



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΟΛΟΓΩΝ ΚΑΙ ΑΕΡΟΝΑΥΠΗΓΩΝ ΜΗΧΑΝΙΚΩΝ
[ΤΟΜΕΑΣ]
ΤΟΜΕΑΣ ΔΙΟΙΚΗΣΗΣ ΚΑΙ ΟΡΓΑΝΩΣΗΣ

ΣΠΟΥΔΑΣΤΙΚΗ ΕΡΓΑΣΙΑ

Βιβλιογραφική επισκόπηση αλγορίθμων μηχανικής μάθησης, νευρωνικών
δικτύων και νευρωνικών δικτύων γραφών

ΑΠΟΣΤΟΛΟΣ Α. ΔΟΪΤΣΙΔΗΣ

1054556

ΚΑΡΑΚΑΠΙΛΙΔΗΣ ΝΙΚΟΛΑΟΣ, ΚΑΘΗΓΗΤΗΣ

Σπουδαστική εργασία υποβληθείσα στο Τμήμα Μηχανολόγων και Αεροναυπηγών του
Πανεπιστημίου Πατρών

ΠΑΤΡΑ, Οκτώβρης 2023

Η έγκριση της σπουδαστικής εργασίας δεν υποδηλοί την αποδοχή των γνώμων του συγγραφέα.
Κατά τη συγγραφή τηρήθηκαν οι αρχές της ακαδημαϊκής δεοντολογίας.

Βιβλιογραφική επισκόπηση αλγορίθμων μηχανικής μάθησης, νευρωνικών δικτύων και νευρωνικών δικτύων γράφων

Απόστολος Α. Δοϊτσίδης

ΠΕΡΙΛΗΨΗ

Ο τομέας της Τεχνητής Νοημοσύνης έχει σημειώσει αξιοσημείωτες εξελίξεις τα τελευταία χρόνια, ιδίως στον τομέα της μηχανικής μάθησης. Αυτές οι εξελίξεις άνοιξαν το δρόμο για μια επαναστατική εποχή, γνωστή ως εποχή της τεχνητής νοημοσύνης. Η εποχή αυτή χαρακτηρίζεται από καινοτόμες εξελίξεις σε διάφορους επιστημονικούς κλάδους, οι οποίες καθοδηγούνται από τη δύναμη των αλγορίθμων μηχανικής μάθησης και των νευρωνικών δικτύων (Coccia, 2019). Η παρούσα εργασία διενεργεί μια βιβλιογραφική επισκόπηση των βασικότερων αλγορίθμων και μεθόδων που συνιστούν το επιστημονικό αυτό πεδίο, εξετάζοντας τον βασικό τρόπο λειτουργίας τους.

Η εργασία έχει χωριστεί σε τρία κεφάλαια. Στο πρώτο αναλύουμε αλγορίθμους μηχανικής μάθησης, εξετάζοντας τις αρχές λειτουργίας τους και τις διαφορετικές τους κατηγορίες. Παρουσιάζουμε αλγορίθμους επιβλεπόμενης μάθησης όπως την γραμμική και λογιστική παλινδρόμηση, τον **K-NN**, τα **SVMs** αλλά και μη επιβλεπόμενης μάθησης όπως οι αλγόριθμοι συσταδοποίησης **K-Means** και **DBSCAN**.

Στο δεύτερο κεφάλαιο πραγματευόμαστε το αντικείμενο των νευρωνικών δικτύων και των βασικών αρχιτεκτονικών τους. Τα νευρωνικά δίκτυα αντιπροσωπεύουν μια ισχυρή κλάση αλγορίθμων μηχανικής μάθησης, εμπνευσμένη από τον τρόπο λειτουργίας του ανθρώπινου εγκεφάλου. Αυτοί οι αλγόριθμοι έχουν τη δυνατότητα να μαθαίνουν από τα δεδομένα, εξάγοντας δομικά και αριθμητικά χαρακτηριστικά που ενσωματώνονται σε πολυεπίπεδες δομές, γνωστές και ως "νευρωνικά δίκτυα" (Aggarwal, 2023). Αυτό το κεφάλαιο παρουσιάζει μια εισαγωγή στην βασική τους αρχιτεκτονική, αναλύοντας τα διάφορα επίπεδα που τους αποτελούν και τις αλληλεπιδράσεις μεταξύ τους. Ξεκινώντας με τον απλούστερο τύπο νευρωνικού δικτύου, το **Perceptron** (Rosenblatt, 1958), προχωράμε στα πολυεπίπεδα

νευρωνικά δίκτυα που αποτελούν τη βάση των περισσότερων σύγχρονων αρχιτεκτονικών. Εστιάζουμε στην εκπαίδευση των δικτύων με τον αλγόριθμο back-propagation και τέλος αναφερόμαστε σε προβλήματα που μπορούν να προκύψουν κατά τη διάρκεια της εκπαίδευσης των νευρωνικών δικτύων και πώς αυτά μπορούν να αντιμετωπιστούν. Το κεφάλαιο ολοκληρώνεται με την ανάλυση της σημασίας των νευρωνικών δικτύων στη σύγχρονη μηχανική μάθηση και την ευρύτερη επιστημονική κοινότητα.

Στο τρίτο και τελευταίο κεφάλαιο αναφερόμαστε στα Νευρωνικά Δίκτυα Γράφων (Graph Neural Networks - GNNs), τα οποία επεκτείνουν τις δυνατότητες των παραδοσιακών νευρωνικών δικτύων για να αντιμετωπίσουν δεδομένα που δομούνται σε μορφή γράφων (Scarselli et al., 2009). Πολλά προβλήματα του πραγματικού κόσμου, όπως κοινωνικά δίκτυα, βιολογικά δίκτυα, συστήματα συστάσεων και γνωστικοί γράφοι, μπορούν να αναπαρασταθούν φυσικά ως γράφοι (Kipf & Welling, 2016). Μερικά από τα νευρωνικά δίκτυα γράφων που αναφέρουμε είναι τα Graph Convolutional Networks, το GraphSage (Hamilton, Ying & Leskovec, 2017) και τα Graph Attention Networks (Veličković et al., 2017). Με την βιβλιογραφική επισκόπηση των παραπάνω, επιτυγχάνουμε την καλύτερη κατανόηση και οργάνωση της γνώσης στο πολύπλοκο αυτό επιστημονικό πεδίο.

Λέξεις κλειδιά

Μηχανική Μάθηση - Νευρωνικά Δίκτυα - Νευρωνικά Δίκτυα Γράφων

ABSTRACT

The field of Artificial Intelligence has seen remarkable developments in recent years, in particular in the area of machine learning. These developments have paved the way for a revolutionary era known as the age of artificial intelligence. This era is characterized by innovative developments in various scientific disciplines, driven by the power of machine learning algorithms and neural networks (Coccia, 2019). This paper conducts a literature review of the main algorithms and methods that constitute this scientific field, examining their basic mode of operation.

The work has been divided into three chapters. In the first one we analyze machine learning algorithms, examining their operating principles and their different categories. We present supervised learning algorithms such as linear and logistic regression, K-NN, SVMs and unsupervised learning algorithms such as K-Means and DBSCAN clustering algorithms.

In the second chapter we discuss the subject of neural networks and their basic architectures. Neural networks represent a powerful class of machine learning algorithms inspired by the way the human brain works. These algorithms have the ability to learn from data, extracting structural and numerical features embedded in layered structures, also known as "neural networks" (Aggarwal, 2023). This chapter presents an introduction to their basic architecture, discussing the different layers that make them up and the interactions between them. Starting with the simplest type of neural network, the Perceptron (Rosenblatt, 1958), we move on to the multilevel neural networks that form the basis of most modern architectures. We focus on the training of networks with the back-propagation algorithm and finally we discuss problems that can arise during the training of neural networks and how they can be addressed. The chapter concludes with an analysis of the importance of neural networks in modern machine learning and the wider scientific community.

In the third and final chapter we discuss Graph Neural Networks (GNNs), which extend the capabilities of traditional neural networks to deal with data structured in graph form (Scarselli et al, 2009). Many real-world problems, such as social networks, biological networks, recommender systems and cognitive graphs, can be physically represented as graphs (Kipf & Welling, 2016). Some of the graph neural networks we report on are Graph Convolutional Networks, GraphSage (Hamilton, Ying & Leskovec, 2017) and Graph Attention Networks

(Veličković et al, 2017). By reviewing the literature of the above, we achieve a better understanding and organization of knowledge in this complex scientific field.

Λέξεις κλειδιά

Machine Learning - Neural Networks - Graph Neural Networks

Πίνακας Περιεχομένων

ΠΕΡΙΛΗΨΗ.....	iv
ABSTRACT	vi
Πίνακας Περιεχομένων	viii
Πίνακας Εικόνων	x
Κεφάλαιο 1 – Μηχανική Μάθηση	1
1.1 Η τεχνητή νοημοσύνη.....	1
1.2 Βασικές προσεγγίσεις	2
1.2.1 Η Εποπτευόμενη μάθηση	3
1.2.2 Η μη εποπτευόμενη μάθηση.....	4
1.2.3 Η ενισχυτική μάθηση	4
1.3 Αλγόριθμοι Μηχανικής Μάθησης.....	4
1.3.1 Linear & Logistic Regression.....	4
1.3.2 k-Nearest Neighbors (k-NN)	7
1.3.3 Support Vector Machines (SVM)	9
1.3.4 Decision Trees	11
1.3.5 Random Forests	12
1.3.6 k-Means Clustering	14
1.3.7 DBSCAN (Density-Based Spatial Clustering of Applications with Noise)...	17
1.3.8 Hierarchical Clustering	18
1.4 Εφαρμογές Μηχανικής Μάθησης	19
Κεφάλαιο 2 – Νευρωνικά Δίκτυα.....	22
2.1 Εισαγωγή.....	22
2.2 Βασική Αρχιτεκτονική των Νευρωνικών Δικτύων	23
2.2.1 Δίκτυο Perceptron.....	24
2.2.2 Πολυεπίπεδα NN	26
2.2.3 Εκπαίδευση Δικτύου με backpropagation.....	30
2.2.4 Προβλήματα κατά την εκπαίδευση NN	31
2.3 Προηγμένες αρχιτεκτονικές νευρωνικών δικτύων για συστήματα συστάσεων.....	32
2.3.1 Convolutional Neural Networks (CNN)	33

2.3.2 Recurrent Neural Networks (RNN)	34
2.3.3 Long Short-Term Memory (LSTM)	35
2.4 Graph Neural Networks - Νευρωνικά Δίκτυα Γράφων	36
2.4.1 Εισαγωγή στους γράφους.....	36
2.4.2 Εισαγωγή στα νευρωνικά δίκτυα γράφων.....	39
2.4.3 Πρακτικές και αξιοσημείωτες εφαρμογές των GNN.....	41
2.4.4 Βασικές αρχιτεκτονικές GNNs.....	44
2.4.5 Νευρωνικά Δίκτυα Γράφων για ταξινόμηση κόμβων.....	47
2.4.6 Νευρωνικά δίκτυα γράφων για πρόβλεψη συνδέσμου	48
2.5 Μάθηση με αναπαράσταση - Representation Learning.....	51
2.5.1 Μάθηση με αναπαράσταση για επεξεργασία φυσικής γλώσσας.....	52
2.5.2 Αλγόριθμος Word2vec για εκμάθηση αναπαραστάσεων λέξεων	53
2.6 Μάθηση αναπαραστάσεων γράφων – Graph representation learning.....	55
Σύνοψη	1
Βιβλιογραφία	1

Πίνακας Εικόνων

Εικόνα 1 Παραδείγματα μοντελοποίησης δεδομένων με λογιστική και γραμμική παλινδρόμηση αντίστοιχα.....	6
Εικόνα 2 Γραφική αναπαράσταση της λογιστικής συνάρτησης σχήματος S.....	7
Εικόνα 3 Παράδειγμα ταξινόμησης του x όταν $k=5$	8
Εικόνα 4 Υλοποίηση ταξινομητή SVM.	11
Εικόνα 5 Δέντρο αποφάσεων.	12
Εικόνα 6 Ταξινομητής Random Forest.	14
Εικόνα 7 Διάγραμμα συνάρτησης WCSS για τον προσδιορισμό του K	16
Εικόνα 8 Απεικόνιση του αλγορίθμου K-means.	17
Εικόνα 9 Σύγκριση βιολογικού νευρώνα και τεχνητού νευρωνικού δικτύου.	22
Εικόνα 10 Το απλούστερο μαθηματικό μοντέλο ενός νευρώνα που ονομάζεται Perceptron.	24
Εικόνα 11 Απλοποιημένο παράδειγμα κλασικού συνελκτικού νευρωνικού δικτύου.....	33
Εικόνα 12 Σύγκριση μεταξύ επαναλαμβανόμενου νευρωνικού δικτύου (RNN) και νευρωνικού δικτύου εμπρόσθιας τροφοδοσίας (FFNN).	35
Εικόνα 13 Διάγραμμα Συνάρτησης Μεταφοράς Βασικής Μονάδας LSTM.....	36
Εικόνα 14 Μη κατευθυνόμενος και κατευθυνόμενος γράφος αντίστοιχα.	37
Εικόνα 15 Ομοιογενής και ετερογενής γράφος αντίστοιχα.	38
Εικόνα 16 Παράδειγμα υπεργράφου με τέσσερις υπερακμές.....	38
Εικόνα 17 Σταθμισμένος γράφος.....	39
Εικόνα 18 Πεδία εφαρμογής νευρωνικών δικτύων γράφων	40
Εικόνα 19 Απεικόνιση της μεθόδου δειγματοληψίας και συγκέντρωσης του GraphSAGE... ..	45
Εικόνα 20 Εφαρμογές Νευρωνικών Δικτύων Γράφων: Ταξινόμηση κόμβων, πρόβλεψη ακμής, ταξινόμηση γράφου, ανίχνευση κοινότητας, ανίχνευση ανωμαλίας.	49
Εικόνα 21 Παραδείγματα μάθησης αναπαράστασης γράφων.....	56

Κεφάλαιο 1 – Μηχανική Μάθηση

1.1 Η τεχνητή νοημοσύνη

Η Τεχνητή Νοημοσύνη (AI) είναι ένας πολυεπιστημονικός τομέας της επιστήμης των υπολογιστών που στοχεύει στη δημιουργία ευφύων συστημάτων ικανών να εκτελούν εργασίες που απαιτούν συνήθως ανθρώπινη νοημοσύνη. Αυτά τα καθήκοντα περιλαμβάνουν μάθηση, συλλογισμό, επίλυση προβλημάτων, αντίληψη, κατανόηση φυσικής γλώσσας και λήψη αποφάσεων. Ο απώτερος στόχος της είναι να αναπτύξει μηχανές ή λογισμικό που μπορούν να μαθαίνουν αυτόνομα, επιτρέποντάς τους να προσαρμοστούν σε νέες καταστάσεις και να βελτιώσουν την απόδοσή τους με την πάροδο του χρόνου (Winston, 1992).

Η τεχνητή νοημοσύνη μπορεί να κατηγοριοποιηθεί ευρέως σε δύο τύπους: στενή (ή αδύναμη) τεχνητή νοημοσύνη και γενική (ή ισχυρή) τεχνητή νοημοσύνη (Goertzel & Pennachin, 2007):

- Ο όρος «Narrow AI» αναφέρεται σε συστήματα που έχουν σχεδιαστεί για να εκτελούν συγκεκριμένες εργασίες χωρίς να διαθέτουν ανθρώπινη νοημοσύνη. Αυτά τα συστήματα λειτουργούν κάτω από ένα περιορισμένο σύνολο περιορισμών και επικεντρώνονται σε μια μεμονωμένη εργασία ή σε ένα στενό εύρος εργασιών (Goertzel & Pennachin, 2007).
- Ο όρος «Artificial General Intelligence» (γενική τεχνητή νοημοσύνη, επίσης γνωστή ως ισχυρή τεχνητή νοημοσύνη ή τεχνητή γενική νοημοσύνη - AGI) αναφέρεται σε συστήματα που διαθέτουν ανθρώπινη νοημοσύνη και μπορούν να εκτελέσουν οποιαδήποτε πνευματική εργασία μπορεί να κάνει ένας άνθρωπος. Σε αντίθεση με τη στενή τεχνητή νοημοσύνη, η γενική τεχνητή νοημοσύνη μπορεί να κατανοήσει, να μάθει και να εφαρμόσει τη γνώση σε ένα ευρύ φάσμα τομέων (Goertzel & Pennachin, 2007).

Η τεχνητή νοημοσύνη περιλαμβάνει ένα ευρύ φάσμα δυνατοτήτων, συμπεριλαμβανομένων συστημάτων που μαθαίνουν νέες έννοιες, συλλογίζονται για τον κόσμο και αλληλεπιδρούν με τρόπο παρόμοιο με τον ανθρώπινο, μειώνοντας έτσι τους κινδύνους για τον άνθρωπο και βελτιώνοντας την παραγωγικότητα σε διάφορες εφαρμογές (Shubhendu & Jaiswal, 2013). Τα συστήματα τεχνητής νοημοσύνης

μαθαίνουν από δεδομένα, να λαμβάνουν αποφάσεις και να αλληλεπιδρούν με τους ανθρώπους και το περιβάλλον τους. Η τεχνητή νοημοσύνη έχει βρει εφαρμογές σε διάφορους τομείς της ζωής. Έχει χρησιμοποιηθεί σε κλάδους όπως η χρηματοδότηση, η υγειονομική περίθαλψη, οι βαριές βιομηχανίες, η εξυπηρέτηση πελατών, οι μεταφορές, οι τηλεπικοινωνίες, τα παιχνίδια, η μουσική, η αεροπορία και οι εκδόσεις ειδήσεων. Η τεχνητή νοημοσύνη έχει επίσης συμβάλει στην εκπαίδευση παρέχοντας έξυπνη διδασκαλία με τη βοήθεια υπολογιστή. Στον τομέα της ρομποτικής, η Τεχνητή Νοημοσύνη έχει χρησιμοποιηθεί για τον προγραμματισμό συμπεριφορών ρομπότ και την ανάπτυξη αυτόνομων ρομπότ ικανών να προσαρμόζονται σε μεταβαλλόμενα περιβάλλοντα. Συνολικά, η Τεχνητή Νοημοσύνη έχει τη δυνατότητα να φέρει επανάσταση σε πολλές πτυχές της ζωής μας και να βελτιώσει την παραγωγικότητα και την αποδοτικότητα σε διάφορους κλάδους (Shubhendu & Jaiswal, 2013).

1.2 Βασικές προσεγγίσεις

Η μηχανική μάθηση, ένας ταχέως εξελισσόμενος τομέας, βασίζεται στην υπολογιστική στατιστική, τη μαθηματική βελτιστοποίηση και την τεχνητή νοημοσύνη (LeCun, Bengio, & Hinton, 2015). Ο πρωταρχικός του στόχος είναι να δημιουργήσει μοντέλα που μπορούν αυτόματα να μάθουν και να βελτιωθούν από την εμπειρία χωρίς να είναι ρητά προγραμματισμένα (Samuel, 1959). Ο Mitchell (1997) παρέχει έναν επίσημο ορισμό της μηχανικής μάθησης: *"Ένα πρόγραμμα υπολογιστή λέγεται ότι μαθαίνει από την εμπειρία E σε σχέση με κάποια κατηγορία εργασιών T και το μέτρο απόδοσης P, εάν η απόδοσή του σε εργασίες στο T, όπως μετράται από το P, βελτιώνεται με εμπειρία E."*. Ανάλογα με τον τρόπο εκπαίδευσης ενός αλγορίθμου και με βάση τη διαθεσιμότητα της εξόδου κατά την εκπαίδευση, τα μοντέλα μηχανικής μάθησης μπορούν να ταξινομηθούν σε δέκα κατηγορίες. Αυτές περιλαμβάνουν: μάθηση με επίβλεψη ή εποπτευόμενη, μάθηση με ημиеπίβλεψη ή μη εποπτευόμενη, μάθηση χωρίς επίβλεψη ή μη εποπτευόμενη, ενισχυτική μάθηση, εξελικτική μάθηση, μάθηση συνόλου, τεχνητά νευρωνικά δίκτυα, μάθηση με βάση τις περιπτώσεις, αλγόριθμοι μείωσης διαστάσεων και υβριδική μάθηση (Alzubi, Nayyar, & Kumar, 2018). Η μάθηση με επίβλεψη περιλαμβάνει τη μάθηση μιας συνάρτησης που αντιστοιχίζει μια είσοδο σε μια έξοδο με βάση δείγματα ζευγών εισόδου-εξόδου. Η μάθηση χωρίς επίβλεψη περιλαμβάνει τη μάθηση από μη επισημασμένα δεδομένα για την ανακάλυψη μοτίβων ή σχέσεων. Η

μάθηση με ημιεπίβλεψη συνδυάζει επισημασμένα και μη επισημασμένα δεδομένα για τη δημιουργία μοντέλων. Η ενισχυτική μάθηση βασίζεται στην ανταμοιβή ή την ποινή και αποσκοπεί στην ανάληψη δράσης για την αύξηση της ανταμοιβής ή την ελαχιστοποίηση του κινδύνου (Sarker, 2021).

