConvlSTM εμφανίζουν μια αστάθεια στις προβλέψεις τους, οι οποίες υπολείπονται των πραγματικών τιμών κατανάλωσης.

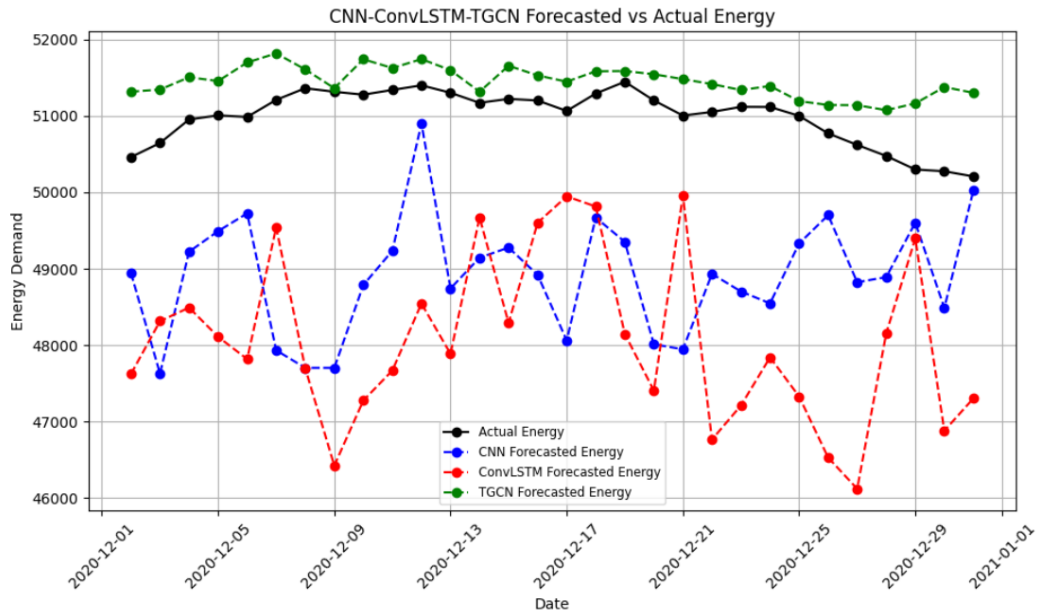
Από τη σταθερή πρόβλεψη χαμηλότερων των πραγματικών τιμών στα CNN και ConvlSTM, μπορούμε να συμπεράνουμε ότι τα δίκτυα GCN που χρησιμοποιούνται στο T-GCN μοντέλο λειτουργούν καλύτερα από τις συνεκτικές στοιβάδες των άλλων μοντέλων στην ανάλυση και αξιοποίηση της χωρικής πληροφορίας των δεδομένων, μιας και τα ConvlSTM και T-GCN χρησιμοποιούν το ίδιο στρώμα LSTM.

Τέλος, παρατηρώντας τα αποτελέσματα του πίνακα 4.2, που αφορούν στις προβλέψεις για κάθε σταθμό, βλέπουμε ότι στην πρόβλεψη της 1 ημέρας και τα τρία μοντέλα αποδίδουν το ίδιο, εμφανίζοντας ελάχιστες αποκλίσεις, ενώ στη μεσοπρόθεσμη και μακροπρόθεσμη πρόβλεψη των 7 και 30 ημερών, αντίστοιχα, το μοντέλο CNN, (το μοναδικό που δεν περιέχει στρώμα LSTM για τις χρονικές εξαρτήσεις) σημειώνει τις καλύτερες επιδόσεις.

Παρακάτω, υπάρχουν κάποια διαγράμματα που δημιουργήθηκαν κατά τη διάρκεια της εκπόνησης της εργασίας και αφορούν τις προβλέψεις στο **συνολικό σύστημα** (Πίνακας 4.1). Τα διαγράμματα αυτά υλοποιήθηκαν με τη χρήση της βιβλιοθήκης Matplotlib [19]:



Διάγραμμα 4.1: Αποτελέσματα για τις τελευταίες 7 ημέρες δεδομένων.