Σε ένα σύνολο δεδομένων με ετικέτες, κάθε παράδειγμα αντιστοιχίζεται με μια ετικέτα εξόδου. Για παράδειγμα, σε ένα σύνολο δεδομένων που χρησιμοποιείται για ένα πρόβλημα ταξινόμησης όπως η ανίχνευση ανεπιθύμητων μηνυμάτων, κάθε μήνυμα ηλεκτρονικού ταχυδρομείου (η είσοδος) χαρακτηρίζεται είτε ως "ανεπιθύμητο" είτε ως "όχι ανεπιθύμητο" (η ετικέτα εξόδου). Τα δεδομένα με ετικέτες χρησιμοποιούνται συνήθως στη μάθηση με επίβλεψη, όπου ο στόχος είναι να μάθουμε μια αντιστοίχιση από τις εισόδους στις εξόδους. Με την εκπαίδευση σε ένα σύνολο δεδομένων με ετικέτες, το μοντέλο μηχανικής μάθησης έχει ως στόχο να γενικεύσει καλά σε νέα, αθέατα δεδομένα με βάση τα πρότυπα που έχει εντοπίσει μεταξύ των ετικετών εισόδου και εξόδου κατά τη φάση της εκπαίδευσης (Sarker, 2021).

Τα μη επισημασμένα δεδομένα, από την άλλη πλευρά, αποτελούνται μόνο από τα δεδομένα εισόδου, χωρίς τις αντίστοιχες ετικέτες εξόδου. Τα μη επισημασμένα δεδομένα χρησιμοποιούνται κυρίως σε αλγορίθμους μάθησης χωρίς επίβλεψη, οι οποίοι προσπαθούν να εντοπίσουν εγγενείς δομές ή μοτίβα στα δεδομένα. Για παράδειγμα, ένας αλγόριθμος ομαδοποίησης μπορεί να χωρίσει ένα σύνολο δεδομένων από μη επισημειωμένα μηνύματα ηλεκτρονικού ταχυδρομείου σε διαφορετικές ομάδες με βάση τις ομοιότητες που εντοπίζει μεταξύ τους, παρόλο που δεν γνωρίζει εκ των προτέρων τι αντιπροσωπεύει κάποια από τις ομάδες (Sarker, 2021).

1.2.1 Η Εποπτευόμενη μάθηση

Η εποπτευόμενη μάθηση είναι ο πιο κοινός τύπος μηχανικής μάθησης, όπου τα δεδομένα εκπαίδευσης με ετικέτα χρησιμοποιούνται για τη δημιουργία ενός μοντέλου που αντιστοιχίζει τα χαρακτηριστικά εισόδου στα αποτελέσματα (Bishop, 2006). Αυτή η διαδικασία περιλαμβάνει την ελαχιστοποίηση μιας συνάρτησης απώλειας που ποσοτικοποιεί τη διαφορά μεταξύ προβλεπόμενων και πραγματικών ετικετών. Οι δημοφιλείς αλγόριθμοι εποπτευόμενης μάθησης περιλαμβάνουν τη γραμμική

παλινδρόμηση, την λογιστική παλινδρόμηση, τις μηχανές διανυσμάτων υποστήριξης (SVM) και τα νευρωνικά δίκτυα (Goodfellow, Bengio, & Courville, 2016).

1.2.2 Η μη εποπτευόμενη μάθηση

Στην μάθηση χωρίς επίβλεψη, ο αλγόριθμος λαμβάνει δεδομένα χωρίς ετικέτα και έχει ως αποστολή την ανακάλυψη υποκείμενων μοτίβων ή δομών (Hinton & Sejnowski, 1999). Οι κοινές τεχνικές περιλαμβάνουν ομαδοποίηση, μείωση διαστάσεων και εκτίμηση πυκνότητας. Αλγόριθμοι όπως το K-means, η ιεραρχική ομαδοποίηση, η ανάλυση κύριου συστατικού (PCA) και η t-κατανεμημένη ενσωμάτωση στοχαστικού γείτονα (t-SNE) χρησιμοποιούνται ευρέως για εργασίες μάθησης χωρίς επίβλεψη (Hastie, Tibshirani, & Friedman, 2009).

1.2.3 Η ενισχυτική μάθηση

Η ενισχυτική μάθηση εστιάζει σε μοντέλα εκπαίδευσης για τη λήψη αποφάσεων αλληλεπιδρώντας με ένα περιβάλλον (Sutton & Barto, 2018). Το μοντέλο, ή ο πράκτορας, λαμβάνει ανατροφοδότηση με τη μορφή ανταμοιβών ή κυρώσεων, τις οποίες χρησιμοποιεί για να μάθει και να βελτιστοποιήσει τις ενέργειές του. Η ενισχυτική μάθηση έχει εφαρμοστεί με επιτυχία σε διάφορους τομείς όπως η ρομποτική, τα παιχνίδια και τα συστήματα συστάσεων (Silver et al., 2016; Mnih et al., 2015).

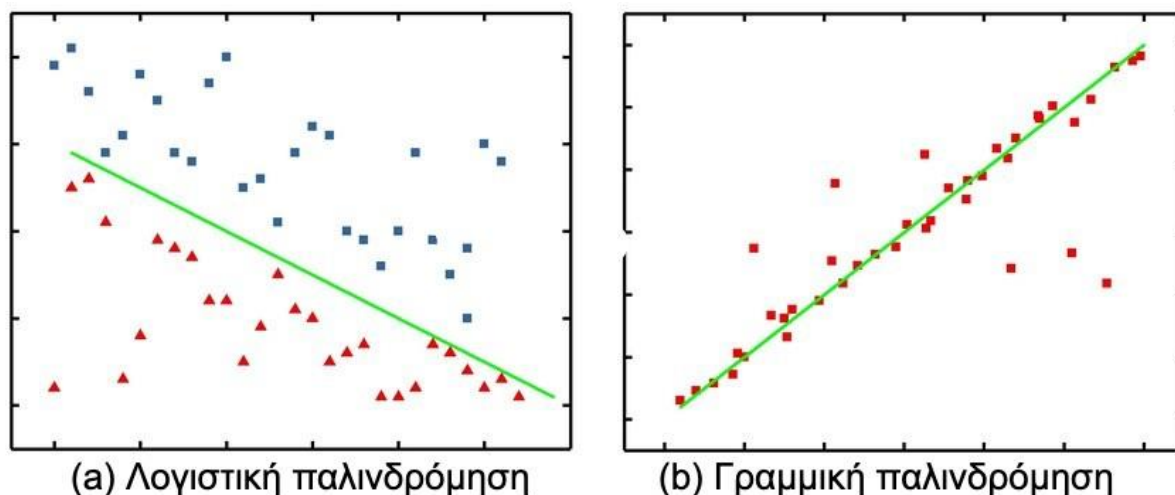
1.3 Αλγόριθμοι Μηχανικής Μάθησης

1.3.1 Linear & Logistic Regression

Η γραμμική παλινδρόμηση είναι μια στατιστική μέθοδος για τη μοντελοποίηση της σχέσης μεταξύ μιας συνεχούς εξαρτημένης μεταβλητής και μιας ή περισσότερων ανεξάρτητων μεταβλητών. Η γραμμική παλινδρόμηση αποσκοπεί στην καθιέρωση γραμμικής σχέσης μεταξύ των εξαρτημένων και ανεξάρτητων μεταβλητών. Με μαθηματικούς όρους, η σχέση αυτή εκφράζεται ως γραμμική εξίσωση, η οποία συχνά γράφεται με τη μορφή

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n + \epsilon$$

όπου y είναι η εξαρτημένη μεταβλητή, $x_1, x_2, \dots, x_n$ είναι οι ανεξάρτητες μεταβλητές και ϵ είναι ο όρος σφάλματος. Οι συντελεστές β αντιπροσωπεύουν τις κλίσεις της γραμμής ή του υπερεπιπέδου, οι οποίες εκτιμώνται από τα δεδομένα. Σε αυτό το πλαίσιο, ο όρος "γραμμική" αναφέρεται στην ιδέα ότι η εξίσωση σχηματίζει μια ευθεία γραμμή (ή υπερεπίπεδο σε υψηλότερες διαστάσεις) όταν απεικονίζεται σε ένα γράφημα, επιτρέποντας μια απλοποιημένη αλλά αποτελεσματική αναπαράσταση της σχέσης μεταξύ των μεταβλητών (Draper & Smith, 1998). Ο αλγόριθμος της γραμμικής παλινδρόμησης βασίζεται στην ελαχιστοποίηση του αθροίσματος των τετραγωνικών υπολειμμάτων, που είναι η διαφορά μεταξύ των παρατηρούμενων τιμών και των προβλεπόμενων τιμών από το μοντέλο. Στη γραμμική παλινδρόμηση, μπορούν να χρησιμοποιηθούν διάφορες τεχνικές βελτιστοποίησης για την εύρεση της γραμμής που ταιριάζει καλύτερα σε ένα σύνολο σημείων δεδομένων. Η μέθοδος των συνήθων ελαχίστων τετραγώνων (OLS) είναι μια παραδοσιακή μέθοδος που ελαχιστοποιεί το άθροισμα των τετραγώνων των υπολοίπων για την εύρεση των βέλτιστων παραμέτρων. Είναι υπολογιστικά αποδοτική, αλλά προϋποθέτει ότι τα δεδομένα είναι γραμμικά διαχωρίσιμα και τα σφάλματα κατανέμονται κανονικά. Από την άλλη πλευρά, η μέθοδος Gradient Descent είναι ένας επαναληπτικός αλγόριθμος βελτιστοποίησης που προσαρμόζει τις παραμέτρους του μοντέλου προς την κατεύθυνση της πιο απότομης καθόδου της συνάρτησης κόστους. Είναι ιδιαίτερα χρήσιμος για μεγάλα σύνολα δεδομένων και μπορεί να χειριστεί μη γραμμικές σχέσεις, αλλά μπορεί να απαιτεί συντονισμό υπερπαραμέτρων όπως ο ρυθμός μάθησης (Karimi et al., 2016). Επιπλέον, υπάρχουν μέθοδοι βελτιστοποίησης χωρίς παράγωγα που κατασκευάζουν απλουστευμένα μοντέλα γραμμικής παλινδρόμησης για τα κατάλοιπα και προσφέρουν ευελιξία και ευρωστία, ιδίως παρουσία θορύβου (Cartis et al., 2018). Ορισμένοι αλγόριθμοι συνδυάζουν ακόμη και μεθόδους βασισμένες στην κλίση και τις μεθόδους Newton για τη βελτιστοποίηση σε σενάρια εκτίμησης κατάστασης με κινούμενο ορίζοντα (Alessandri & Gaggero, 2017).

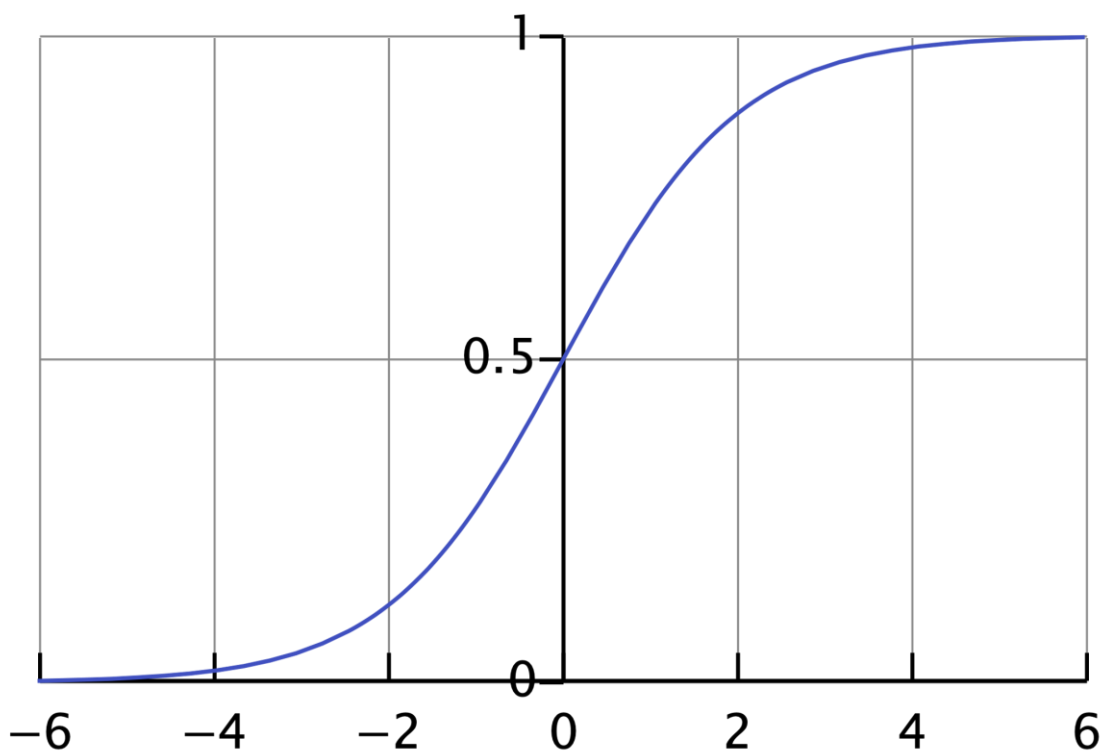


Εικόνα 1 Παραδείγματα μοντελοποίησης δεδομένων με λογιστική και γραμμική παλινδρόμηση αντίστοιχα.

Πηγή: (Fang et al. 2019)

Αντίστοιχα, η λογιστική παλινδρόμηση είναι μια στατιστική μέθοδος για τη μοντελοποίηση της πιθανότητας ενός δυαδικού αποτελέσματος, δεδομένου ενός συνόλου μεταβλητών πρόβλεψης. Επεκτείνει το πλαίσιο γραμμικής παλινδρόμησης εφαρμόζοντας τη λογιστική συνάρτηση, η οποία είναι μια καμπύλη σχήματος S που αντιστοιχίζει οποιαδήποτε είσοδο με πραγματική τιμή σε μια τιμή μεταξύ 0 και 1 (Hosmer Jr, Lemeshow, & Sturdivant, 2013). Η συνάρτηση έχει τη μορφή: $f(x) = \frac{1}{1+e^{-x}}$

Η έξοδος αυτής της συνάρτησης μοιάζει με ένα σύμβολο "S", γι' αυτό και συχνά αναφέρεται ως "καμπύλη S". Αυτή η καμπύλη S είναι ζωτικής σημασίας για τη λογιστική παλινδρόμηση, διότι επιτρέπει στο μοντέλο να χειρίζεται περιπτώσεις όπου η πιθανότητα ενός συγκεκριμένου αποτελέσματος δεν σχετίζεται γραμμικά με τις μεταβλητές εισόδου. Η καμπύλη ξεκινά από το 0, αυξάνεται σταδιακά και προσεγγίζει ασυμπτωτικά το 1. Αυτό διασφαλίζει ότι η έξοδος είναι οριοθετημένη μεταξύ 0 και 1, καθιστώντας την ερμηνεύσιμη ως πιθανότητα. Παρακάτω σχεδιάζουμε την συνάρτηση:



Εικόνα 2 Γραφική αναπαράσταση της λογιστικής συνάρτησης σχήματος S.

Η καμπύλη σχήματος S είναι ομαλή, γεγονός που την καθιστά διαφοροποιήσιμη, και αυτό είναι ιδιαίτερα χρήσιμο για αλγορίθμους βελτιστοποίησης όπως η κάθοδος κλίσης. Η καμπύλη είναι απότομη στο μέσο της, επιτρέποντας μια αποφασιστική ταξινόμηση, και είναι πιο επίπεδη στις ουρές, παρέχοντας κάποια ευελιξία για τις περιπτώσεις που είναι πιο δύσκολο να ταξινομηθούν.

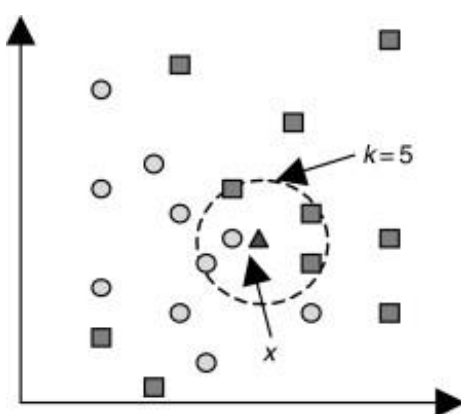
Αυτό επιτρέπει στην λογιστική παλινδρόμηση να μοντελοποιήσει την πιθανότητα να συμβεί ένα συμβάν (π.χ. επιτυχία έναντι αποτυχίας) ως συνάρτηση των μεταβλητών πρόβλεψης. Ο αλγόριθμος εκτιμά τις παραμέτρους της λογιστικής συνάρτησης χρησιμοποιώντας εκτίμηση μέγιστης πιθανότητας (MLE) ή άλλες τεχνικές βελτιστοποίησης. Η λογιστική παλινδρόμηση μπορεί να επεκταθεί για να χειριστεί πολλαπλές κλάσεις χρησιμοποιώντας τεχνικές όπως η μία έναντι όλων ή η πολωνυμική λογιστική παλινδρόμηση.

1.3.2 k-Nearest Neighbors (k-NN)

Ο αλγόριθμος k-Nearest Neighbors (k-NN) είναι ένας τύπος μεθόδου μάθησης βασισμένης σε δείγματα και χρησιμοποιείται στη μηχανική μάθηση για εργασίες

ταξινόμησης και παλινδρόμησης. Το " k " στο k -NN αντιπροσωπεύει τον αριθμό των πλησιέστερων γειτόνων που εξετάζει ο αλγόριθμος όταν πρέπει να κάνει μια πρόβλεψη. Η θεμελιώδης ιδέα πίσω από τον k -NN είναι η πρόβλεψη της κλάσης ή της τιμής ενός σημείου με βάση την πλειοψηφία ή τον μέσο όρο των γειτονικών του σημείων στον χώρο (Aha, Kibler, & Albert, 1991). Δεδομένου ενός συνόλου επισημασμένων σημείων δεδομένων εκπαίδευσης και ενός νέου, μη επισημασμένου σημείου, ο αλγόριθμος k -NN υπολογίζει την απόσταση μεταξύ του μη επισημασμένου σημείου και όλων των σημείων εκπαίδευσης. Οι αποστάσεις συνήθως μετρώνται χρησιμοποιώντας μετρήσεις Ευκλείδειας γεωμετρίας, ή άλλες μετρήσεις απόστασης. Στη συνέχεια, ο αλγόριθμος επιλέγει τα k σημεία εκπαίδευσης που είναι πιο κοντά στο σημείο του ερωτήματος και καθορίζει την πρόβλεψη ως εξής:

- Για εργασίες ταξινόμησης, ο αλγόριθμος k -NN εκχωρεί το σημείο ερωτήματος στην τάξη που έχει την πλειοψηφία μεταξύ των k -πλησιέστερων γειτόνων.
- Για εργασίες παλινδρόμησης, ο αλγόριθμος k -NN προβλέπει την τιμή του σημείου ερωτήματος με βάση τον μέσο όρο ή τον σταθμισμένο μέσο όρο των τιμών των k -πλησιέστερων γειτόνων του.



Εικόνα 3 Παράδειγμα ταξινόμησης του x όταν $k=5$.
Πηγή: (Chaudry et al. 2007)

Η επιλογή του k είναι μια κρίσιμη παράμετρος στον αλγόριθμο k -NN. Μια μικρή τιμή k μπορεί να οδηγήσει σε υπερπροσαρμογή και υψηλή ευαισθησία στο θόρυβο, ενώ μια μεγάλη τιμή k μπορεί να οδηγήσει σε υποπροσαρμογή και μειωμένη ευαισθησία στα τοπικά μοτίβα στα δεδομένα (Cover & Hart, 1967). Η υπερπροσαρμογή είναι ένα συνηθισμένο πρόβλημα στη μηχανική μάθηση, όπου ένα μοντέλο μαθαίνει πολύ καλά

τα δεδομένα εκπαίδευσης, συμπεριλαμβανομένου του θορύβου και των ακραίων τιμών τους, αλλά έχει κακές επιδόσεις σε νέα, αθέατα δεδομένα.

Ένα από τα πλεονεκτήματα του αλγορίθμου k-NN είναι η απλότητα και η ευκολία εφαρμογής του. Ωστόσο, ένα μειονέκτημα της χρήσης του αλγορίθμου είναι το υψηλό υπολογιστικό κόστος, ειδικά όταν πρόκειται για μεγάλα σύνολα δεδομένων, καθώς απαιτεί τον υπολογισμό των αποστάσεων μεταξύ του σημείου ερωτήματος και όλων των σημείων εκπαίδευσης. Επίσης, η απόδοση του αλγορίθμου μπορεί να επηρεαστεί αρνητικά από την παρουσία άσχετων ή εξαιρετικά συσχετισμένων χαρακτηριστικών.