Διάγραμμα 4.2: Αποτελέσματα για τις τελευταίες 30 ημέρες δεδομένων.

4.7 Σύγκριση με αντίστοιχη έρευνα

Στο πλαίσιο της αναζήτησης σχετικής state-of-the-art βιβλιογραφίας που να έχει εμβαθύνει στο συγκεκριμένο πρόβλημα, εντοπίσαμε αρκετές έρευνες που χρησιμοποιούν GCN δίκτυα για την πρόβλεψη ενεργειακής κατανάλωσης. Μία από αυτές της έρευνες είναι και αυτή των Hüttel, F.B, Peled, I., Rodrigues, F., Pereira, F.C, 2020 [16].

Στη συγκεκριμένη έρευνα οι συγγραφείς χρησιμοποιούν το ίδιο σύνολο δεδομένων, αυτό της πόλης του Palo Alto (CA). Στη μελέτη των Peled et. al. (2020), η υλοποίηση του μοντέλου γραφικών συνελκτικών δικτύων με στρώματα LSTM γίνεται με τη χρήση της βιβλιοθήκης StellarGraph [22]. Το συγκεκριμένο μοντέλο T-GCN, όπως και το T-GCN της παρούσας εργασίας, κάνει χρήση 2 στρωμάτων GCN με 16 και 10 φίλτρα, αντίστοιχα, όπως και ενός στρώματος LSTM με 50 νευρώνες. Το συγκεκριμένο μοντέλο που ανέπτυξαν οι συγγραφείς εκπαιδεύτηκε χρησιμοποιώντας το ίδιο σύνολο δεδομένων, αυτό της πόλης του Πάλο Άλτο -το οποίο έχουμε χρησιμοποιήσει και εμείς στο πλαίσιο της παρούσας εργασίας- για τις καταγραφές των ετών 2012 έως 2019. Αυτή η διαφορά αναφορικά με το έτος 2020 μεταφράζεται σε 14 λιγότερους σταθμούς φόρτισης, και, κατα συνέπεια, 14 λιγότερους κόμβους στο γράφημα που δημιουργείται για το μοντέλο της έρευνας, συγκριτικά με την υλοποίηση αυτής της εργασίας. Στον παρακάτω πίνακα 4.3 φαίνεται η σύγκριση των αποτελεσμάτων των 2 μοντέλων.

Μοντέλο	Πρόβλεψη 1 Ημέρας	Πρόβλεψη 7 Ημερών	Πρόβλεψη 30 ημερών
T-GCN (StellarGraph)	61 ± 8	184 ± 9	161 ± 15
T-GCN (Thesis)	116	136	108

Πίνακας 4.3: Σύγκριση RMSE επιδόσεων των 2 μοντέλων T-GCN στο συνολικό σύστημα.

Το μοντέλο T-GCN χρησιμοποιεί μια προσαρμοσμένη υλοποίηση των δικτύων GCN. Σε αυτή την υλοποίηση, η κανονικοποίηση του πίνακα γίνεται χειροκίνητα, προσθέτοντας self-loops και υπολογίζοντας τον πίνακα βαθμών (degree matrix). Τόσο η αρχικοποίηση των βαρών με τη μέθοδο Glorot Uniform, όσο και η L2 κανονικοποίηση, παρέχουν τη δυνατότητα πειραματισμού για τη βελτίωση του μοντέλου. Η ευελιξία αυτή οδηγεί στην καλύτερη

εκπαίδευση του μοντέλου για μακροπρόθεσμες προβλέψεις, όπως η πρόβλεψη των 30 ημερών. Από την άλλη, το GCN της βιβλιοθήκης StellarGraph που χρησιμοποιείται στο μοντέλο T-GCN της έρευνας, αποτυπώνει μια πιο κλιμακούμενη προσέγγιση, η οποία, όπως φαίνεται, αποδίδει καλύτερα στις βραχυπρόθεσμες προβλέψεις.