1.3.3 Support Vector Machines (SVM)

Ο αλγόριθμος Support Vector Machines (SVM) είναι ένας ισχυρός αλγόριθμος εποπτευόμενης μάθησης που χρησιμοποιείται για εργασίες ταξινόμησης και παλινδρόμησης στη μηχανική μάθηση. Η βασική ιδέα πίσω από τα SVMs είναι να βρεθεί το βέλτιστο υπερ-επίπεδο που διαχωρίζει τα σημεία δεδομένων διαφορετικών κλάσεων με το μέγιστο περιθώριο (Cortes & Vapnik, 1995). Ως περιθώριο ορίζεται ως η απόσταση μεταξύ του υπερ-επίπεδου και των πλησιέστερων σημείων δεδομένων από κάθε κλάση, που ονομάζονται διανύσματα υποστήριξης. Μεγιστοποιώντας το περιθώριο, ο SVM στοχεύει να επιτύχει την καλύτερη απόδοση γενίκευσης και να ελαχιστοποιήσει τον κίνδυνο υπερβολικής προσαρμογής.

Ο SVM μπορεί να χειριστεί γραμμικά διαχωρίσιμα δεδομένα, δηλαδή με δεδομένα που μπορούν να χωριστούν σε διακριτές κλάσεις με μια ευθεία γραμμή. Γραμμικά διαχωρίσιμα είναι τα σημεία δεδομένων από διαφορετικές κλάσεις τα οποία μπορούν να διαχωριστούν εξ ολοκλήρου από μια γραμμή, ένα επίπεδο ή ένα υπερεπίπεδο δηλαδή σε περισσότερες από τρεις διαστάσεις. Ο SVM βρίσκει ένα γραμμικό όριο απόφασης, δηλαδή την ευθεία γραμμή, το επίπεδο ή το υπερεπίπεδο που διαχωρίζει τις διάφορες κλάσεις, υπολογίζοντας την εξίσωση που μεγιστοποιεί την απόσταση μεταξύ των πλησιέστερων σημείων των διαφορετικών κλάσεων, γνωστή ως περιθώριο.

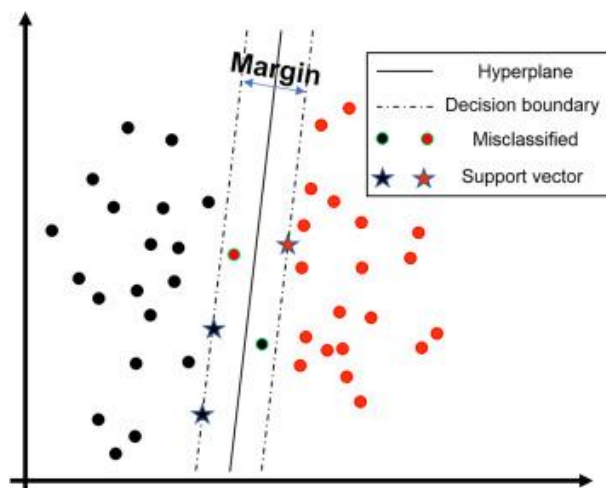
Σε περιπτώσεις όπου τα δεδομένα δεν διαχωρίζονται γραμμικά, χρησιμοποιεί μια τεχνική για να μετατρέψει τα δεδομένα εισόδου σε ένα χώρο υψηλότερης διάστασης όπου ο γραμμικός διαχωρισμός είναι δυνατός (Boser, Guyon, & Vapnik, 1992). Το SVM χρησιμοποιεί μια τεχνική που ονομάζεται kernel trick. Αυτή η τεχνική

απεικονίζει τα δεδομένα εισόδου σε έναν χώρο υψηλότερων διαστάσεων όπου αυτά καθίστανται γραμμικά διαχωρίσιμα. Η χαρτογράφηση είναι υπολογιστικά αποδοτική και επιτρέπει στο SVM να βρει ένα υπερεπίπεδο σε αυτόν τον νέο χώρο που μπορεί να διαχωρίσει αποτελεσματικά τα σημεία δεδομένων. Οι συναρτήσεις που χρησιμοποιούνται συνήθως περιλαμβάνουν τη γραμμική, πολυωνυμική, ακτινική συνάρτηση βάσης (RBF) και σιγμοειδείς πυρήνες.

Οι γραμμικοί πυρήνες είναι η απλούστερη μορφή των συναρτήσεων πυρήνα στο SVM. Χρησιμοποιούνται όταν τα δεδομένα είναι σχεδόν ή πλήρως γραμμικά διαχωρίσιμα. Ο γραμμικός πυρήνας είναι ουσιαστικά το τετραγωνικό γινόμενο των διανυσμάτων χαρακτηριστικών. Είναι λιγότερο εντατικός υπολογιστικά, αλλά μπορεί να μην είναι κατάλληλος για πολύπλοκα, μη γραμμικά διαχωρίσιμα δεδομένα (Noble, 2006). Ο πολυωνυμικός πυρήνας είναι κατάλληλος για δεδομένα που ακολουθούν πολυωνυμική σχέση, αλλά ο βαθμός του πολυωνύμου πρέπει να επιλέγεται προσεκτικά για να αποφεύγεται η υπερβολική προσαρμογή. Ο πυρήνας Radial Basis Function (RBF) χρησιμοποιείται συχνά για μη γραμμικά δεδομένα και απαιτεί συντονισμό παραμέτρων όπως η τιμή γάμμα. Ο σιγμοειδής πυρήνας μιμείται τη συμπεριφορά ενός νευρωνικού δικτύου με σιγμοειδή συνάρτηση ενεργοποίησης, αλλά χρησιμοποιείται λιγότερο συχνά (Noble, 2006).

Το SVM μπορεί να επεκταθεί σε προβλήματα ταξινόμησης πολλαπλών τάξεων χρησιμοποιώντας προσεγγίσεις όπως στρατηγικές one-vs-one ή one-vs-all. Επιπλέον, ο SVM μπορεί να προσαρμοστεί για εργασίες παλινδρόμησης εισάγοντας μια εναλλακτική συνάρτηση απώλειας που ονομάζεται απώλεια ε-insensitive, με αποτέλεσμα τον αλγόριθμο Support Vector Regression (SVR) (Drucker, Burges, Kaufman, Smola, & Vapnik, 1997). Ενώ η προβολή σε έναν χώρο υψηλότερων διαστάσεων μπορεί να κάνει τα δεδομένα γραμμικά διαχωρίσιμα, μπορεί επίσης να οδηγήσει στο πρόβλημα γνωστό ως dimensionality curse, όπου ο αριθμός των πιθανών λύσεων αυξάνεται εκθετικά, καθιστώντας δυσκολότερο για οποιονδήποτε αλγόριθμο να επιλέξει τη σωστή (Noble, 2006).

Η παρακάτω εικόνα παρουσιάζει μια υλοποίηση ταξινομητή SVM χωρίς συνάρτηση πυρήνα σε ένα σύνολο δεδομένων που έχει δύο χαρακτηριστικά και δύο κλάσεις:



Εικόνα 4 Υλοποίηση ταξινομητή SVM.
Πηγή : (Misra & Li, 2020)

Όλα τα δείγματα εκπαίδευσης αντιπροσωπεύονται ως κύκλοι ή αστέρια. Τα διανύσματα υποστήριξης (που συμβολίζονται ως αστέρια) προέρχονται από τα δείγματα εκπαίδευσης έτσι ώστε να είναι πιο κοντά στο υπερεπίπεδο μεταξύ των άλλων δειγμάτων εκπαίδευσης για καθεμία από τις δύο κατηγορίες. Δύο δείγματα εκπαίδευσης έχουν ταξινομηθεί εσφαλμένα επειδή βρίσκονται στη λάθος πλευρά του υπερεπιπέδου.

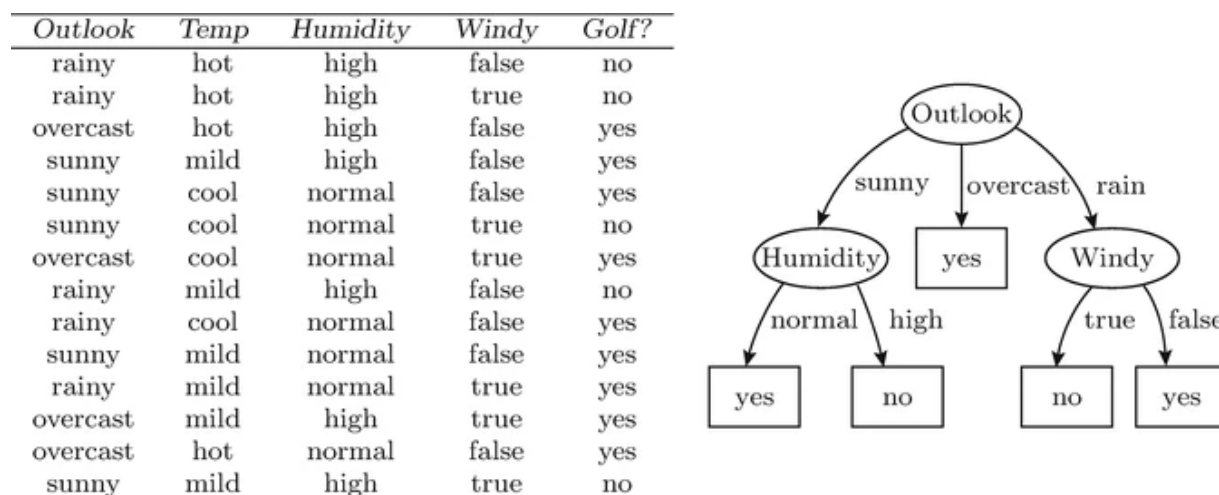
Με αυτήν την αναπαράσταση επεκτείνεται η περιγραφή του αλγορίθμου Support Vector Machines (SVM), αποτυπώνοντας ένα παράδειγμα ενός συγκεκριμένου σεναρίου στο οποίο το SVM χρησιμοποιείται για την ταξινόμηση δεδομένων.

1.3.4 Decision Trees

Οι αλγόριθμοι δέντρου αποφάσεων είναι μια κατηγορία εποπτευόμενων μεθόδων εκμάθησης που χρησιμοποιούνται για εργασίες ταξινόμησης και παλινδρόμησης στη μηχανική μάθηση. Αυτοί οι αλγόριθμοι διαχωρίζουν αναδρομικά τον χώρο των χαρακτηριστικών εισόδου σε διακριτές περιοχές με βάση τις τιμές των χαρακτηριστικών εισόδου, με αποτέλεσμα μια δομή που μοιάζει με δέντρο όπου οι εσωτερικοί κόμβοι αντιπροσωπεύουν διαχωρισμούς χαρακτηριστικών και οι κόμβοι φύλλων αντιπροσωπεύουν τις προβλεπόμενες ετικέτες ή τιμές κλάσεων (Breiman, Friedman, Stone , & Olshen, 1984). Υπάρχουν αρκετοί αλγόριθμοι δέντρων αποφάσεων, οι πιο συνηθισμένοι είναι οι ID3, C4.5 και CART (Classification and Regression Trees) (Quinlan, 1986; Quinlan, 1993; Breiman et al., 1984). Η κύρια

διαφορά μεταξύ αυτών των αλγορίθμων έγκειται στα κριτήρια διαχωρισμού και στο χειρισμό των συνεχών χαρακτηριστικών ή των δεδομένων που λείπουν. Τα κριτήρια διαχωρισμού χρησιμοποιούνται για τον προσδιορισμό του βέλτιστου χαρακτηριστικού και της τιμής για κάθε βήμα κατάτμησης. Τα κοινά χρησιμοποιούμενα κριτήρια διαχωρισμού περιλαμβάνουν την μετρική Gini impurity, το κέρδος πληροφοριών και τη μείωση της διακύμανσης. Ο αλγόριθμος του δέντρου αποφάσεων στοχεύει στη μεγιστοποίηση του επιλεγμένου κριτηρίου διαχωρισμού, οδηγώντας στις πιο ομοιογενείς ομάδες σημείων δεδομένων σε κάθε βήμα. Τα δέντρα αποφάσεων είναι εύκολα κατανοητά, μπορούν να χειριστούν τόσο αριθμητικά όσο και κατηγορικά δεδομένα και δεν απαιτούν εκτεταμένη προεπεξεργασία χαρακτηριστικών. Ωστόσο, είναι επιρρεπείς στην υπερβολική εφαρμογή, ειδικά όταν το δέντρο μεγαλώνει πολύ βαθιά. Τεχνικές όπως το κλάδεμα, η πρόωρη διακοπή ή ο περιορισμός του βάθους του δέντρου μπορούν να βοηθήσουν στον μετριασμό αυτού του προβλήματος.

Στην παρακάτω εικόνα παρουσιάζεται ένα σύνολο δεδομένων που περιγράφει τις καιρικές συνθήκες και μια μεταβλητή στόχο (Play Golf?) και ένα δέντρο αποφάσεων για αυτό το σύνολο δεδομένων:



Εικόνα 5 Δέντρο αποφάσεων.

Πηγή: (Quinlan 1986)

1.3.5 Random Forests

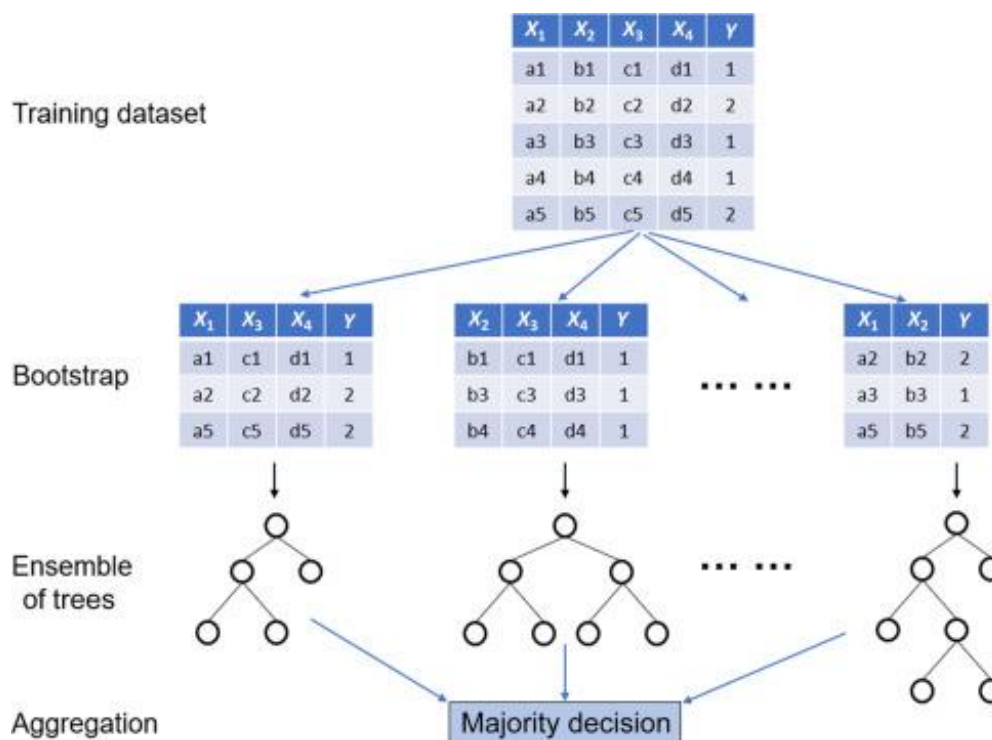
Τα τυχαία δάση είναι ακόμη ένας αλγόριθμος εκμάθησης συνόλου στη μηχανική μάθηση, που χρησιμοποιείται κυρίως για εργασίες ταξινόμησης και παλινδρόμησης. Εισήχθη από τον Leo Breiman το 2001 ως επέκταση των αλγορίθμων δέντρων

αποφάσεων. Η κύρια ιδέα πίσω από τα τυχαία δάση είναι να κατασκευαστούν πολλαπλά δέντρα απόφασης και να συνδυάσουν τις προβλέψεις τους μέσω πλειοψηφίας (για ταξινόμηση) ή κατά μέσο όρο (για παλινδρόμηση) για να παραχθεί ένα πιο ακριβές και ισχυρό μοντέλο (Breiman, 2001). Ο αλγόριθμος εισάγει δύο βασικές έννοιες για τη βελτίωση της απόδοσης των μεμονωμένων δέντρων αποφάσεων: συσσώρευση (συσσώρευση bootstrap) και τυχειότητα χαρακτηριστικών.

- Bagging: Πολλαπλά δέντρα αποφάσεων δημιουργούνται χρησιμοποιώντας δείγματα bootstrapped του αρχικού συνόλου δεδομένων. Ένα bootstrapped δείγμα δημιουργείται σχεδιάζοντας τυχαία σημεία δεδομένων με αντικατάσταση από το αρχικό σύνολο δεδομένων. Κάθε δέντρο απόφασης εκπαιδεύεται σε ένα διαφορετικό δείγμα bootstrapped, το οποίο εισάγει τη διαφορετικότητα στο σύνολο και μειώνει την υπερπροσαρμογή.
- Τυχειότητα χαρακτηριστικών: Κατά την κατασκευή κάθε δέντρου αποφάσεων, επιλέγεται ένα τυχαίο υποσύνολο χαρακτηριστικών σε κάθε διαχωρισμό, αντί να λαμβάνονται υπόψη όλα τα χαρακτηριστικά. Αυτή η τυχειότητα χαρακτηριστικών αυξάνει περαιτέρω την ποικιλομορφία μεταξύ των μεμονωμένων δέντρων και συμβάλλει στη συνολική ευρωστία του μοντέλου.

Ο αλγόριθμος συνδυάζει τις προβλέψεις μεμονωμένων δέντρων υπολογίζοντας τον μέσο όρο των εξόδων τους σε περίπτωση παλινδρόμησης ή λαμβάνοντας πλειοψηφία στην περίπτωση ταξινόμησης. Αυτή η προσέγγιση αξιοποιεί τη γνώση του πλήθους, οδηγώντας σε καλύτερη γενίκευση και μειωμένη υπερπροσαρμογή σε σύγκριση με τα δέντρα μεμονωμένων αποφάσεων.

Στην παρακάτω εικόνα παρουσιάζεται η υλοποίηση του ταξινομητή RF σε ένα σύνολο δεδομένων που έχει τέσσερα χαρακτηριστικά (X_1 , X_2 , X_3 και X_4) και δύο κλάσεις ($Y = 1$ και 2). Ο ταξινομητής RF είναι μια μέθοδος συνόλου που εκπαιδεύει πολλά δέντρα απόφασης παράλληλα με την εκκίνηση που ακολουθείται από τη συγκέντρωση. Κάθε δέντρο εκπαιδεύεται σε διαφορετικά υποσύνολα δείγματος εκπαίδευσης και χαρακτηριστικών.



Εικόνα 6 Ταξινομητής Random Forest.

Πηγή : (Misra & Li, 2020)

1.3.6 k-Means Clustering

Πρόκειται για έναν αλγόριθμο μάθησης χωρίς επίβλεψη που χρησιμοποιείται ευρέως στη μηχανική μάθηση για τη διαίρεση ενός συνόλου δεδομένων σε K διακριτά, μη επικαλυπτόμενα συμπλέγματα με βάση την ομοιότητα των σημείων δεδομένων. Ο αλγόριθμος προτάθηκε για πρώτη φορά από τον Stuart Lloyd το 1957 και αργότερα διαδόθηκε από τον James MacQueen το 1967. Στο πλαίσιο της συσταδοποίησης K-means, το κεντροειδές είναι ουσιαστικά ο μέσος όρος όλων των σημείων δεδομένων που ανήκουν σε μια συγκεκριμένη συστάδα. Χρησιμοποιεί ως το κέντρο αυτής της συστάδας. Ο αλγόριθμος στοχεύει στην ελαχιστοποίηση του αθροίσματος των τετραγωνικών αποστάσεων μεταξύ κάθε σημείου δεδομένων σε μια συστάδα και του κεντροειδούς της εν λόγω συστάδας. Το κεντροειδές επαναυπολογίζεται επαναληπτικά καθώς τα σημεία ανακατατάσσονται μεταξύ συστάδων και ο αλγόριθμος συγκλίνει όταν τα κεντροειδή δεν αλλάζουν πλέον σημαντικά. Ελαχιστοποιώντας το άθροισμα των τετραγωνικών αποστάσεων από το κεντροειδές σε κάθε συστάδα, ο αλγόριθμος K-means προσπαθεί να δημιουργήσει συστάδες που είναι συμπαγείς και καλά διαχωρισμένες μεταξύ τους. Η έννοια του κεντροειδούς είναι θεμελιώδης για τον

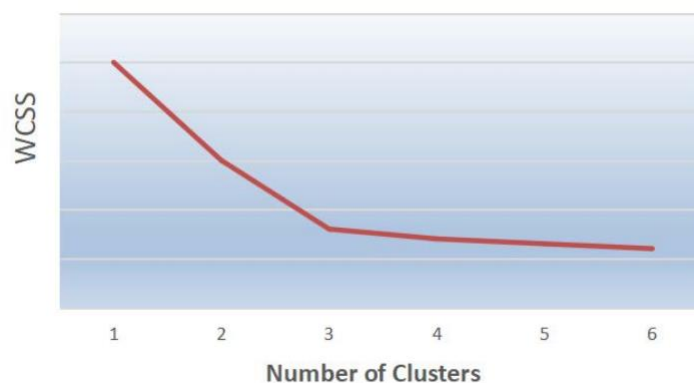
αλγόριθμο K-means και παίζει καθοριστικό ρόλο στον καθορισμό της ποιότητας της συσταδοποίησης (MacQueen, 1967). Ο αλγόριθμος K-means ακολουθεί μια επαναληπτική διαδικασία όπως η παρακάτω:

1. Επιλογή K σημείων δεδομένων από το σύνολο δεδομένων.
2. Αντιστοίχιση κάθε σημείου δεδομένων στο πλησιέστερο κέντρο με βάση μια μέτρηση απόστασης, συνήθως την Ευκλείδεια απόσταση.
3. Ενημέρωση των κεντροειδών υπολογίζοντας τον μέσο όρο όλων των σημείων δεδομένων που έχουν εκχωρηθεί σε κάθε κέντρο.
4. Επανάληψη βημάτων 2 και 3 μέχρι τη σύγκλιση, δηλαδή, όταν οι θέσεις των κεντροειδών δεν αλλάζουν σημαντικά μεταξύ των επαναλήψεων ή όταν επιτευχθεί ένας μέγιστος αριθμός επαναλήψεων.