5. Συμπεράσματα - Μελλοντικές Εργασίες

5.1 Συμπεράσματα

Ο φιλικός προς το περιβάλλον τρόπος ζωής που προσπαθεί να εφαρμόσει τα τελευταία χρόνια ο άνθρωπος, επηρεάζει σε πολλούς τομείς την καθημερινότητά του. Ένας από αυτούς τους τομείς είναι και η μετακίνησή του, στην οποία, πλέον, έχουν εισαχθεί τα ηλεκτρικά οχήματα. Ένα πρόβλημα που έχει δημιουργηθεί με την είσοδο των ηλεκτρικών οχημάτων στην αυτοκίνηση, είναι οι μεγάλες ενεργειακές απαιτήσεις για τη φόρτιση των οχημάτων αυτών. Έτσι, η πρόβλεψη της ενεργειακής ζήτησης χαρακτηρίζεται ως αναγκαία και σε πολλές περιπτώσεις επιβεβλημένη διαδικασία αφού μπορεί να συμβάλει στην επίλυση αυτού του προβλήματος.

Στη συγκεκριμένη πτυχιακή εργασία στοχεύσαμε στην πρόβλεψη της απαιτούμενης ενέργειας για ένα δίκτυο φορτιστών. Η μεθοδολογία που αναπτύξαμε συνέκρινε τρία μοντέλα μηχανικής μάθησης, τα οποία ανταποκρίθηκαν στις απαιτήσεις του εν λόγω προβλήματος. Αυτό αποδεικνύεται από τις επιδόσεις των μοντέλων στους διάφορους χρονικούς ορίζοντες πρόβλεψης.

Συγκρίνοντας τα τρία μοντέλα που αναπτύχθηκαν, συμπεραίνουμε ότι η χρήση γραφημάτων συνελκτικών νευρωνικών δικτύων είναι πιο αποδοτική για τις προβλέψεις σε όλους τους ορίζοντες πρόβλεψης από τα άλλα μοντέλα, καθώς είναι καλύτερα στον εντοπισμό χωρικών (spatial) και χρονικών (temporal) συσχετίσεων (correlation) στο δίκτυο φορτιστών σε σχέση με τα απλά μοντέλα συνελκτικών νευρωνικών δικτύων.

5.2 Μελλοντικές Εργασίες

Τα μοντέλα που αναπτύχθηκαν στο πλαίσιο αυτής της πτυχιακής εργασίας, προϋποθέτουν την ύπαρξη ενός σταθερού αριθμού φορτιστών σε ένα δίκτυο. Σε μελλοντικές έρευνες, θα μπορούσε να γίνει κάποια ενδεχόμενη ενσωμάτωση δυναμικά εξελισσόμενων γράφων για την εφαρμογή της μεθοδολογίας σε ένα αναπτυσσόμενο δίκτυο φορτιστών [20].

Τέλος, ένα συνηθισμένο πρόβλημα σε τέτοιες έρευνες είναι η αντιμετώπιση περιορισμών στην πληροφορία που ορίζουν οι εταιρείες των σταθμών φόρτισης, όπως η ζήτηση που χάνεται προς ανταγωνιστικές εταιρείες ή η μέγιστη χωρητικότητα των φορτιστών ενός δικτύου. Η αντιμετώπιση του προβλήματος της περιορισμένης πληροφορίας έχει διατυπωθεί σε αντίστοιχες έρευνες, τα αποτελέσματα των οποίων μπορούν να ενσωματωθούν στη μεθοδολογία που παρουσιάστηκε [21].