Μία από τις κύριες προκλήσεις στην ομαδοποίηση K-means είναι ο καθορισμός της βέλτιστης τιμής για το K, του αριθμού των συστάδων. Διάφορες μέθοδοι, όπως η μέθοδος του αγκώνα, η ανάλυση σιλουέτας ή η στατιστική του χάσματος, μπορούν να βοηθήσουν στην εκτίμηση της κατάλληλης τιμής K. Για παράδειγμα η μέθοδος του αγκώνα ή elbow method, χρησιμοποιεί το άθροισμα τετραγώνων εντός της συστάδας (WCSS) για να αξιολογήσει την ποιότητα της συσταδοποίησης για διαφορετικές τιμές του K. Το WCSS υπολογίζεται ως

$$WCSS = \sum_{i=1}^K \sum_{x \in C_i} ||x - m_i||^2$$

όπου C_i είναι η i-οστή συστάδα και m_i το κεντροειδές της. Το σημείο "αγκώνας" στο διάγραμμα του WCSS σε σχέση με τις τιμές του K υποδεικνύει το βέλτιστο K (Cui, 2020). Χρησιμοποιούμε το ακόλουθο σχήμα ως παράδειγμα. Όταν το k είναι ίσο με 3, η καμπύλη παρουσιάζει σημείο καμπής, το οποίο μπορεί να θεωρηθεί ως ο "αγκώνας" αυτής της καμπύλης. Καθορίζεται δηλαδή η τιμή k, δηλαδή ο βέλτιστος αριθμός συστάδων ο οποίος είναι τρεις σε αυτό το παράδειγμα.



Εικόνα 7 Διάγραμμα συνάρτησης WCSS για τον προσδιορισμό του K.
Πηγή: (Cui, M. ,2020)

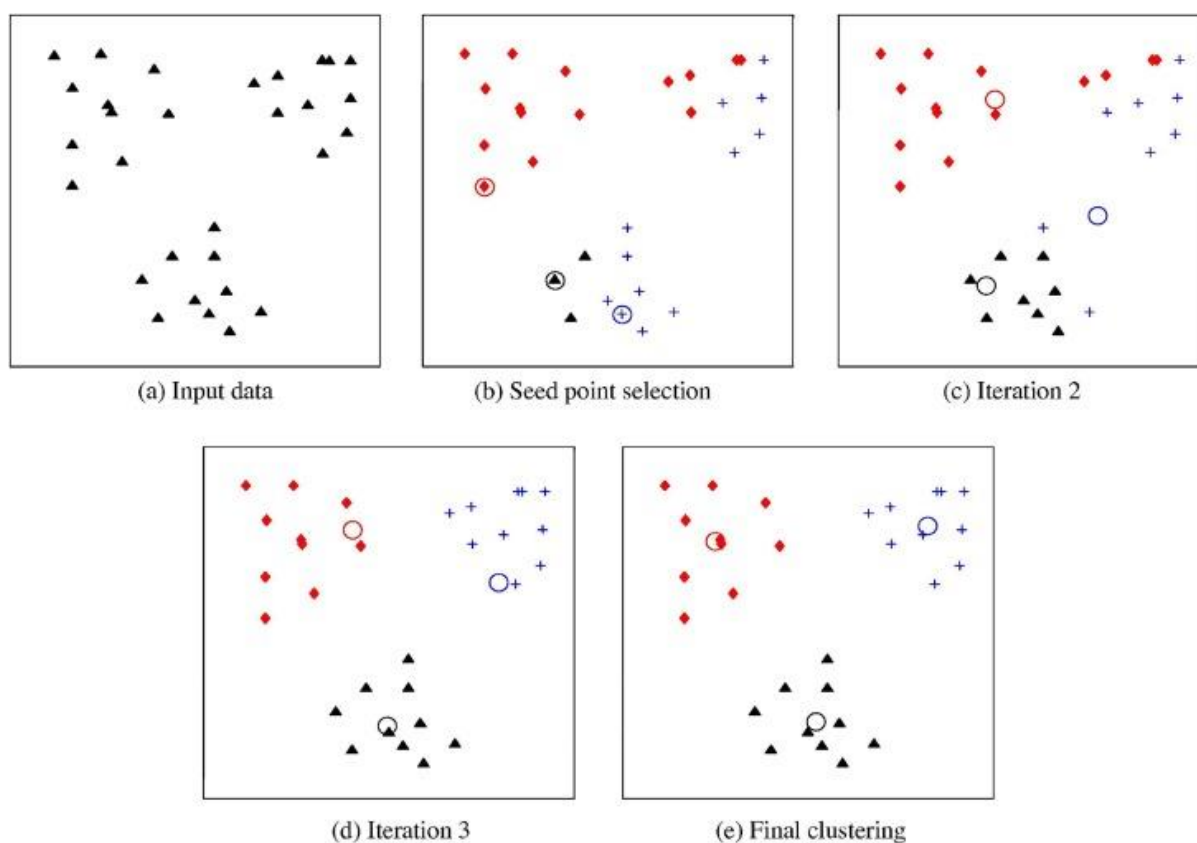
Μια άλλη πρόκληση είναι ότι ο αλγόριθμος K-means είναι ευαίσθητος στην αρχική τοποθέτηση των κεντροειδών, γεγονός που μπορεί να οδηγήσει σε σύγκλιση σε ένα τοπικό ελάχιστο αντί για το καθολικό ελάχιστο. Πολλαπλές εκτελέσεις με διαφορετικές αρχικοποιήσεις ή χρήση τεχνικών όπως το K-means++ κατά τους Arthur και Vassilvitskii (2007) μπορούν να βοηθήσουν στην άμβλυνση αυτού του ζητήματος. Ο αλγόριθμος K-means++ είναι μια βελτίωση του τυπικού αλγορίθμου K-means, ειδικά σχεδιασμένος για να αντιμετωπίσει την ευαισθησία του K-means στην αρχική τοποθέτηση των κεντροειδών. Στον τυπικό αλγόριθμο K-means, τα αρχικά κεντροειδή επιλέγονται συνήθως τυχαία, γεγονός που μπορεί να οδηγήσει σε κακή σύγκλιση ή να κολλήσει σε τοπικά ελάχιστα. Ο K-means++ το βελτιώνει αυτό παρέχοντας έναν πιο έξυπνο τρόπο αρχικής τοποθέτησης των κεντροειδών.

Στον K-means++, το πρώτο κεντροειδές επιλέγεται ομοιόμορφα τυχαία από τα σημεία δεδομένων που ομαδοποιούνται. Το επόμενο κεντροειδές επιλέγεται από τα υπόλοιπα σημεία δεδομένων με πιθανότητα ανάλογη του τετραγώνου της απόστασης του σημείου από το πλησιέστερο υπάρχον κεντροειδές. Αυτή η διαδικασία επαναλαμβάνεται μέχρι να επιλεγούν k κεντροειδή.

Στο παρακάτω σύνολο εικόνων παρουσιάζονται τα βήματα του αλγορίθμου K-means σε ένα σύνολο δεδομένων:

- (α) Αρχικά, απεικονίζονται δισδιάστατα δεδομένα εισόδου που ανήκουν σε τρεις συστάδες

- (β) Στη συνέχεια, επιλέγονται τρία αρχικά κέντρα συστάδων (σημεία έναρξης) και πραγματοποιείται η αρχική αντιστοίχιση των σημείων δεδομένων σε συστάδες.
- (γ) και (δ) Εμφανίζονται οι ενδιάμεσες επαναλήψεις του αλγορίθμου, κατά τις οποίες ενημερώνονται οι ετικέτες των συστάδων και τα κέντρα τους.
- (ε) Τέλος, παρουσιάζεται η τελική ομαδοποίηση που επιτυγχάνεται με τον αλγόριθμο K-means κατά τη διαδικασία σύγκλισης.



Εικόνα 8 Απεικόνιση του αλγορίθμου K-means.

Πηγή: (Ikotun et al., 2023)

1.3.7 DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

Η χωρική ομαδοποίηση εφαρμογών με θόρυβο βάσει πυκνότητας (DBSCAN) είναι ένας αλγόριθμος μάθησης χωρίς επίβλεψη που χρησιμοποιείται για την ομαδοποίηση εργασιών στη μηχανική μάθηση. Ο DBSCAN, ο οποίος εισήχθη από τους Martin Ester, Hans-Peter Kriegel, Jörg Sander και Xiaowei Xu το 1996, είναι ένας αλγόριθμος ομαδοποίησης που βασίζεται στην πυκνότητα που ομαδοποιεί τα σημεία δεδομένων με βάση την εγγύτητα και την πυκνότητά τους, ενώ επίσης προσδιορίζει και διαχωρίζει

σημεία θορύβου (Ester, Kriegel, Sander, & Xu, 1996). Το DBSCAN λειτουργεί με βάση την αρχή ότι ένα σύμπλεγμα είναι μια πυκνή περιοχή σημείων δεδομένων που χωρίζονται από περιοχές χαμηλότερης πυκνότητας σημείου. Ο αλγόριθμος απαιτεί δύο παραμέτρους: ένα έψιλον ακτίνας (ϵ) που ορίζει τη γειτονιά γύρω από κάθε σημείο δεδομένων και έναν ελάχιστο αριθμό σημείων (MinPts) που καθορίζει την ελάχιστη πυκνότητα που απαιτείται για να σχηματιστεί ένα σύμπλεγμα. Ο αλγόριθμος DBSCAN ακολουθεί τα εξής βήματα:

1. Για κάθε σημείο δεδομένων, προσδιορισμός όλων των σημείων γύρω του
2. Εάν ο αριθμός των σημείων εντός της ϵ -γειτονιάς είναι μεγαλύτερος ή ίσος με MinPts, σημειώνεται το σημείο δεδομένων ως κεντρικό σημείο και σχηματίζεται ένα νέο σύμπλεγμα. Σε διαφορετική περίπτωση, σημειώνεται ως οριακό σημείο ή θόρυβο.
3. Για κάθε κεντρικό σημείο, συγχωνεύονται επαναληπτικά τα γειτονικά σημεία του με το τρέχον σύμπλεγμα, συμπεριλαμβανομένων τυχόν πρόσθετων βασικών σημείων που ανακαλύφθηκαν στη διαδικασία.
4. Συνέχεια της διαδικασίας έως ότου υποβληθούν σε επεξεργασία όλα τα βασικά σημεία και αντιστοιχιστούν σε συμπλέγματα, ενώ τα σημεία συνόρων εκχωρούνται στο σύμπλεγμα του πλησιέστερου πυρήνα και τα σημεία θορύβου παραμένουν μη εκχωρημένα.

Το DBSCAN έχει πολλά πλεονεκτήματα σε σχέση με τους αλγόριθμους ομαδοποίησης που βασίζονται σε διαμερίσματα, όπως το K-means. Μπορεί να βρει συμπλέγματα αυθαίρετων σχημάτων, να χειριστεί αποτελεσματικά το θόρυβο και δεν απαιτεί από τον χρήστη να καθορίσει τον αριθμό των συστάδων. Ωστόσο, είναι ευαίσθητο στην επιλογή των παραμέτρων ϵ και MinPts, οι οποίες μπορούν να επηρεάσουν σημαντικά τα αποτελέσματα της ομαδοποίησης.

1.3.8 Hierarchical Clustering

Η ιεραρχική ομαδοποίηση είναι ένας αλγόριθμος μάθησης χωρίς επίβλεψη που χρησιμοποιείται για την ομαδοποίηση εργασιών στη μηχανική μάθηση. Ομαδοποιεί τα σημεία δεδομένων σε μια ένθετη ιεραρχία συστάδων με βάση την ομοιότητά τους, με

αποτέλεσμα μια δομή που μοιάζει με δέντρο που ονομάζεται δενδρογράμμα. Η ιεραρχική ομαδοποίηση μπορεί να ταξινομηθεί σε δύο κύριες προσεγγίσεις: αθροιστική (από κάτω προς τα πάνω) και διαιρετική (από πάνω προς τα κάτω) (Sneath & Sokal, 1973; Johnson, 1967).

- Συγκεντρωτική Ιεραρχική Ομαδοποίηση: Σε αυτήν την προσέγγιση από κάτω προς τα πάνω, κάθε σημείο δεδομένων σχηματίζει αρχικά τη δική του συστάδα. Ο αλγόριθμος συγχωνεύει επαναληπτικά το πλησιέστερο ζεύγος συστάδων μέχρι να σχηματιστεί ένα μοναδικό σύμπλεγμα που περιέχει όλα τα σημεία δεδομένων.
- Διαιρετική ιεραρχική ομαδοποίηση: Σε αυτήν την προσέγγιση από πάνω προς τα κάτω, όλα τα σημεία δεδομένων ανήκουν αρχικά σε ένα μόνο σύμπλεγμα. Ο αλγόριθμος χωρίζει επαναληπτικά τα συμπλέγματα σε μικρότερα με βάση την ανομοιότητα έως ότου κάθε σημείο δεδομένων σχηματίσει το δικό του σύμπλεγμα.

Αμφότερες οι αθροιστικές και οι διαιρετικές μέθοδοι ιεραρχικής ομαδοποίησης απαιτούν μια μέτρηση απόστασης για τη μέτρηση της ομοιότητας μεταξύ των σημείων δεδομένων. Τα κοινά κριτήρια σύνδεσης περιλαμβάνουν απλή σύνδεση (ελάχιστη απόσταση), πλήρη σύνδεση (μέγιστη απόσταση), μέση σύνδεση (μέση απόσταση) και σύνδεση Ward (ελάχιστη αύξηση διακύμανσης). Τέλος, ένα σημαντικό πλεονέκτημα της ιεραρχικής ομαδοποίησης είναι ότι δεν απαιτεί από τον χρήστη να καθορίσει τον αριθμό των συστάδων *a priori*. Αντίθετα, ο κατάλληλος αριθμός συστάδων μπορεί να προσδιοριστεί εξετάζοντας το δενδρογράμμα και εφαρμόζοντας ένα κατάλληλο όριο ή χρησιμοποιώντας έναν δείκτη εγκυρότητας συστάδας.

1.4 Εφαρμογές Μηχανικής Μάθησης

Στη σύγχρονη εποχή, η μηχανική μάθηση έχει αναδειχθεί σε μια τεχνολογία που έχει αναδιαμορφώσει διάφορες πτυχές της ανθρώπινης ζωής. Οι εφαρμογές της είναι εκτεταμένες, εκτείνονται από την επιστημονική έρευνα έως τις κοινωνικές λειτουργίες και έχουν τη δυνατότητα να επιλύσουν ορισμένες από τις πιο πιεστικές προκλήσεις της εποχής μας. παρακάτω κάνουμε μια επισκόπηση των ποικίλων εφαρμογών της

μηχανικής μάθησης τόσο στην επιστήμη όσο και στην κοινωνία, αναδεικνύοντας τη σημασία της και τον μετασχηματιστικό αντίκτυπο που έχει επιφέρει σε πολλούς τομείς:

- Η μηχανική μάθηση έχει φέρει επανάσταση στον τομέα της αναγνώρισης εικόνων, επιτρέποντας στους υπολογιστές να ερμηνεύουν και να λαμβάνουν αποφάσεις με βάση οπτικά δεδομένα. Οι προηγμένοι αλγόριθμοι, ιδίως εκείνοι που βασίζονται σε συνεπτυγμένα νευρωνικά δίκτυα (CNN) (LeCun et al., 2015; He et al., 2016), έχουν θέσει νέα σημεία αναφοράς σε εργασίες όπως η ανίχνευση αντικειμένων, η αναγνώριση προσώπου και η σημασιολογική κατάτμηση. Αυτές οι δυνατότητες έχουν ευρύ φάσμα εφαρμογών, όπως στην ιατρική απεικόνιση, τη διάγνωση ασθενειών, την ανακάλυψη φαρμάκων και την εξατομικευμένη ιατρική, όπου οι αλγόριθμοι μπορούν να εντοπίσουν ανωμαλίες όπως όγκους σε ακτίνες X ή μαγνητικές τομογραφίες (Rajkumar, Dean, & Kohane, 2019), έως τα αυτόνομα οχήματα που βασίζονται στην αναγνώριση εικόνων σε πραγματικό χρόνο για την ασφαλή πλοήγηση στους δρόμους (Du, Guo, & Simpson, 2019).
- Στους τομείς του λιανικού εμπορίου και του ηλεκτρονικού εμπορίου, η μηχανική μάθηση έχει γίνει μια τεχνολογία ακρογωνιαίος λίθου που βελτιώνει σημαντικά την εμπειρία των πελατών και τη λειτουργική αποδοτικότητα. Μία από τις πιο ορατές εφαρμογές είναι στα συστήματα συστάσεων, όπου οι αλγόριθμοι μηχανικής μάθησης αναλύουν τη συμπεριφορά και τις προτιμήσεις των πελατών για να προτείνουν εξατομικευμένα προϊόντα, αυξάνοντας έτσι τις πωλήσεις και τη δέσμευση των πελατών (Portugal, Alencar, & Cowan, 2018). Η εξυπηρέτηση πελατών είναι ένας άλλος τομέας που επωφελείται από τη μηχανική μάθηση, με τα chatbots που είναι ικανά να χειρίζονται μια σειρά από ερωτήματα πελατών σε πραγματικό χρόνο (Suta et al., 2020). Οι αλγόριθμοι ανίχνευσης απάτης βοηθούν στην ασφάλεια των συναλλαγών με τον εντοπισμό ασυνήθιστων μοτίβων που μπορεί να υποδηλώνουν δόλια δραστηριότητα (Chen, 2020).
- Μοντέλα βαθιάς μάθησης όπως τα RNN, τα δίκτυα μακράς βραχυπρόθεσμης μνήμης (LSTM) και οι μετασχηματιστές έχουν επιδείξει αξιοσημείωτη επιτυχία σε εργασίες NLP, συμπεριλαμβανομένης της μηχανικής μετάφρασης, της

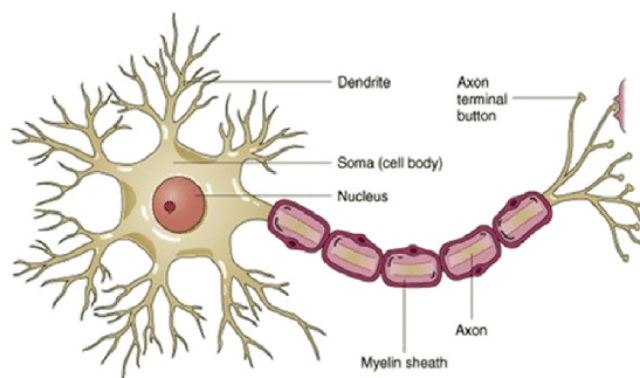
ανάλυσης συναισθημάτων και της απάντησης ερωτήσεων (Sutskever, Vinyals, & Le, 2014; Vaswani et al., 2017).

Οι εφαρμογές της μηχανικής μάθησης είναι πολυάριθμες και σε άλλα πεδία όπως στην εκπαίδευση, την κυβερνοασφάλεια, σε υπηρεσίες μετάφρασης, ρομποτική, στην έρευνα κ.ά. Οι εφαρμογές αυτές αναδεικνύουν τον ευρύ και μετασχηματιστικό αντίκτυπο της μηχανικής μάθησης σε πολλούς τομείς.