Πίνακας Ορολογιών

Ξενόγλωσση Ορολογία	Ελληνική Ορολογία
Activation	Ενεργοποίηση
Adjacency Matrix	Πίνακας Γειτνίασης
Artificial Intelligence	Τεχνητή Νοημοσύνη
Artificial Neural Network	Τεχνητό Νευρωνικό Δίκτυο
Convolutional Layers	Συνελικτικές Στοιβάδες
Convolutional Neural Network	Συνελικτικό Νευρωνικό Δίκτυο
Correlation	Συσχέτιση
Currency	Νόμισμα Συναλλαγής
Decision Tree	Δέντρο Απόφασης
Deep Learning	Βαθιά Μάθηση
Degree Matrix	Πίνακας Βαθμών
Dictionary	Λεξικό
Driver Postal Code	Ταχυδρομικός Κώδικας Οδηγού
Edge	Ακμή
Electric Vehicle	Ηλεκτρικό Όχημα
Electric Vehicle Charging Station	Σταθμός Φόρτισης Ηλεκτρικών Οχημάτων
Ensemble Learning	Μάθηση Συνόλου
Feature Channel	Κανάλι Χαρακτηριστικών
Fee	Κόστος
Filter	Φίλτρο
Forget Gate	Πύλη Λήθης
Fully Connected Layers	Πλήρως Συνδεδεμένες Στοιβάδες
Gaussian Process Regression	Γκαουσιανή Διαδικασία Παλινδρόμησης
Geographic Information Systems	Γεωγραφικό Πληροφοριακά Συστήματα
Geographical Point	Γεωγραφικό Σημείο
Geospatial Data	Γεωχωρικά Δεδομένα
Global Positioning System	Παγκόσμιο Σύστημα Γεωεντοπισμού
Graph	Γράφος

Ξενόγλωσση Ορολογία	Ελληνική Ορολογία
Graph Convolutional Network	Γραφήματα Συνελικτικών Νευρωνικών Δικτύων
Grid	Πλέγμα
Haversine Formula	Φόρμουλα Haversine
Image Recognition	Αναγνώριση Εικόνων
Input Gate	Πύλη Εισόδου
Intelligent Agents	Ευφυείς Πράκτορες
K-Nearest Neighbors	K-Κοντινότεροι Γείτονες
Knowledge Discovery process from Data	Διαδικασία Ανακάλυψης Γνώσης από Δεδομένα
Label	Ετικέτα
Large Language Model	Μεγάλα Γλωσσικά Μοντέλα
Latitude	Γεωγραφικό Πλάτος
Logistic Regression	Λογιστική Παλινδρόμηση
Long Short-Term Memory	Μακρά Βραχυπρόθεσμη Μνήμη
Longitude	Γεωγραφικό Μήκος
Machine Learning	Μηχανική Μάθηση
Mean Absolute Error	Μέσο Απόλυτο Σφάλμα
Metadata	Μεταδεδομένα
Neural Network	Νευρωνικό Δίκτυο
Node	Κόμβος
Optimizer	Βελτιστοποιητής
Output Gate	Πύλη Εξόδου
Pattern Recognition	Αναγνώριση Προτύπων
Plug Type	Τύπος Φορτιστή
Point Of Interest	Σημείο Ενδιαφέροντος
Pooling	Υποδειγματοληψία
Pooling Layers	Επίπεδα Υποδειγματοληψίας
Port Type	Τύπος Θύρας
Recurrent Neural Networks	Αναδρομικά Νευρωνικά Δίκτυα

Ξενόγλωσση Ορολογία	Ελληνική Ορολογία
Reinforcement Learning	Ενισχυτική Μάθηση
Root Mean Squared Error	Ριζικό Μέσο Τετραγωνικό Σφάλμα
Semi-supervised Learning	Ημι-επιβλεπόμενη Μάθηση
Sequential Data	Ακολουθιακά Δεδομένα
Spatial Correlation	Χωρική Συσχέτιση
Supervised Learning	Επιβλεπόμενη Μάθηση
Support Vector Machine	Υποστηρικτική Μηχανή Διανυσμάτων
Support Vector Regression	Υποστήριξη Διανυσματικής Παλινδρόμησης
Temporal Correlation	Χρονική Συσχέτιση
Temporal Graph Convolutional Network	Γραφήματα Συνελκτικών Νευρωνικών Δικτύων με Χρονικές Διαστάσεις
Testing	Δοκιμή
Testing Set	Σύνολο Δοκιμής
Timeseries	Χρονοσειρά
Timestamp	Χρονοσφραγίδα
Training	Εκπαίδευση
Training Set	Σύνολο Εκπαίδευσης
Transfer Learning	Μάθηση από Μεταφορά
Unsupervised Learning	Μη επιβλεπόμενη Μάθηση
User ID	Αριθμός Χρήστη