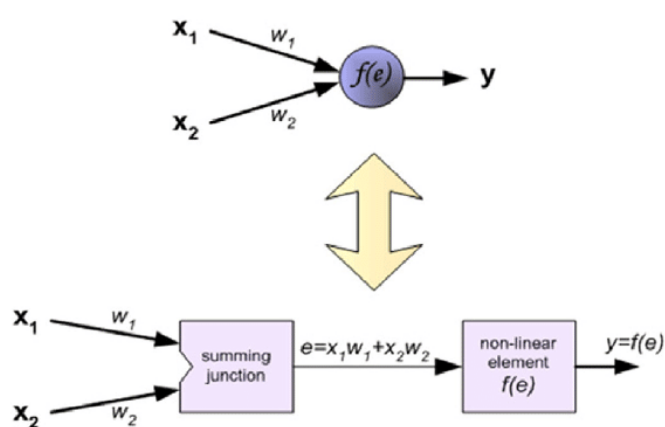
Κεφάλαιο 2 – Νευρωνικά Δίκτυα

2.1 Εισαγωγή

Ο όρος «νευρωνικό δίκτυο» έχει τις ρίζες του στις προσπάθειες εύρεσης μαθηματικών αναπαραστάσεων της επεξεργασίας πληροφοριών σε βιολογικά συστήματα (McCulloch και Pitts, 1943; Widrow and Hoff, 1960; Rosenblatt, 1962; Rumelhart et al., 1986). Στο ανθρώπινο νευρικό σύστημα τα νευρικά κύτταρα συνδέονται μεταξύ τους μέσω των συνάψεων. Η αντοχή των συνάψεων αλλάζει ανάλογα με τα εξωτερικά ερεθίσματα. Η βασική αρχή του τεχνητού νευρωνικού δικτύου, τα οποία θα αποκαλούμε απλώς νευρωνικά δίκτυα, βασίζεται σε αυτόν τον παραλληλισμό με τον μηχανισμό των βιολογικών συστημάτων.



(a) Biological neuron



(b) Artificial neural network

Εικόνα 9 Σύγκριση βιολογικού νευρώνα και τεχνητού νευρωνικού δικτύου.

Πηγή: (Kim et al., 2014)

Στα νευρωνικά δίκτυα κάθε είσοδος μιας υπολογιστικής μονάδας (νευρώνα/κόμβου) πολλαπλασιάζεται με έναν βάρος, και η υπολογιστική μονάδα με τη σειρά της χρησιμοποιεί τις εισόδους για να υπολογίσει μια συνάρτηση. Η εκμάθηση συμβαίνει μέσω της αλλαγής των βαρών. Στην διαδικασία εκπαίδευσης του νευρωνικού δικτύου τα δεδομένα εκπαίδευσης είναι ζευγάρια εισόδου-εξόδου, και το νευρωνικό δίκτυο αλλάζοντας τα βάρη των εισόδων προσπαθεί να παράγει τις σωστές εξόδους, μαθαίνοντας έτσι την συνάρτηση που περιγράφει τα ζευγάρια αυτά. Η διαφορά της εξόδου του NN από την έξοδο που πρέπει θεωρητικά να προβλέψει είναι το σφάλμα, και χρησιμοποιείται ώστε να αλλαχθούν τα βάρη. Μέσω πολλαπλών επαναλήψεων υπολογισμού της εξόδου και κατάλληλης αλλαγής των βαρών, το μοντέλο πρέπει να παράγει όλο και πιο ακριβείς προβλέψεις. Μάλιστα, το μοντέλο πρέπει για εισόδους που δεν έχει ξανασυναντήσει, να παράγει προβλέψεις με μεγάλη ακρίβεια. Αυτό λέγεται ικανότητα γενίκευσης του μοντέλου και είναι βασικός στόχος των μοντέλων της μηχανικής μάθησης, γενικεύοντας από προηγούμενα παραδείγματα σε νέα άγνωστα δεδομένα. Στην πιο απλή του μορφή, ένα νευρωνικό δίκτυο ισοδυναμεί με κλασσικούς αλγορίθμους εκμάθησης της μηχανικής μάθησης. Τα νευρωνικά δίκτυα διαφοροποιούνται σε μεγάλο βαθμό όταν οι βασικές υπολογιστικές μονάδες συνδυάζονται μεταξύ τους σχηματίζοντας διάφορες αρχιτεκτονικές, και τότε αποκτούν την δυνατότητα να μάθουν περίπλοκες συναρτήσεις.

Στα επόμενα κεφάλαια θα εξετάσουμε τις βασικές αρχές λειτουργίας των νευρωνικών δικτύων και θα εμβαθύνουμε σε μια αρχιτεκτονική, αυτή των συνελκτικών νευρωνικών δικτύων.

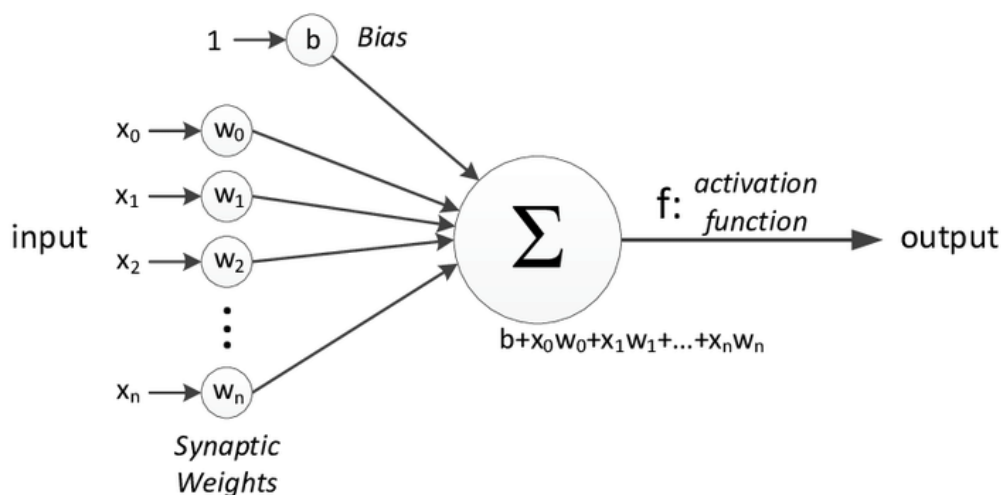
2.2 Βασική Αρχιτεκτονική των Νευρωνικών Δικτύων

Υπάρχουν νευρωνικά δίκτυα μονού επιπέδου (single layer) και δίκτυα πολλαπλών επιπέδων (multi layer). Η πρώτη μορφή νευρωνικού δικτύου που θα εξετάσουμε είναι μονού επιπέδου και ονομάζεται Perceptron. Ένα σύνολο εισόδων αντιστοιχίζεται απευθείας σε μια έξοδο μέσω μιας γενικευμένης γραμμικής συνάρτησης. Στα δίκτυα πολλών στρωμάτων οι νευρώνες είναι τοποθετημένοι ο ένας μετά τον άλλον, δηλαδή τα επίπεδα εισόδου και εξόδου είναι διαχωρισμένα με ένα σύνολο από άλλα άλλα που ονομάζονται κρυφά επίπεδα (hidden layers).

2.2.1 Δίκτυο Perceptron

Στον τομέα της τεχνητής νοημοσύνης και της μηχανικής μάθησης, το Perceptron κατέχει σημαντική θέση ως το θεμελιώδες δομικό στοιχείο των νευρωνικών δικτύων. Εισήχθη από τον Frank Rosenblatt (1958) και έθεσε τις βάσεις για την ανάπτυξη πιο περίπλοκων και ισχυρών μοντέλων βαθιάς μάθησης που βλέπουμε σήμερα. Αυτό το κεφάλαιο στοχεύει να διερευνήσει την αρχιτεκτονική, τον αλγόριθμο μάθησης και τον ρόλο του στο ευρύτερο πλαίσιο των νευρωνικών δικτύων. Το Perceptron είναι το απλούστερο δίκτυο ενός υπολογιστικού επιπέδου και αποτελεί το απλούστερο νευρωνικό δίκτυο. Σύμφωνα με τον Aggarwal (2023) το Perceptron αποτελείται από ένα μόνο επίπεδο εισόδου (input layer) και έναν κόμβο εξόδου. Συμβολίζοντας ένα παράδειγμα εκμάθησης ως $(\bar{X}, y)$ όπου $\bar{X} = [x_1, \dots, x_d]$ το διάνυσμα των d features και $y = \{-1, 1\}$ η παρατηρούμενη τιμή της δυαδικής μεταβλητής κλάσης (binary class variable). Κάθε κόμβος εισόδου έχει ακμές με βάρος που περιγράφεται στο διάνυσμα $\bar{W} = [w_1, \dots, w_d]$. Η γραμμική συνάρτηση $\bar{W} \cdot \bar{X} = \sum_{i=1}^d w_i \cdot x_i$ υπολογίζεται στον κόμβο εξόδου και χρησιμοποιώντας το πρόσημο του αποτελέσματος καθορίζεται η μεταβλητή κλάσης (εξαρτημένη μεταβλητή) ως εξής :

$$\hat{y} = \text{sign}\{\bar{W} \cdot \bar{X}\} = \text{sign}\left\{\sum_{i=1}^d w_i \cdot x_i\right\}$$



Εικόνα 10 Το απλούστερο μαθηματικό μοντέλο ενός νευρώνα που ονομάζεται Perceptron. Πηγή: (Rosenblatt, 1958)

Η συνάρτηση αυτή που χρησιμοποιείται συγκεκριμένα για δυαδική ταξινόμηση, έχει ως έξοδο τις τιμές $\{-1, +1\}$. Στην εξίσωση (1), η συνάρτηση προσήμου έχει τον ρόλο της συνάρτησης ενεργοποίησης (activation function). Με μαθηματικούς όρους, η συνάρτηση προσήμου ορίζεται ως εξής:

$$\hat{y} = \text{sign}\{\bar{W} \cdot \bar{X} + b\} = \text{sign}\left\{\sum_{j=1}^d w_j x_j + b\right\}$$

Η συνάρτηση προσήμου σε ένα perceptron βοηθά να αποφασιστεί αν μια είσοδος ανήκει σε μια ή σε μια άλλη κλάση. Δίνει στην έξοδο +1 ή -1. Η συνάρτηση είναι απλή και γρήγορη στον υπολογισμό, αλλά δεν είναι διαφορίσιμη στο σημείο όπου μεταβαίνει από το -1 στο +1 (στο μηδέν). Εξαιτίας αυτού του περιορισμού, δεν είναι ιδανική για μεθόδους που τελειοποιούν το μοντέλο κάνοντας μικρές προσαρμογές, όπως η κάθοδος κλίσης (gradient descent). Αυτός είναι ο λόγος για τον οποίο τα νεότερα μοντέλα χρησιμοποιούν συχνά άλλες συναρτήσεις που δεν έχουν αυτό το πρόβλημα, όπως οι συναρτήσεις sigmoid ή ReLU, οι οποίες μπορούν να παραγωγηθούν πιο εύκολα.

Μια συνάρτηση ενεργοποίησης είναι μια συνάρτηση που μετασχηματίζει τα δεδομένα εισόδου χρησιμοποιώντας μη γραμμική μέθοδο. Χωρίς αυτή, δεν υπάρχει καμία διαφορά από το να εκτελούμε μια σειρά από παλινδρομήσεις. Οι συναρτήσεις Sigmoid και ReLU είναι οι πιο συνηθισμένες συναρτήσεις ενεργοποίησης. Συχνά συναντάμε στην βιβλιογραφία έναν όρο, που ονομάζεται bias και ενσωματώνεται στην εξίσωση ως εξής:

$$\hat{y} = \text{sign}\{\bar{W} \cdot \bar{X} + b\} = \text{sign}\left\{\sum_{j=1}^d w_j x_j + b\right\}$$

Ο ρόλος του είναι να “προσομοιάσει” το αμετάβλητο μέρος της συνάρτησης το οποίο προκύπτει από την εκάστοτε μορφή των δεδομένων εισόδου και μπορεί να αγνοηθεί στα επόμενα κεφάλαια, καθώς μπορεί πάντα να ενσωματωθεί στους νευρώνες εισόδου. Το σφάλμα του υπολογισμού της συνάρτησης που δείξαμε είναι $E(\bar{X}) = y - \hat{y}$ και παίρνει τις τιμές $\{-2, 0, 2\}$. Όταν το σφάλμα είναι διάφορο του μηδενός τότε γίνεται μια διορθωτική ενημέρωση των βαρών προς την κλίση του σφάλματος (error gradient). Γενικά, ο στόχος του δικτύου είναι να ελαχιστοποιήσει το σφάλμα, οπότε επιλέγοντας

για παράδειγμα αυθαίρετα τον κανόνα ελαχίστων τετραγώνων μπορούμε να διατυπώσουμε την εξής συνάρτηση:

$$\text{Minimize}_{\bar{W}} L = \sum_{(\bar{X}, y) \in \mathcal{D}} (y - \hat{y})^2 = \sum_{(\bar{X}, y) \in \mathcal{D}} (y - \text{sign}\{\bar{W} \cdot \bar{X}\})^2$$

Τέτοιου είδους συναρτήσεις ελαχιστοποίησης σφάλματος ονομάζονται συναρτήσεις κόστους. Σχηματίζεται δηλαδή ένα πρόβλημα ελαχιστοποίησης μιας συνάρτησης σφάλματος ως προς το διάνυσμα των παραμέτρων του δικτύου. Πολλές μέθοδοι αριθμητικής ελαχιστοποίησης βασίζονται στις μερικές παραγώγους και η πιο δημοφιλής μέθοδος για τα νευρωνικά δίκτυα είναι η gradient descent (κάθοδος βασισμένη στην κλίση).

2.2.2 Πολυεπίπεδα NN

Τα πολυεπίπεδα νευρωνικά δίκτυα (Multi-Layer Neural Networks - MLNN) αναφέρονται σε μια κατηγορία μοντέλων βαθιάς μάθησης που ενσωματώνουν πολλαπλά επίπεδα αναπαράστασης και αφαίρεσης στις αρχιτεκτονικές τους. Αυτά τα δίκτυα έχουν σχεδιαστεί για να μαθαίνουν πολύπλοκες, ιεραρχικές αναπαραστάσεις των δεδομένων εισόδου, αξιοποιώντας τη δύναμη των πολυεπίπεδων, μη γραμμικών μετασχηματισμών. Τα MLNN έχουν εφαρμοστεί με επιτυχία σε ποικίλες εφαρμογές, όπως η όραση υπολογιστή, η επεξεργασία φυσικής γλώσσας και η ενισχυτική μάθηση. Η αρχιτεκτονική των MLNN ποικίλλει ανάλογα με τον συγκεκριμένο τύπο μοντέλου βαθιάς μάθησης που χρησιμοποιείται. Τα Συνελκτικά νευρωνικά δίκτυα (CNN) έχουν σχεδιαστεί για να αναγνωρίζουν μοτίβα σε μια δομή δεδομένων που μοιάζει με πλέγμα, όπως μια εικόνα. Αυτό το επιτυγχάνουν εφαρμόζοντας φίλτρα (γνωστά και ως πυρήνες) σε διάφορα τμήματα της εισόδου. Αυτά τα φίλτρα μπορούν να ανιχνεύουν απλά χαρακτηριστικά όπως ακμές και γωνίες στα αρχικά στρώματα και πιο σύνθετα χαρακτηριστικά όπως σχήματα στα βαθύτερα στρώματα. Το δίκτυο προσαρμόζει αυτά τα φίλτρα κατά τη διάρκεια της διαδικασίας εκπαίδευσης για να βελτιώσει την απόδοσή του (Li et al., 2020). Παρακάτω συνοψίζουμε τα βασικά χαρακτηριστικά τους:

- Επίπεδο εισόδου: Αυτό το επίπεδο λαμβάνει τα ακατέργαστα δεδομένα εισόδου, όπως εικόνες ή χάρτες χαρακτηριστικών.

- Συνελικτικά επίπεδα: Αυτά τα επίπεδα εκτελούν συναρτήσεις συνέλιξης χρησιμοποιώντας μικρά φίλτρα για την εξαγωγή τοπικών χαρακτηριστικών από τα δεδομένα εισόδου. Ανιχνεύουν μοτίβα όπως ακμές, γωνίες και υφές.
- Συναρτήσεις ενεργοποίησης: Οι μη γραμμικές συναρτήσεις ενεργοποίησης, όπως η ReLU (Διορθωμένη Γραμμική Μονάδα), εφαρμόζονται βάσει στοιχείων στην έξοδο των συνελικτικών επιπέδων, εισάγοντας μη γραμμικότητα στο δίκτυο.
- Pooling Layers: Αυτά τα επίπεδα εκτελούν συναρτήσεις downsampling, όπως max-pooling ή average pooling, για να μειώσουν τις χωρικές διαστάσεις των χαρτών χαρακτηριστικών, καθιστώντας το δίκτυο πιο εύρωστο σε μικρές παραλλαγές στα δεδομένα εισόδου.
- Πλήρως συνδεδεμένα επίπεδα: Αυτά τα επίπεδα, που συνήθως τοποθετούνται στο τέλος του δικτύου, μετατρέπουν τα χαρακτηριστικά υψηλού επιπέδου που εξάγονται από τα επίπεδα συνέλιξης και συγκέντρωσης σε τελικές εξόδους, όπως πιθανότητες κλάσης για εργασίες ταξινόμησης.
- Επίπεδο εξόδου: Αυτό το επίπεδο παράγει τις τελικές προβλέψεις ή βαθμολογίες, συνήθως χρησιμοποιώντας μια συνάρτηση ενεργοποίησης softmax για προβλήματα πολλαπλών κλάσεων.

Τα επαναλαμβανόμενα νευρωνικά δίκτυα (Recurrent Neural Networks, RNN) εισήχθησαν από τους David Rumelhart, Geoffrey Hinton και Ronald Williams το 1986. Παρουσίασαν τον αλγόριθμο οπισθοδιάδοσης για την εκπαίδευση των RNNs στην πρωτοποριακή εργασία τους, θέτοντας τα θεμέλια για την ανάπτυξη και την εφαρμογή των RNNs σε διάφορους τομείς. Ωστόσο, αξίζει να σημειωθεί ότι η έννοια των συνδέσεων ανατροφοδότησης στα νευρωνικά δίκτυα, η οποία είναι κεντρική για τα RNN, είχε διερευνηθεί και πριν από αυτό σε διάφορες μορφές. Το έργο των Rumelhart, Hinton και Williams επισημοποίησε την έννοια και παρείχε έναν ισχυρό τρόπο εκπαίδευσης τέτοιων δικτύων. Τα επαναλαμβανόμενα νευρωνικά δίκτυα (Recurrent Neural Networks - RNN) αποτελούν έναν τύπο τεχνητού νευρωνικού δικτύου που έχει σχεδιαστεί για την επεξεργασία διαδοχικών δεδομένων, όπως χρονοσειρές ή κείμενο. Κύριο χαρακτηριστικό των RNN είναι η ικανότητά τους να διατηρούν και να

χρησιμοποιούν πληροφορία από προηγούμενες εισόδους, κάνοντάς τα κατάλληλα για εργασία με δεδομένα που έχουν χρονική διάσταση ή συνεχόμενη ακολουθία. Αυτή η ικανότητα τους τους καθιστά χρήσιμα σε εφαρμογές όπως η πρόβλεψη χρονοσειρών, η αναγνώριση φωνής, η μηχανική μετάφραση και πολλές άλλες που συνδέονται με την ανάλυση σειρών δεδομένων (Unadkat, Ciocoiu, & Medsker, 2001, pp. 64-67). Τα κύρια μέρη των RNN είναι:

- Επίπεδο εισόδου: Αυτό το επίπεδο λαμβάνει τα δεδομένα εισόδου με διαδοχικό τρόπο.
- Επαναλαμβανόμενα επίπεδα: Αυτά τα επίπεδα διατηρούν μια κρυφή κατάσταση που ενημερώνεται σε κάθε χρονικό βήμα με βάση τόσο την τρέχουσα είσοδο όσο και την προηγούμενη κρυφή κατάσταση. Αυτό επιτρέπει στο δίκτυο να μάθει χρονικές εξαρτήσεις και μοτίβα στα δεδομένα.
- Συναρτήσεις ενεργοποίησης: Οι μη γραμμικές συναρτήσεις ενεργοποίησης εφαρμόζονται στην έξοδο των επαναλαμβανόμενων επιπέδων.
- Επίπεδο εξόδου: Αυτό το επίπεδο παράγει τις τελικές προβλέψεις ή βαθμολογίες σε κάθε χρονικό βήμα, συνήθως χρησιμοποιώντας μια συνάρτηση ενεργοποίησης softmax για προβλήματα πολλαπλών κλάσεων.

Διάφορες παραλλαγές των RNN (π.χ., δίκτυα μακράς βραχυπρόθεσμης μνήμης και Gated Recurrent Units) έχουν αναπτυχθεί για την αντιμετώπιση του προβλήματος της εξαφάνισης της κλίσης (Vanishing Gradient Problem) και τη βελτίωση της εκμάθησης εξαρτήσεων μεγάλης εμβέλειας. Πιο συγκεκριμένα, το πρόβλημα της εξαφανιζόμενης κλίσης προκύπτει κατά την εκπαίδευση των RNN, ιδίως όταν πρόκειται για μακροχρόνιες εξαρτήσεις σε διαδοχικά δεδομένα Hochreiter (1998). Όπως πολλά νευρωνικά δίκτυα, τα RNN χρησιμοποιούν τεχνικές βελτιστοποίησης με βάση την κλίση, όπως η κάθοδος κλίσης, για να ενημερώσουν τα εσωτερικά τους βάρη κατά τη διάρκεια της εκπαίδευσης. Η ιδέα είναι η ελαχιστοποίηση μιας συνάρτησης απώλειας με την προσαρμογή των βαρών με βάση τις κλίσεις της συνάρτησης απώλειας σε σχέση με αυτά τα βάρη. Αυτή η διαδικασία είναι γνωστή ως οπισθοδιάδοση. Το πρόβλημα της εξαφανιζόμενης κλίσης εμφανίζεται επειδή, καθώς οι κλίσεις διαδίδονται οπισθοδρομικά μέσω πολλών χρονικών βημάτων, τείνουν να γίνουν εξαιρετικά μικρές.

Αυτό συμβαίνει επειδή οι κλίσεις υπολογίζονται ως γινόμενα των παραγώγων των συναρτήσεων ενεργοποίησης και των πινάκων βαρών. Στα RNNs, μια κοινή συνάρτηση ενεργοποίησης είναι η σιγμοειδής συνάρτηση, Αυτή η συνάρτηση έχει την ιδιότητα ότι όταν οι εισόδοι της είναι μεγάλες, η έξοδός της περιορίζεται, δηλαδή δεν αυξάνεται απεριόριστα. Αντίθετα, καθώς οι εισόδοι μειώνονται προς το μηδέν, η έξοδος της σιγμοειδούς συνάρτησης προσεγγίζει το μηδέν. Όταν πολλαπλασιάζονται οι μικρές παράγωγοι για πολλά χρονικά βήματα, η συνολική κλίση γίνεται εξαιρετικά μικρή. Αυτό σημαίνει ότι κατά την εκπαίδευση του RNN, η κλίση που μεταφέρεται πίσω στο δίκτυο μπορεί να γίνεται τόσο μικρή που τα βάρη των συνδέσεων δεν ενημερώνονται επαρκώς. Αυτό οδηγεί σε δυσκολία στην εκμάθηση μακροπρόθεσμων εξαρτήσεων στα δεδομένα, καθώς η κλίση για αυτές τις εξαρτήσεις γίνεται σχεδόν ανύπαρκτη Hochreiter (1998). Οι μετασχηματιστές, ειδικά σχεδιασμένοι για εργασίες επεξεργασίας φυσικής γλώσσας, αποτελούνται από τα ακόλουθα κύρια στοιχεία:

- Επίπεδο εισόδου: Αυτό το επίπεδο λαμβάνει τα δεδομένα εισόδου, όπως το κείμενο με διακριτικό, και το αναπαριστά ως συνεχή διανύσματα χρησιμοποιώντας ένα επίπεδο ενσωμάτωσης.
- Μπλοκ κωδικοποιητή και αποκωδικοποιητή: Τα δίκτυα μετασχηματιστών ακολουθούν συνήθως μια αρχιτεκτονική κωδικοποιητή-αποκωδικοποιητή. Τόσο το μπλοκ κωδικοποιητή όσο και του αποκωδικοποιητή αποτελούνται από πολλαπλά επίπεδοτα μηχανισμών αυτοπροσοχής και νευρωνικών δικτύων προώθησης. Οι μηχανισμοί αυτοπροσοχής επιτρέπουν στο δίκτυο να σταθμίζει τη σημασία διαφορετικών τμημάτων της ακολουθίας εισόδου όταν κάνει προβλέψεις.
- Κωδικοποίηση θέσης: Εφόσον οι μετασχηματιστές δεν έχουν εγγενή διαδοχική δομή, χρησιμοποιούν κωδικοποίηση θέσης για να εισάγουν πληροφορίες σχετικά με τις θέσεις των διακριτικών εισόδου.
- Επίπεδο εξόδου: Αυτό το επίπεδο παράγει τις τελικές προβλέψεις ή βαθμολογίες, συνήθως χρησιμοποιώντας μια συνάρτηση ενεργοποίησης softmax για προβλήματα πολλαπλών κλάσεων.

Η αρχιτεκτονική των MLNN ποικίλλει σημαντικά ανάλογα με τον συγκεκριμένο τύπο δικτύου και το πρόβλημα που έχει σχεδιαστεί να επιλύει. Τα παραπάνω παραδείγματα παρέχουν μια επισκόπηση ορισμένων κοινών αρχιτεκτονικών MLNN και των κύριων στοιχείων τους.

2.2.3 Εκπαίδευση Δικτύου με **backpropagation**

Ο backpropagation, συντομογραφία για την "προς τα πίσω διάδοση σφαλμάτων", είναι ένας εποπτευόμενος αλγόριθμος εκμάθησης που χρησιμοποιείται για την εκπαίδευση τεχνητών νευρωνικών δικτύων, συμπεριλαμβανομένων των πολυεπίπεδων νευρωνικών δικτύων. Ο αλγόριθμος προσαρμόζει τα βάρη και τις σταθερές του δικτύου με σκοπό να ελαχιστοποιήσει το σφάλμα μεταξύ της προβλεπόμενης εξόδου και των πραγματικών τιμών. Ο αλγόριθμος αυτός αποτελείται από δύο βασικά βήματα, την εμπρόσθια διάδοση και την οπισθοδιάδοση. Ο αλγόριθμος backpropagation επαναλαμβάνει αυτά τα δύο βήματα πολλές φορές κατά τη διάρκεια της εκπαίδευσης, προσαρμόζοντας σταδιακά τα βάρη του δικτύου έτσι ώστε να βελτιστοποιήσει την ακρίβεια των προβλέψεων του.

Ο αλγόριθμος περιλαμβάνει δύο βασικά βήματα:

- Εμπρόσθια διάδοση: Κατά τη διάρκεια αυτής της φάσης, τα δεδομένα εισόδου διαβιβάζονται μέσω του νευρωνικού δικτύου από το επίπεδο εισόδου στο επίπεδο εξόδου. Κάθε επίπεδο υπολογίζει ένα σταθμισμένο άθροισμα των εισόδων του, εφαρμόζει μια μη γραμμική συνάρτηση ενεργοποίησης και προωθεί το αποτέλεσμα στο επόμενο επίπεδο. Αυτή η διαδικασία συνεχίζεται μέχρι να επιτευχθεί το επίπεδο εξόδου και να γίνει μια πρόβλεψη.
- Διάδοση προς τα πίσω: Μετά το πέραςμα προς τα εμπρός, το σφάλμα μεταξύ της προβλεπόμενης εξόδου και των πραγματικών τιμών στόχου υπολογίζεται χρησιμοποιώντας μια συνάρτηση απώλειας (π.χ. μέσο τετραγωνικό σφάλμα για παλινδρόμηση ή διασταυρούμενη εντροπία για ταξινόμηση). Ο στόχος της μαθησιακής διαδικασίας είναι να ελαχιστοποιήσει αυτή τη λειτουργία απώλειας. Κατά τη διάρκεια του βήματος διάδοσης προς τα πίσω, η κλίση της συνάρτησης απώλειας σε σχέση με κάθε βάρος και προκατάληψη υπολογίζεται χρησιμοποιώντας τον κανόνα της αλυσίδας του λογισμού. Αυτή η κλίση

υποδεικνύει πόσο μια αλλαγή σε ένα συγκεκριμένο βάρος ή μεροληψία θα επηρεάσει το συνολικό σφάλμα.

Στη συνέχεια, οι διαβαθμίσεις χρησιμοποιούνται για την ενημέρωση των βαρών και των προκαταλήψεων του δικτύου χρησιμοποιώντας έναν αλγόριθμο βελτιστοποίησης, όπως η στοχαστική κλίση κατάβασης (SGD) ή πιο προηγμένους βελτιστοποιητές όπως ο Adam ή το RMSProp. Ο ρυθμός εκμάθησης, μια υπερπαράμετρος, καθορίζει το μέγεθος βήματος των ενημερώσεων. Ολόκληρη αυτή η διαδικασία διάδοσης προς τα εμπρός και προς τα πίσω εκτελείται επαναληπτικά για πολλαπλές εποχές (διέρχεται από ολόκληρο το σύνολο δεδομένων) μέχρι το σφάλμα να συγκλίνει σε μια ελάχιστη τιμή ή να ικανοποιηθεί ένα προκαθορισμένο κριτήριο για τη διακοπή της διαδικασίας.

2.2.4 Προβλήματα κατά την εκπαίδευση NN

Κατά τη διαδικασία εκμάθησης ενός νευρωνικού δικτύου, μπορεί να προκύψουν αρκετές προκλήσεις και ζητήματα που μπορεί να εμποδίσουν την απόδοση ή τη σύγκλιση του μοντέλου. Ένα τέτοιο πρόβλημα είναι το πρόβλημα της εξαφάνισης κλίσης (the vanishing gradient problem) (Hochreiter et al., 2001). Αυτό το ζήτημα προκύπτει κατά την εκπαίδευση σε βαθιά νευρωνικά δίκτυα με πολλά επίπεδα, ειδικά σε επαναλαμβανόμενα νευρωνικά δίκτυα (RNN). Κατά τη διάρκεια της αντίστροφης διάδοσης, οι διαβαθμίσεις της συνάρτησης απώλειας μπορεί να γίνουν πολύ μικρές καθώς διαδίδονται μέσω του δικτύου. Κατά συνέπεια, οι ενημερώσεις βάρους γίνονται αμελητέες, με αποτέλεσμα το δίκτυο να σταματήσει να μαθαίνει ή να μαθαίνει πολύ αργά. Τα δίκτυα μακροπρόθεσμης βραχυπρόθεσμης μνήμης (LSTM) και οι Gated Recurrent Units (GRU) έχουν προταθεί για την λύση του προβλήματος της εξαφάνισης της κλίσης στα RNN (Hochreiter and Schmidhuber, 1997).

Επίσης, το πρόβλημα εκρηκτικής κλίσης (exploding gradient) όπως αναφέρουν οι Pascanu et al. (2013). Αυτό το πρόβλημα είναι το αντίθετο από το πρόβλημα της κλίσης εξαφάνισης, όπου οι διαβαθμίσεις γίνονται πολύ μεγάλες κατά τη διάδοση προς τα πίσω. Αυτό μπορεί να οδηγήσει σε ασταθή μάθηση και να προκαλέσει απόκλιση των βαρών. Το απόκομμα κλίσης είναι μια τεχνική που χρησιμοποιείται για την αποφυγή έκρηξης κλίσεων περιορίζοντας τη μέγιστη τιμή κλίσης κατά τη διάρκεια της προπόνησης.

Ένα ακόμη πρόβλημα είναι η υπερπροσαρμογή (overfitting) δεδομένων που συμβαίνει όταν το νευρωνικό δίκτυο μαθαίνει πολύ καλά τα δεδομένα εκπαίδευσης (Hawkins, 2004), καταγράφοντας θόρυβο και άσχετα μοτίβα αντί για τις υποκείμενες σχέσεις στα δεδομένα. Ως αποτέλεσμα, το μοντέλο έχει κακή απόδοση σε αόρατα δεδομένα. Τεχνικές όπως η κανονικοποίηση (regularization) (L1, L2 ή εγκατάλειψη-dropout), η πρόωρη διακοπή (early stopping) και η αύξηση δεδομένων μπορούν να βοηθήσουν στην αποφυγή υπερβολικής προσαρμογής (Srivastava et al., 2014). Αντίστοιχο πρόβλημα είναι και η υποπροσαρμογή, όταν το νευρωνικό δίκτυο αποτυγχάνει να καταγράψει τα υποκείμενα μοτίβα στα δεδομένα, με αποτέλεσμα κακή απόδοση τόσο στα δεδομένα εκπαίδευσης όσο και στα δεδομένα δοκιμής. Αυτό μπορεί να οφείλεται σε ένα υπερβολικά απλοϊκό μοντέλο ή σε ανεπαρκή εκπαίδευση. Για την αντιμετώπιση της υποπροσαρμογής, μπορεί κανείς να αυξήσει τη χωρητικότητα του μοντέλου (π.χ. να προσθέσει περισσότερα επίπεδα ή νευρώνες), να εκπαιδεύσει το μοντέλο για περισσότερες εποχές ή να χρησιμοποιήσει έναν πιο εξελιγμένο αλγόριθμο βελτιστοποίησης.

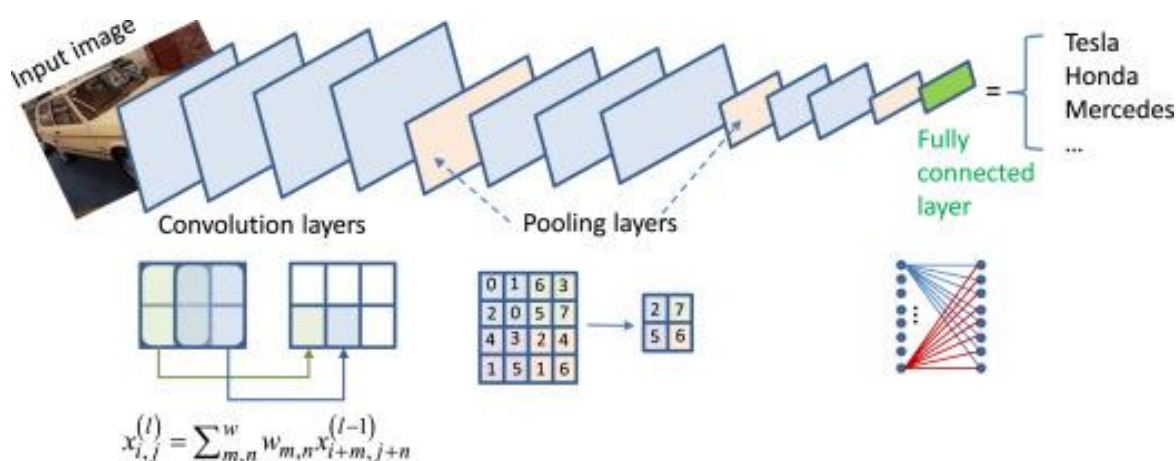
Τέλος, υπάρχουν και τα τοπικά ελάχιστα (Dauphin et al., 2014), όταν κατά τη διάρκεια της εκπαίδευσης, ο αλγόριθμος βελτιστοποίησης μπορεί να κολλήσει σε ένα τοπικό ελάχιστο της συνάρτησης απώλειας, εμποδίζοντας το μοντέλο να φτάσει στο παγκόσμιο ελάχιστο και να πετύχει βέλτιστη απόδοση.

2.3 Προηγμένες αρχιτεκτονικές νευρωνικών δικτύων για συστήματα συστάσεων

Στο κεφάλαιο αυτό εξετάζουμε μερικές τρεις προηγμένες αρχιτεκτονικές νευρωνικών δικτύων, τα συνελικτικά νευρωνικά δίκτυα (CNNs), τα επαναλαμβανόμενα (RNNs), και τα Long Short-Term Memory (LSTM) δίκτυα. Σημειώνουμε ότι οι αρχιτεκτονικές αυτές αποτελούν ένα μόνο κομμάτι των προηγμένων αρχιτεκτονικών. Μερικές αρχιτεκτονικές που δεν θα συζητηθούν είναι οι αυτοκωδικοποιητές (Autoencoders), οι μετασχηματιστές (transformers), οι μηχανισμοί προσοχής (GAT) και τα παραγωγικά αντιπαραθετικά δίκτυα (Generative adversarial networks).

2.3.1 Convolutional Neural Networks (CNN)

Τα συνελκτικά νευρωνικά δίκτυα (CNN) είναι μια κατηγορία μοντέλων βαθιάς μάθησης που χρησιμοποιούνται κυρίως για εργασίες όρασης υπολογιστή, όπως ταξινόμηση εικόνων, ανίχνευση αντικειμένων και σημασιολογική τμηματοποίηση. Τα CNN εμπνεύστηκαν από την οργάνωση και τη λειτουργία του ανθρώπινου οπτικού φλοιού και η ανάπτυξή τους μπορεί να εντοπιστεί στη δουλειά των Hubel και Wiesel στη δεκαετία του 1960 (Hubel & Wiesel, 1962). Η πρωτοποριακή εργασία του Yann LeCun και των συναδέλφων του στα τέλη της δεκαετίας του 1980, η οποία εισήγαγε την αρχιτεκτονική LeNet-5, σηματοδότησε την αρχή των σύγχρονων CNN (LeCun et al., 1998). Ένα CNN αποτελείται από πολλαπλά επίπεδα, συμπεριλαμβανομένων των συνελκτικών επιπέδων, των επιπέδων συγκέντρωσης και των πλήρως συνδεδεμένων επιπέδων. Το κύριο δομικό στοιχείο, το συνελκτικό επίπεδο, εφαρμόζει μια σειρά από φίλτρα ή πυρήνες στην εικόνα εισόδου ή στο χάρτη χαρακτηριστικών για να μάθει τοπικά μοτίβα, όπως ακμές ή υφές. Τα επίπεδα συγκέντρωσης χρησιμοποιούνται για τη μείωση των χωρικών διαστάσεων των χαρτών χαρακτηριστικών, διατηρώντας παράλληλα τις πιο σημαντικές πληροφορίες. Τέλος, χρησιμοποιούνται πλήρως συνδεδεμένα επίπεδα για να συνδυάσουν τα μαθησιακά χαρακτηριστικά και να παράγουν την τελική έξοδο, όπως πιθανότητες κλάσης στην περίπτωση ταξινόμησης εικόνων.



Εικόνα 11 Απλοποιημένο παράδειγμα κλασικού συνελκτικού νευρωνικού δικτύου.

Πηγή: (Djordjevic, 2021)

Το βασικό πλεονέκτημα των CNN έναντι των παραδοσιακών πλήρως συνδεδεμένων νευρωνικών δικτύων είναι η ικανότητά τους να εκμεταλλεύονται τη χωρική δομή και

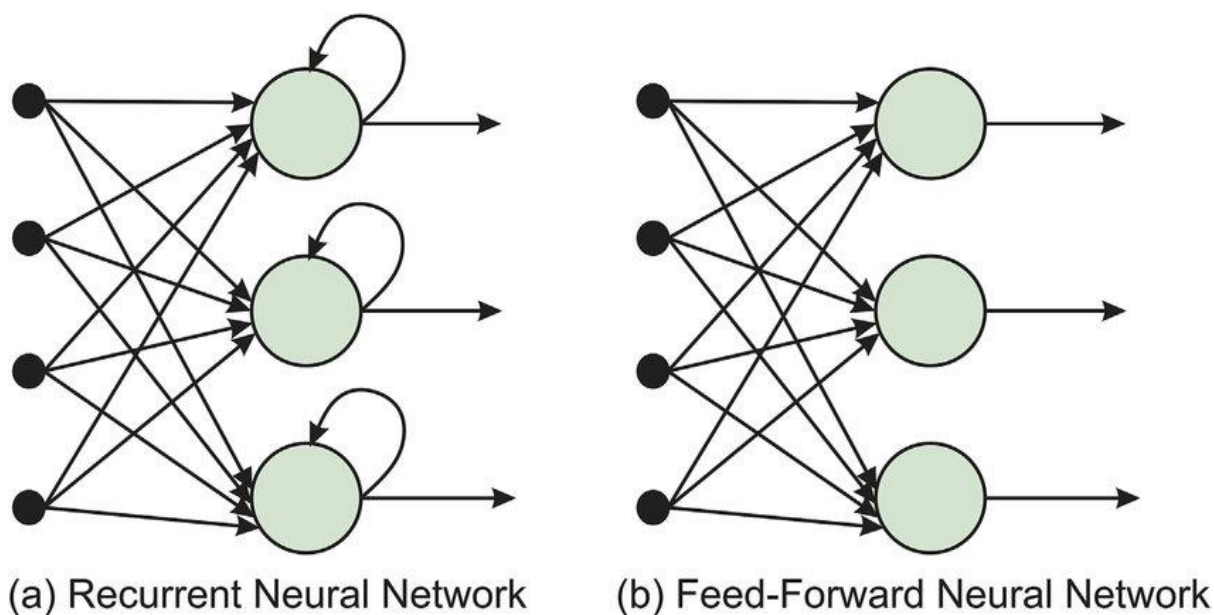
την ιεραρχία στα δεδομένα εισόδου μέσω της κατανομής βάρους και της τοπικής συνδεσιμότητας. Αυτό καθιστά τα CNN πιο αποτελεσματικά και στιβαρά, απαιτώντας λιγότερες παραμέτρους και μειώνοντας τον κίνδυνο υπερβολικής τοποθέτησης. Από την ανάπτυξη του LeNet-5, έχουν προταθεί πολυάριθμες αρχιτεκτονικές του CNN, συμπεριλαμβανομένων των διάσημων AlexNet (Krizhevsky, Sutskever, & Hinton, 2012), VGG (Simonyan & Zisserman, 2014), GoogLeNet/Inception (Szegedy et al., 2015) και ResNet (He, Zhang, Ren, & Sun, 2016), το καθένα βελτιώνεται σε σχέση με τους προκατόχους του όσον αφορά την ακρίβεια και την αποτελεσματικότητα.