Πίνακας Συντμήσεων - Αρκτικόλεξων - Ακρωνύμιων

Αρκτικόλεξο	Πλήρης Σημασία
AI	Artificial Intelligence
ANN	Artificial Neural Network
ARIMA	AutoRegressive Integrated Moving Average
CA	California
CNN	Convolutional Neural Network
ConvLSTM	Convolutional LSTM
DT	Decision Tree
EV	Electric Vehicle
EVCS	Electric Vehicle Charging Station
GCN	Graph Convolutional Network
GEORGE	enerGy dEmand fOrecasting foR charGing infrastructurE
GIS	Geographic Information System
GPR	Gaussian Process Regression
GPS	Global Positioning System
GPT	Generative Pre-trained Transformer
KDD	Knowledge Discovery process from Data
KNN	K-Nearest Neighbors
kW	kiloWatt
kWh	kiloWatt hour
LLM	Large Language Model
LSTM	Long Short-Term Memory
MAE	Mean Absolute Error
ML	Machine Learning
NN	Neural Network
POI	Points Of Interest
ReLU	Rectified Linear Unit

Αρκτικόλεξο	Πλήρης Σημασία
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
SVM	Support Vector Machines
SVR	Support Vector Regression
T-GCN	Temporal Graph Convolutional Network
XGBoost	eXtreme Gradient Boost

Βιβλιογραφία

- [1] LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* 521, 436–444 (2015). URL: <https://doi.org/10.1038/nature14539>
- [2] Sepp Hochreiter, Jürgen Schmidhuber; Long Short-Term Memory. *Neural Comput* 1997; 9 (8): 1735–1780. URL: <https://doi.org/10.1162/neco.1997.9.8.1735>
- [3] Han, X.; Gong, S. LST-GCN: Long Short-Term Memory Embedded Graph Convolution Network for Traffic Flow Forecasting. *Electronics* 2022, 11, 2230. <https://doi.org/10.3390/electronics11142230>
- [4] Ali Agga, Ahmed Abbou, Moussa Labbadi, Yassine El Houm, Imane Hammou Ou Ali, CNN-LSTM: An efficient hybrid deep learning architecture for predicting short-term photovoltaic power production, *Electric Power Systems Research*, Volume 208, 2022, 107908, ISSN 0378-7796, <https://doi.org/10.1016/j.epsr.2022.107908>
- [5] Le, T.; Vo, M.T.; Vo, B.; Hwang, E.; Rho, S.; Baik, S.W. Improving Electric Energy Consumption Prediction Using CNN and Bi-LSTM. *Appl. Sci.* 2019, 9, 4237. <https://doi.org/10.3390/app9204237>
- [6] Ryu, S., Lim, J., Hong, S. H. & Kim, W. Y. Deeply learning molecular structure-property relationships using attention- and gate-augmented graph convolutional network, 2018. <https://doi.org/10.48550/arXiv.1805.10988>
- [7] Ke, Z.; Li, Z.; Zhou, C.; Sheng, J.; Silamu, W.; Guo, Q. Rumor Detection on Social Media via Fused Semantic Information and a Propagation Heterogeneous Graph. *Symmetry* 2020, 12, 1806. <https://doi.org/10.3390/sym12111806>
- [8] Vontzos, G.; Laitos, V.; Charakopoulos, A.; Bargiotas, D.; Karakasidis, T.E. Estimating Spatio-Temporal Building Power Consumption Based on Graph Convolution Network Method. *Dynamics* 2024, 4, 337-356. <https://doi.org/10.3390/dynamics4020020>
- [9] Krielkinge, G.V., Cauwer, C.D, Sapountzoglou, N., Coosemans, T., Messagie, M. (2021). Day-Ahead Forecast of Electric Vehicle Charging Demand with Deep Neural Networks. Proceedings of 34th International Electric Vehicle Symposium and Exhibition (EVS34), Nanjing, China, 25-28 June 2021. URL: <https://www.mdpi.com/2032-6653/12/4/178>
- [10] Meddi, S., Cavaglioni, S., Cerquitelli, T., Manfredi, E., Regalia, A., Menolascino, R., Zardo, G. (2023). Data-driven energy demand forecasting for electric vehicle charging infrastructure. Proceedings of the EDBT/ICDT 2023 Joint Conference, March 28-March 31, 2023, Ioannina, Greece. URL: https://ceur-ws.org/Vol-3379/DARLI-AP_2023_7.pdf
- [11] Kim, Y.; Kim, S. Forecasting Charging Demand of Electric Vehicles Using Time-Series Models. *Energies* 2021, 14, 1487. <https://doi.org/10.3390/en14051487>
- [12] M. C. El Rai, S. A. Hadi, H. A. Damis and A. Gawanmeh, "Prediction of Electric Vehicle Charging Stations Distribution Using Machine Learning," *2022 5th International Conference on Signal Processing and Information Security (ICSPIS)*, Dubai, United Arab Emirates, 2022, pp. 154-157, doi: 10.1109/ICSPIS57063.2022.10002556