2.3.2 Recurrent Neural Networks (RNN)

Τα επαναλαμβανόμενα νευρωνικά δίκτυα (RNN) είναι μια κατηγορία τεχνητών νευρωνικών δικτύων που έχουν σχεδιαστεί για την επεξεργασία διαδοχικών δεδομένων, όπως χρονοσειρές, φυσική γλώσσα ή ηχητικά σήματα. Τα RNN εισήχθησαν για πρώτη φορά από τους David Rumelhart, Geoffrey Hinton και Ronald J. Williams το 1986, ως ένας τρόπος αντιμετώπισης των περιορισμών των νευρωνικών δικτύων τροφοδοσίας στον χειρισμό ακολουθιών (Rumelhart, Hinton, & Williams, 1986). Σε αντίθεση με τα συμβατικά νευρωνικά δίκτυα, τα RNN περιέχουν βρόχους ανάδρασης που τους επιτρέπουν να διατηρήσουν μια εσωτερική κατάσταση ή μνήμη. Αυτοί οι βρόχοι επιτρέπουν στα RNN να μαθαίνουν μοτίβα σε διάφορα χρονικά βήματα, καθιστώντας τα ιδιαίτερα κατάλληλα για εργασίες που περιλαμβάνουν ακολουθίες, όπως η μοντελοποίηση γλώσσας, η αναγνώριση ομιλίας και η αυτόματη μετάφραση. Μία από τις βασικές προκλήσεις που αντιμετωπίζουν τα RNN είναι το πρόβλημα της κλίσης που εξαφανίζεται και εκρήγνυται, το οποίο εμποδίζει την εκμάθηση εξαρτήσεων μεγάλης εμβέλειας σε ακολουθίες. Για την αντιμετώπιση αυτού του ζητήματος, αναπτύχθηκαν προηγμένες αρχιτεκτονικές RNN, όπως τα δίκτυα Long-Short-Term Memory (LSTM) (Hochreiter & Schmidhuber, 1997) και Gated Recurrent Units (GRUs) (Cho et al., 2014). Αυτές οι αρχιτεκτονικές εισάγουν μηχανισμούς πύλης που επιτρέπουν πιο αποτελεσματική μάθηση και διατήρηση μακροπρόθεσμων εξαρτήσεων.

Στην παρακάτω εικόνα αναπαριστάται η σύγκριση μεταξύ επαναλαμβανόμενου νευρωνικού δικτύου (RNN) και νευρωνικού δικτύου εμπρόσθιας τροφοδοσίας

(FFNN). Παρατηρείται ότι στο FFNN υπάρχει μόνο μία κατεύθυνση για να μετακινηθούν τα δεδομένα, ενώ στο RNN υπάρχει ένας βρόχος ανάδρασης.



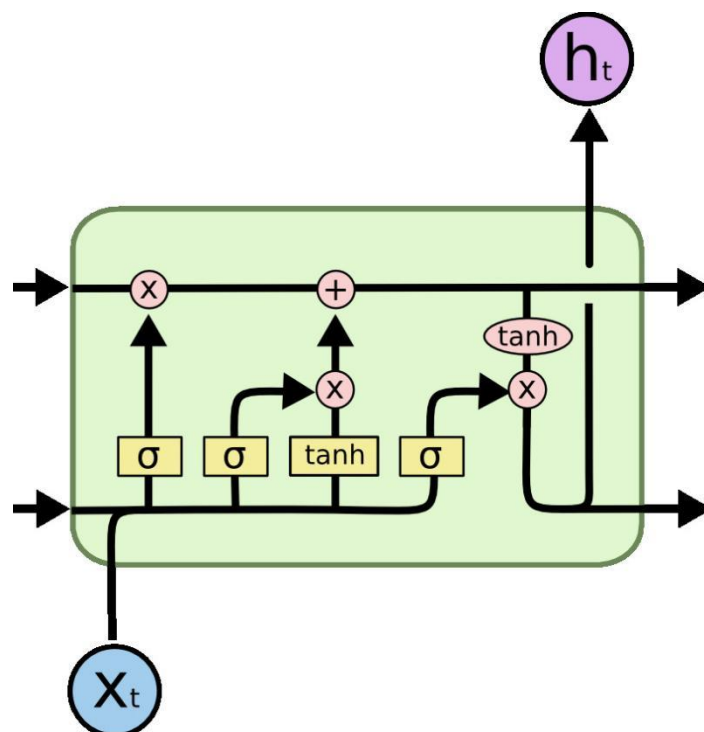
Εικόνα 12 Σύγκριση μεταξύ επαναλαμβανόμενου νευρωνικού δικτύου (RNN) και νευρωνικού δικτύου εμπρόσθιας τροφοδοσίας (FFNN).

Πηγή: (Eliasy & Przychodzen, 2020)

2.3.3 Long Short-Term Memory (LSTM)

Τα νευρωνικά δίκτυα τύπου Long Short-Term Memory (LSTM) είναι μια παραλλαγή των RNNs που έχει σχεδιαστεί ειδικά για την αντιμετώπιση του προβλήματος διαβάθμισης που εξαφανίζεται και εκρήγνυται που εμφανίζεται κατά την εκμάθηση εξαρτήσεων μεγάλης εμβέλειας σε διαδοχικά δεδομένα (Hochreiter & Schmidhuber, 1997). Τα LSTM έχουν γίνει δημοφιλή για ένα ευρύ φάσμα εργασιών αλληλουχίας σε ακολουθία, όπως η μοντελοποίηση γλώσσας, η αναγνώριση ομιλίας και η αυτόματη μετάφραση, λόγω της ικανότητάς τους να καταγράφουν μακροπρόθεσμες εξαρτήσεις πιο αποτελεσματικά από τα παραδοσιακά RNN. Εισάγουν μια κυψέλη μνήμης και ένα σύνολο μηχανισμών πύλης που ελέγχουν τη ροή πληροφοριών μέσα στο δίκτυο. Η αρχιτεκτονική περιλαμβάνει τρεις πύλες: την πύλη εισόδου, την πύλη διαχωρισμού και την πύλη εξόδου. Αυτές οι πύλες, μαζί με το κελί μνήμης, ρυθμίζουν τη ροή πληροφοριών, επιτρέποντας στα LSTM να θυμούνται ή να ξεχνούν επιλεκτικά πληροφορίες σε διάφορα χρονικά βήματα. Η πύλη εισόδου καθορίζει πόση από τις νέες πληροφορίες εισόδου πρέπει να προστεθεί στο κελί μνήμης. Η πύλη διαχωρισμού αποφασίζει τον όγκο των πληροφοριών από την προηγούμενη κατάσταση της κυψέλης

μνήμης που θα διατηρηθούν ή θα απορριφθούν. Τέλος η πύλη εξόδου ελέγχει τη συμβολή του κελιού μνήμης στην κρυφή κατάσταση, η οποία στη συνέχεια χρησιμοποιείται για την έξοδο του δικτύου. Οι μηχανισμοί πύλης στα LSTM τους επιτρέπουν να διατηρούν πιο αποτελεσματικά μακροπρόθεσμες εξαρτήσεις, καθιστώντας τους κατάλληλους για εργασίες που περιλαμβάνουν ακολουθίες με ποικίλα μήκη και πολύπλοκες χρονικές σχέσεις.



Εικόνα 13 Διάγραμμα Συνάρτησης Μεταφοράς Βασικής Μονάδας LSTM.
Πηγή: (Olah, 2015)

2.4 Graph Neural Networks - Νευρωνικά Δίκτυα Γράφων

Σε αυτό το κεφάλαιο, εξερευνούμε το πεδίο των νευρωνικών δικτύων γράφων. Ξεκινάμε με μια εισαγωγική επισκόπηση των θεμελιωδών εννοιών, και στη συνέχεια προχωράμε στη συζήτηση των βασικών αρχιτεκτονικών και εφαρμογών τους. Το κεφάλαιο καλύπτει πρακτικές και αξιοσημείωτες χρήσεις των Νευρωνικών Δικτύων Γράφων, τις βασικές αρχιτεκτονικές τους, καθώς και τους ρόλους τους στην ταξινόμηση κόμβων και την πρόβλεψη συνδέσμου.

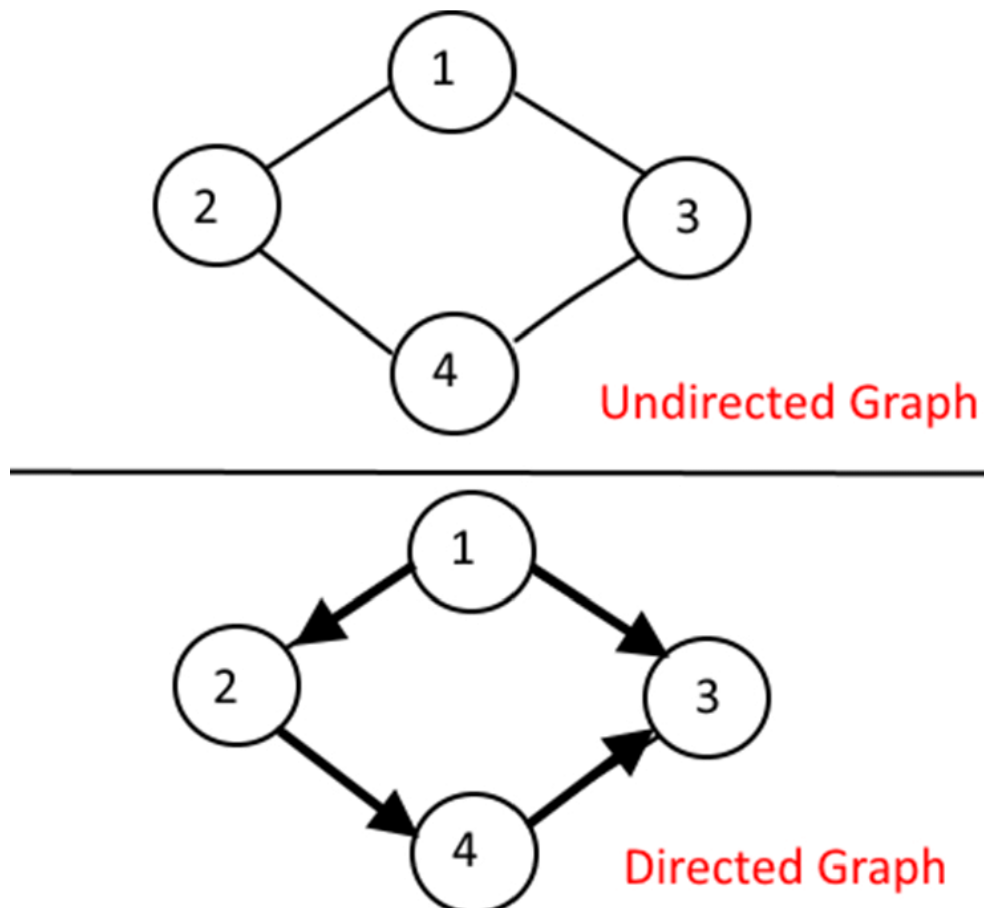
2.4.1 Εισαγωγή στους γράφους

Γενικότερα, ένας γράφος αναπαρίσταται ως $G = (V, E)$, όπου V είναι το σύνολο των κόμβων και E είναι το σύνολο των ακμών. Έστω $v_i \in V$ ένας κόμβος και $e_{ij} =$

$(v_i, v_j) \in E$ μια ακμή που δείχνει από το v_j στο v_i . Η γειτονιά ενός κόμβου v συμβολίζεται ως $N(v) = \{u \in V \mid (v, u) \in E\}$ (Battaglia et al., 2016).

Γενικά, οι γράφοι μπορούν να κατηγοριοποιηθούν ως εξής (Battaglia et al., 2016):

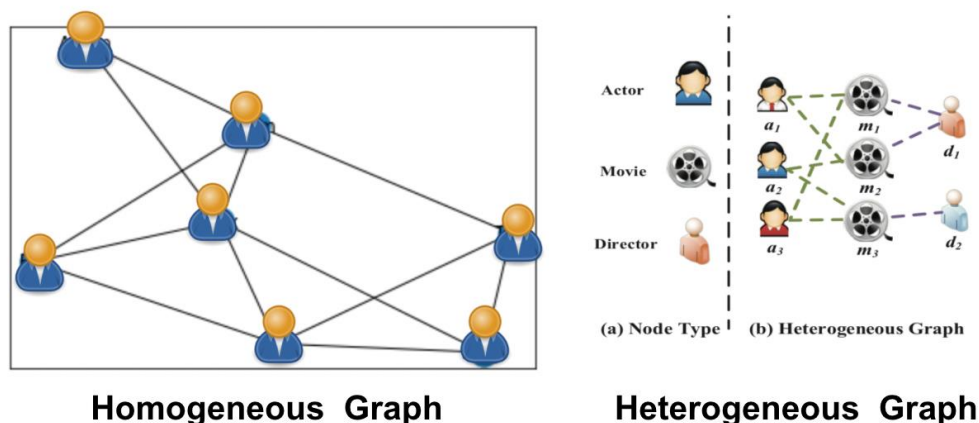
- Κατευθυνόμενος/μη κατευθυνόμενος γράφος: Ένας κατευθυνόμενος γράφος είναι ένας γράφος με όλες τις ακμές να κατευθύνονται από τον ένα κόμβο στον άλλο. Ένας μη κατευθυνόμενος γράφος θεωρείται ως ειδική περίπτωση κατευθυνόμενων γράφων όπου υπάρχει ένα ζεύγος ακμών με αντίστροφες κατευθύνσεις αν δύο κόμβοι συνδέονται.



Εικόνα 14 Μη κατευθυνόμενος και κατευθυνόμενος γράφος αντίστοιχα.

Πηγή: (García Cabello, 2019)

- Ομοιογενής/ετερογενής γράφος: Ο ομοιογενής γράφος αποτελείται από έναν τύπο κόμβων και ακμών, ενώ ο ετερογενής γράφος έχει πολλαπλούς τύπους κόμβων ή ακμών. Στην παρακάτω φωτογραφία ο ετερογενής γράφος (δεξιά) απαρτίζεται από τρία διαφορετικά είδη κόμβων, "actor", "movie" και "director" αντίστοιχα.

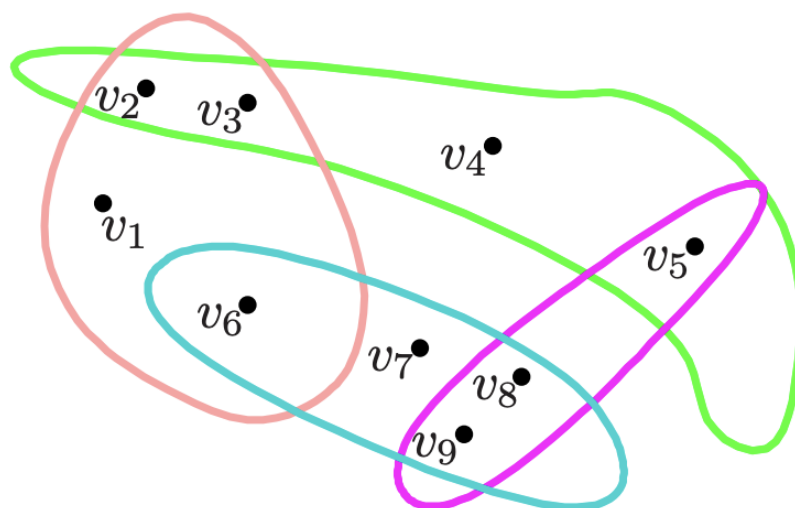


Εικόνα 15 Ομοιογενής και ετερογενής γράφος αντίστοιχα.

Πηγή: <https://medium.com/mlearning-ai/graph-basics-and-application-4de971ce0a0e>

- **Υπεργράφος:** Ένας υπεργράφος είναι μια γενίκευση ενός γράφου στο οποίο μια ακμή μπορεί να συνδέει οποιονδήποτε αριθμό κορυφών. Αποτελεί μια φυσική επέκταση ενός γραφήματος που προκύπτει με την αφαίρεση του περιορισμού στην πληθικότητα μιας ακμής: οποιοδήποτε μη κενό υποσύνολο του V μπορεί να είναι στοιχείο (υπερακμή) του συνόλου ακμών E . Έτσι, αποτυπώνει τις συμπεριφορές ομάδων και τις σχέσεις υψηλότερων διαστάσεων σε πολύπλοκα δίκτυα (Gao et al., 2012). Στην παρακάτω φωτογραφία ο υπεργράφος έχει τέσσερις υπερακμές:

$$(v_1, v_2, v_3, v_6), (v_2, v_3, v_4, v_5), (v_6, v_7, v_8, v_9), \text{ και } (v_5, v_8, v_9)$$

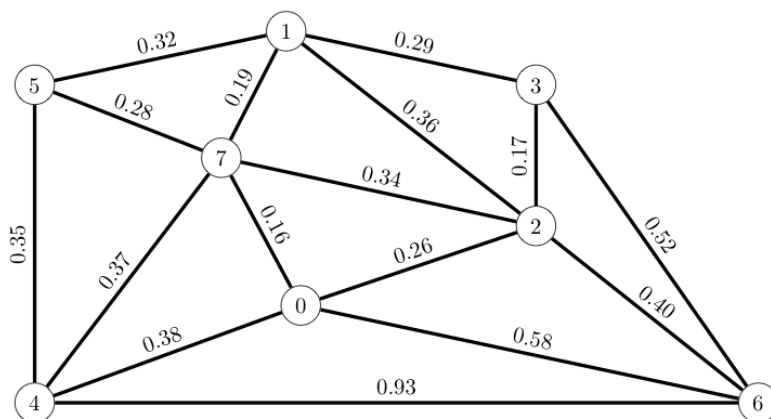


Εικόνα 16 Παράδειγμα υπεργράφου με τέσσερις υπερακμές.

Πηγή: (Gao et al., 2012)

- **Σταθμισμένος γράφος:** Ο σταθμισμένος γράφος είναι ένας τύπος γράφου στον οποίο δίνεται ένα αριθμητικό βάρος για κάθε ακμή. Έτσι, ένας σταθμισμένος

γράφος είναι ένας ειδικός τύπος γράφου με ετικέτες, όπου οι ετικέτες είναι αριθμοί (Das & Soylu, 2023). Στην παρακάτω φωτογραφία φαίνεται ένας σταθμισμένος γράφος όπου τα βάρη σημειώνονται στην κάθε ακμή.



Εικόνα 17 Σταθμισμένος γράφος.

Πηγή: <https://mathcenter.oxford.emory.edu/site/cs171/directedAndEdgeWeightedGraphs/>

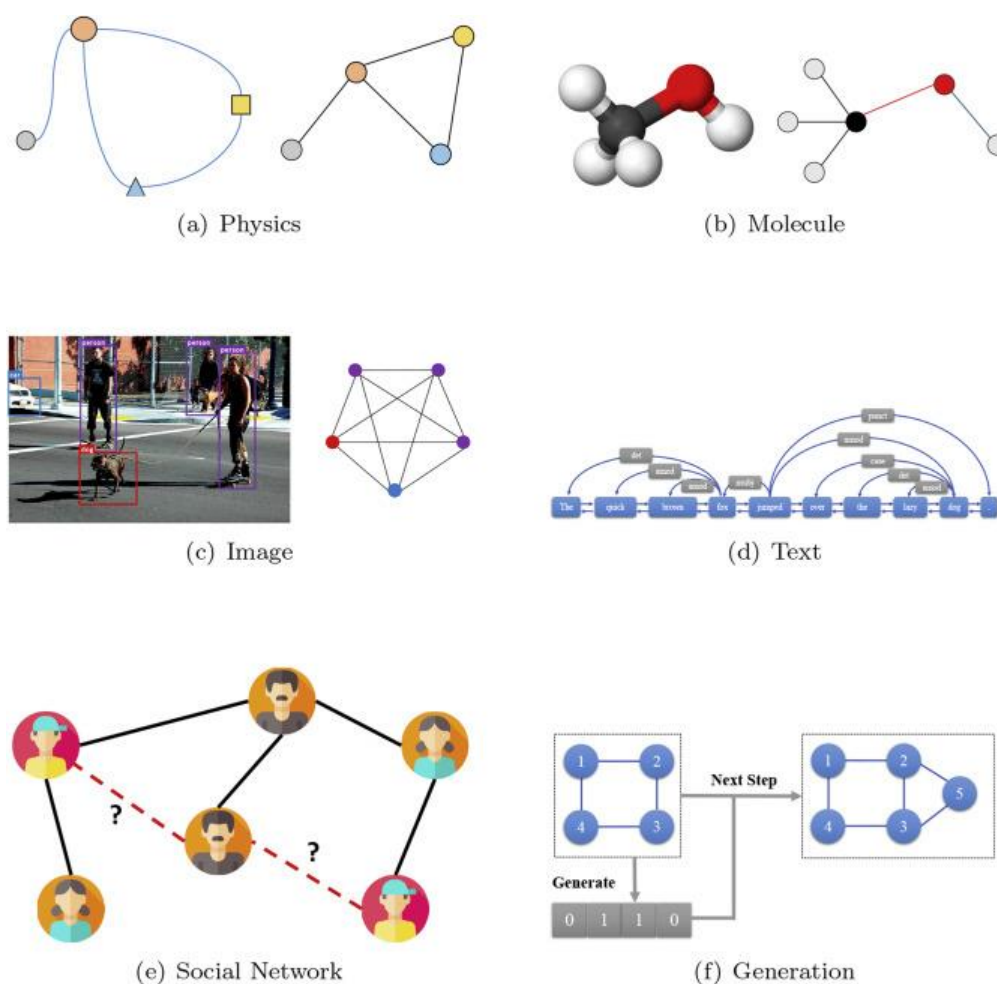
2.4.2 Εισαγωγή στα νευρωνικά δίκτυα γράφων

Τα νευρωνικά δίκτυα γράφων (GNN) είναι μια κατηγορία μοντέλων βαθιάς μάθησης που έχουν σχεδιαστεί για να χειρίζονται δεδομένα δομημένων γράφων, τα οποία αποτελούνται από κόμβους και ακμές που αντιπροσωπεύουν οντότητες και τις σχέσεις τους, αντίστοιχα. Ένα από τα πρώτα παραδείγματα νευρωνικών δικτύων γράφων παρουσιάζεται στο άρθρο με τίτλο "Neural Network for Graphs: A Contextual Hierarchical Approach" των Alex J. Smola και Risi Kondor, που δημοσιεύτηκε στο Neural Information Processing Systems (NIPS) το 2003. Αυτή η εργασία εισήγαγε μια μέθοδο για εκμάθηση αναπαραστάσεων γραφημάτων χρησιμοποιώντας ένα πλαίσιο νευρωνικών δικτύων. Επικεντρώθηκε στην κατασκευή μιας ιεραρχίας αναπαραστάσεων για τους κόμβους και τις γειτονιές τους σε ένα γράφο, επιτρέποντας την ενσωμάτωση τοπικών και καθολικών πληροφοριών της δομής ενός γράφου στη διαδικασία εκμάθησης.