- [13] X. Huang, D. Wu and B. Boulet, "MetaProbformer for Charging Load Probabilistic Forecasting of Electric Vehicle Charging Stations," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 10, pp. 10445-10455, Oct. 2023, doi: 10.1109/TITS.2023.3276947
- [14] Oyedeji, Mojeed O., AlDhaifallah, Mujahed, Rezk, Hegazy, Mohamed, Ahmed Ali A., Computational Models for Forecasting Electric Vehicle Energy Demand, *International Journal of Energy Research*, 2023, 1934188, 16 pages, 2023. <https://doi.org/10.1155/2023/1934188>
- [15] Jiao, C., Heitzler, M., & Hurni, L. (2021). A survey of road feature extraction methods from raster maps. *Transactions in GIS*, 25, 2734–2763. URL: <https://doi.org/10.1111/tgis.12812>
- [16] Hüttel, F.B, Peled, I., Rodrigues, F., Pereira, F.C. (2020). Deep Spatio-Temporal Forecasting of Electrical Vehicle Charging Demand. Proceedings of Tackling Climate Change with Machine Learning Workshop at ICML 2021, Online Event, July 23rd, 2020. URL: <https://arxiv.org/pdf/2106.10940.pdf>
- [17] Robusto, C. C. "The Cosine-Haversine Formula." *The American Mathematical Monthly* 64, no. 1 (1957): 38–40. <https://doi.org/10.2307/2309088>
- [18] Executable Books Community. (2020). *Jupyter Book (v0.10)*. Zenodo. <https://doi.org/10.5281/zenodo.2561065>
- [19] J. D. Hunter, "Matplotlib: A 2D Graphics Environment," in *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90-95, May-June 2007, doi: 10.1109/MCSE.2007.55
- [20] Pareja, A., Domeniconi, G., Chen, J., Ma, T., Suzumura, T., Kanezashi, H., Kaler, T., Schardl, T. and Leiserson, C. 2020. EvolveGCN: Evolving Graph Convolutional Networks for Dynamic Graphs. *Proceedings of the AAAI Conference on Artificial Intelligence*. 34, 04 (Apr. 2020), 5363-5370. DOI: <https://doi.org/10.1609/aaai.v34i04.5984>
- [21] Elia Biganzoli, Patrizia Boracchi, Ettore Marubini, A general framework for neural network models on censored survival data, *Neural Networks*, Volume 15, Issue 2, 2002, Pages 209-218, ISSN 0893-6080, [https://doi.org/10.1016/S0893-6080\(01\)00131-9](https://doi.org/10.1016/S0893-6080(01)00131-9)
- [22] CSIRO's Data61, StellarGraph Machine Learning Library, 2018, GitHub, URL: <https://github.com/stellargraph/stellargraph>