Το "The Graph Neural Network Model" των F. Scarselli, M. Gori, A.C. Tsoi, M. Hagenbuchner, and G. Monfardini, που δημοσιεύτηκε στο IEEE Transactions on Neural Networks, Τόμος 20(1), σελ. 61-80, το 2009, είναι ακόμη ένα από τα πρώιμα έργα με επιρροή στα νευρωνικά δίκτυα γράφων (GNN). Εισήγαγε ένα νέο μοντέλο για νευρωνικά δίκτυα που λειτουργούν απευθείας σε γράφους. Οι συγγραφείς

παρουσίασαν μια μέθοδο για την αναδρομική ενημέρωση των αναπαραστάσεων κόμβων σε μια δομή γράφου, επιτρέποντας την ενσωμάτωση πληροφοριών που αφορούν τη δομή του γράφου στη μαθησιακή διαδικασία.

Γενικά, τα GNN λειτουργούν μαθαίνοντας αναπαραστάσεις κόμβων μέσω μηχανισμών διέλευσης μηνυμάτων, όπου κάθε κόμβος συγκεντρώνει πληροφορίες από τους γειτονικούς κόμβους του επαναληπτικά. Αυτή η διαδικασία επιτρέπει στα GNN να συλλαμβάνουν τόσο τοπικές όσο και καθολικές δομές γράφων, επιτρέποντας την εκμάθηση πολύπλοκων μοτίβων σε δεδομένα δομημένων γράφων.



Εικόνα 18 Πεδία εφαρμογής νευρωνικών δικτύων γράφων
Πηγή: (Zhou et al., 2020)

Δεδομένων των βασικών πληροφοριών που αφορούν τους γράφους, η βασική ιδέα του GNN είναι να συγκεντρώσει επαναληπτικά πληροφορίες χαρακτηριστικών από τους γείτονες και να ενσωματώσει τις συγκεντρωμένες πληροφορίες με την τρέχουσα αναπαράσταση του κεντρικού κόμβου κατά τη διαδικασία διάδοσης. Από την άποψη

της αρχιτεκτονικής του δικτύου, το GNN στοιβάζει πολλαπλά επίπεδα διάδοσης, τα οποία αποτελούνται από τις συναρτήσεις συνάθροισης και ενημέρωσης. Η διατύπωση της διάδοσης είναι (Wu et al., 2016):

$$\text{Aggregation: } n_v^{(l)} = \text{Aggregator}_l(\{h_u^l, \forall u \in \mathcal{N}_v\})$$

$$\text{Update: } h_v^{(l+1)} = \text{Updater}_l(h_v^{(l)}, n_v^{(l)})$$

όπου $h_u^{(l)}$ δηλώνει την αναπαράσταση του κόμβου u στο επίπεδο l και οι Aggregator_l και Updater_l αναπαριστούν τη λειτουργία της λειτουργίας συνάθροισης και της λειτουργίας ενημέρωσης στο επίπεδο l , αντίστοιχα. Το βήμα της συνάθροισης γίνεται με διάφορες μεθόδους όπως για παράδειγμα mean-pooling, max-pooling και pooling με βάση μηχανισμό προσοχής (Kumar & Rajput, 2023). Ενδεικτικά το mean-pooling λαμβάνει το μέσο όρο των χαρακτηριστικών των γειτονικών κόμβων ενώ ο μηχανισμός προσοχής σταθμίζει τη σημασία των διαφορετικών γειτόνων κατά την άθροιση των χαρακτηριστικών τους. Αυτές οι συναρτήσεις συνάθροισης χρησιμοποιούνται σε κάθε επίπεδο ενός δικτύου για την ενημέρωση των χαρακτηριστικών κάθε κόμβου με βάση τους γείτονές του, λειτουργούν δηλαδή σε επίπεδο κόμβου (Kumar & Rajput, 2023). Στο βήμα ενημέρωσης, η αναπαράσταση του κεντρικού κόμβου και η συγκεντρωτική γειτονιά θα ενσωματωθούν στην ενημερωμένη αναπαράσταση του κεντρικού κόμβου. Προκειμένου να προσαρμοστούν σε διαφορετικά σενάρια, προτείνονται διάφορες στρατηγικές για την καλύτερη ενσωμάτωση των δύο αναπαραστάσεων, όπως ο μηχανισμός GRU, η συνένωση με μη γραμμικό μετασχηματισμό και η λειτουργία αθροίσματος.

2.4.3 Πρακτικές και αξιοσημείωτες εφαρμογές των GNN

Τα GNN έχουν κερδίσει δημοτικότητα τα τελευταία χρόνια λόγω της ικανότητάς τους να αξιοποιούν τις πλούσιες σχεσιακές πληροφορίες που υπάρχουν στους γράφους, καθιστώντας τα κατάλληλα για εργασίες όπως η ταξινόμηση κόμβων, η πρόβλεψη ακμών και η ταξινόμηση γράφων (Scarselli et al., 2009; Wu et al., 2020). Πρόσφατα, συστήματα που βασίζονται σε παραλλαγές του GNN έχουν επιδείξει πρωτοποριακές επιδόσεις σε πολλές εργασίες που σχετίζονται με δεδομένα γράφων, όπως τα φυσικά

συστήματα, τα συστήματα συστάσεων, η δομή πρωτεϊνών και ο γράφος γνώσης κ.ά. . Παρακάτω θα παρουσιάσουμε μερικά αξιοσημείωτα επιστημονικά άρθρα

Οι Jumper et al., (2021) παρουσίασαν το πρωτοποριακό σύστημα AlphaFold, το οποίο χρησιμοποιεί τεχνικές βαθιάς μάθησης, συμπεριλαμβανομένων των Δικτύων Νευρωνικών Γράφων, για την πρόβλεψη δομών πρωτεϊνών με εκπληκτική ακρίβεια. Η πρόβλεψη της δομής των πρωτεϊνών είναι ένα θεμελιώδες πρόβλημα στη βιολογία με ευρείς προβληματισμούς, συμπεριλαμβανομένης της ανακάλυψης φαρμάκων, της κατανόησης των νόσων και της βιοτεχνολογίας. Η καινοτόμα χρήση των GNNs και των μηχανισμών προσοχής από το AlphaFold σημείωσε μια σημαντική πρόοδο στην πρόβλεψη δομών πρωτεϊνών, υπερβαίνοντας άλλες μεθόδους και συμβάλλοντας σημαντικά στον τομέα της δομικής βιολογίας.

Το ερευνητικό άρθρο με τίτλο "Pixie: Ένα Σύστημα για τη Σύσταση 3+ Δισεκατομμυρίων Αντικειμένων σε Πάνω από 200 Εκατομμύρια Χρήστες σε Πραγματικό Χρόνο" αποκαλύπτει μια πρωτοποριακή προσέγγιση στον τομέα των συστημάτων πρότασης (Eksombatchai et al., 2017). Το σύστημα Pixie εφαρμόζεται εντός της πλατφόρμας Pinterest και το άρθρο υπογραμμίζει την ικανότητά του να επιλέγει και να προτείνει αντικείμενα γρήγορα από μια εκτεταμένη συλλογή που υπερβαίνει τις 3 δισεκατομμύρια καταχωρίσεις, εξυπηρετώντας μια τεράστια βάση χρηστών που υπερβαίνει τα 200 εκατομμύρια. Αυτό το τεχνολογικό επίτευγμα αντιμετωπίζει τις περίπλοκες προκλήσεις που τίθενται από την ανάγκη για προτάσεις σε πραγματικό χρόνο και εξατομικευμένες σε μια εκτεταμένη ψηφιακή πλατφόρμα, σημειώνοντας ένα σημαντικό βήμα στην εξέλιξη των συστημάτων πρότασης.

Άλλη μια πλατφόρμα που χρησιμοποιεί νευρωνικά δίκτυα για τις συστάσεις τους είναι η Uber (Jain, Liu, Sarda, & Molino, 2019). Συγκεκριμένα η πλατφόρμα του Uber Eats αξιοποιεί τη δύναμη των Γραφικών Νευρωνικών Δικτύων (GNNs) για τη βελτίωση του συστήματος σύστασης πιάτων. Αναπαριστώντας τους χρήστες και τα πιάτα ως κόμβους σε ένα γράφο, η πλατφόρμα απαθητεύει τις περίπλοκες σχέσεις μεταξύ των χρηστών και των συνηθειών παραγγελίας τους. Οι συνδέσεις, ή ακμές, μεταξύ αυτών των κόμβων έχουν βάρος βάσει της συχνότητας των παραγγελιών, επιτρέποντας στο σύστημα να κατανοεί τις προτιμήσεις των χρηστών με πιο λεπτομερή τρόπο. Με τα GNNs, το Uber Eats μπορεί να συγκεντρώνει πληροφορίες από γειτονικούς κόμβους,

κάτι που σημαίνει ότι κατανοεί τις προτιμήσεις ενός χρήστη βάσει των παρελθόντων παραγγελιών του και των πρότυπων παραγγελίας παρόμοιων χρηστών. Αυτή η δομή επιτρέπει στην πλατφόρμα να προσαρμόζεται δυναμικά στις αλλαγές των προτιμήσεων των χρηστών και να κάνει συστάσεις που είναι σχετικές και προσαρμοσμένες. Η ενσωμάτωση των GNNs οδήγησε σε σημαντική βελτίωση στην ακρίβεια των συστάσεων της πλατφόρμας, όπως φαίνεται από τη σημαντική αύξηση των ποσοστών Click-Through κατά τις δοκιμές τους, υπογραμμίζοντας τα πρακτικά πλεονεκτήματα της μηχανικής μάθησης βασισμένης σε γράφους στη βελτίωση της εμπειρίας των χρηστών.

Τα Δίκτυα Νευρωνικών Γραφημάτων (GNNs) έχουν χρησιμοποιηθεί επίσης στον τομέα των φυσικών συστημάτων και προσομοιώσεων. Ένα πρωτοποριακό άρθρο είναι το "Interaction Networks for Learning about Objects, Relations and Physics" των Battaglia et. al. (2016). Παρουσιάζει ένα μοντέλο που χρησιμοποιεί δίκτυα γραφημάτων για να ερμηνεύσει τη συμπεριφορά των φυσικών συστημάτων. Το σύστημα, γνωστό ως Δίκτυο Αλληλεπίδρασης (IN), είναι ένας τύπος νευρωνικού δικτύου γραφήματος που σχεδιάστηκε για να ερμηνεύσει τον φυσικό κόσμο. Λαμβάνει ως είσοδο ένα γράφημα στο οποίο οι κόμβοι αντιπροσωπεύουν αντικείμενα και οι ακμές αντιπροσωπεύουν αλληλεπιδράσεις μεταξύ τους. Το IN χρησιμοποιεί αυτό το γράφο για να προβλέψει φυσικές ιδιότητες του συστήματος και τις αλλαγές που θα υποστεί με την πάροδο του χρόνου.

Ομοίως, το "Learning to Simulate Complex Physics with Graph Networks" παρουσιάζει το πλαίσιο Graph Network-based Simulators (GNS), το οποίο έχει τη δυνατότητα να προσομοιώνει τη δυναμική μιας διευρυμένης γκάμας υλικών (υγρά, στερεά και πλαστικά υλικά) που αλληλεπιδρούν μεταξύ τους, ακόμη και με μεγάλες ποσότητες σωματιδίων κατά τη διάρκεια εκτεταμένων χρονικών περιόδων (Sanchez-Gonzalez et al., 2020).

Τα GNNs έχουν επίσης χρησιμοποιηθεί για ανάλυση τρισδιάστατων σχημάτων, εκτίμηση ανθρώπινης στάσης κ.ά. . Οι Qi et al. (2017) περιγράφουν πώς τα GNNs χρησιμοποιήθηκαν για αναγνώριση τρισδιάστατων αντικειμένων λαμβάνοντας ως είσοδο αδόμητα δεδομένα σημείων σε τρεις διαστάσεις.

Επιπρόσθετα, το "Fake News Detection on Social Media using Geometric Deep Learning" από τους Monti, F., Frasca, F., Eynard, D., Mannion, D., & Bronstein, M. M. (2019) είναι ένα σημαντικό άρθρο στον τομέα της ανίχνευσης ψευδών ειδήσεων. Το άρθρο προτείνει το GeoD, μια προσέγγιση βασισμένη στο Geometric Deep Learning για την ανίχνευση ψευδών ειδήσεων στα κοινωνικά δίκτυα. Οι συγγραφείς υποστηρίζουν ότι οι ψευδείς ειδήσεις και οι πραγματικές ειδήσεις εμφανίζουν διακριτά μοτίβα στα γραφήματα διάδοσής τους: για παράδειγμα, οι ψευδείς ειδήσεις τείνουν να διαδίδονται πιο γρήγορα και ευρέως, και με πιο ιεραρχημένο τρόπο. Το προτεινόμενο μοντέλο GeoD λειτουργεί εφαρμόζοντας δίκτυα γραφηματικής συνέλιξης (GCNs) σε αυτά τα γραφήματα διάδοσης, καταγράφοντας τη διάδοση των ειδήσεων στον υψηλής διάστασης 'χώρο διάδοσης'. Στη συνέχεια, το μοντέλο χρησιμοποιεί αυτές τις πληροφορίες για να ταξινομήσει τις ειδήσεις ως ψευδείς ή πραγματικές. Είναι αξιοσημείωτο ότι το GeoD ήταν ιδιαίτερα αποτελεσματικό στην ανίχνευση ψευδών ειδήσεων στην αρχή της διάδοσής τους, ακόμη και όταν ήταν ορατό μόνο ένα μικρό τμήμα του γραφήματος διάδοσης.

Η ανίχνευση εισβολών στο δίκτυο, η ανίχνευση κακόβουλων URL και η ανίχνευση κακόβουλου λογισμικού είναι ένας τομέας στον οποίο έχουν εφαρμοστεί επίσης τα GNNs. Οι Wang, Xue, & Song (2022) και οι Liu et al. (2020) πρότειναν μοντέλα για την ανίχνευση ανωμαλιών και πιθανών εισβολών στα δεδομένα του δικτύου και ανίχνευση δημιουργίας κακόβουλων λογαριασμών αντίστοιχα.

Αυτά είναι μόνο λίγα παραδείγματα του πώς τα GNNs έχουν εφαρμοστεί πρακτικά σε διάφορους τομείς, με κάθε εφαρμογή να εκμεταλλεύεται τη μοναδική ικανότητα των GNNs να απαθανατίζουν τη δομή και τα χαρακτηριστικά των περίπλοκων γράφων.

2.4.4 Βασικές αρχιτεκτονικές GNNs

Παρακάτω θα συνοψίσουμε πέντε κοινές αρχιτεκτονικές νευρωνικών δικτύων γράφων και θα αναφέρουμε σε λεπτομέρεια τις συναρτήσεις συνάθροισης και ενημέρωσής τους. Η πρώτη αρχιτεκτονική είναι τα Graph Convolutional Networks (GCN) (Kipf & Welling, 2016). Προσεγγίζει την λαπλασιανή ιδιοδιάσπαση πρώτης τάξης του γράφου για να συγκεντρώσει επαναληπτικά πληροφορίες από τους γείτονες. Συγκεκριμένα, ενημερώνει την ενσωμάτωση με:

$$\text{Aggregation: } n_v^{(l)} = \sum_{j \in \mathcal{N}_v} d_{vv}^{-\frac{1}{2}} a_{vj} d_{jj}^{-\frac{1}{2}} h_j^{(l)}$$

$$\text{Update: } h_v^{(l+1)} = \delta(W^{(l)} n_v^{(l)})$$

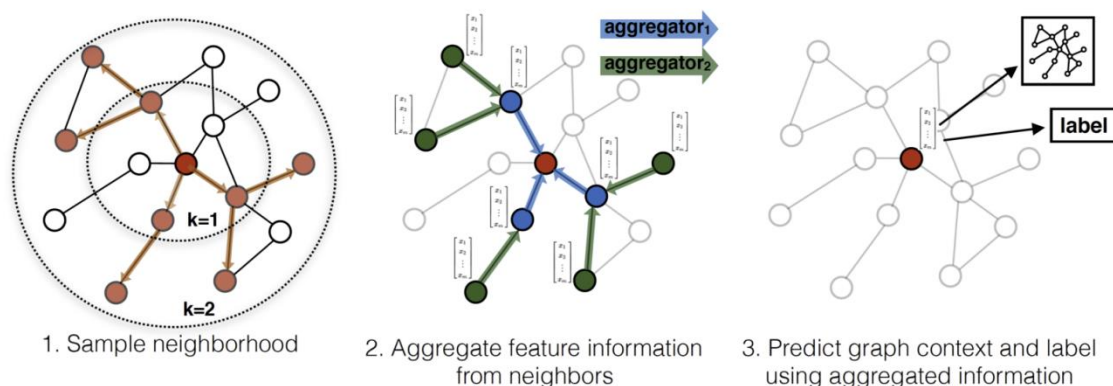
όπου $\delta(\cdot)$ είναι η μη γραμμική συνάρτηση ενεργοποίησης, όπως η ReLU, $W^{(l)}$ είναι ο μαθησιακός πίνακας μετασχηματισμού για το επίπεδο l , $\widetilde{a}_{vj}$ είναι το βάρος γειτνίασης ($\widetilde{a}_{vv} = 1$) και $d_{jj} = \sum_k \widetilde{a}_{jk}$.

Το GraphSAGE (Hamilton, Ying & Leskovec, 2017) δειγματοληπτεί ένα σταθερό μέγεθος γειτονιάς για κάθε κόμβο, προτείνει τον αθροιστή mean/sum/maxpooling και υιοθετεί τη λειτουργία concatenation για την ενημέρωση,

$$\text{Aggregation: } n_v^{(l)} = \text{Aggregator}_l(\{h_u^l, \forall u \in \mathcal{N}_v\})$$

$$\text{Update: } h_v^{(l+1)} = \delta(W^{(l)} \cdot [h_v^{(l)} \oplus n_v^{(l)}])$$

όπου ο Aggregator_l δηλώνει τη συνάρτηση συνάθροισης στο l επίπεδο, $\delta(\cdot)$ είναι η μη γραμμική συνάρτηση ενεργοποίησης και $W^{(l)}$ είναι ο μαθησιακός πίνακας μετασχηματισμού.



Εικόνα 19 Απεικόνιση της μεθόδου δειγματοληψίας και συγκέντρωσης του GraphSAGE.
Πηγή: (Hamilton et al., 2018)

Το Graph Attention Networks (GAT) (Veličković et al., 2017) υποθέτει ότι η επιρροή των γειτόνων δεν είναι ούτε ταυτόσημη ούτε προκαθορισμένη από τη δομή του

γράφου, επομένως διαφοροποιεί τις συνεισφορές των γειτόνων αξιοποιώντας τον μηχανισμό προσοχής και ενημερώνει το διάνυσμα κάθε κόμβου με την προσοχή στους γείτονές του,

$$\text{Aggregation: } n_v^{(l)} = \sum_{j \in \mathcal{N}_v} \alpha_{vj} h_j^{(l)}, \alpha_{vj} = \frac{\exp\left(\text{Att}\left(h_v^{(l)}, h_j^{(l)}\right)\right)}{\sum_{k \in \mathcal{N}_v} \exp\left(\text{Att}\left(h_v^{(l)}, h_k^{(l)}\right)\right)}$$

$$\text{Update: } h_v^{(l+1)} = \delta\left(W^{(l)} n_v^{(l)}\right)$$

όπου $\text{Att}(\cdot)$ είναι μια συνάρτηση προσοχής (π.χ., η συνάρτηση $\text{LeakyReLU} = a^T[W^{(l)} h_u^{(l)} \oplus W^{(l)} h_j^{(l)}]$)

Το Gated Graph Sequence Neural Networks (GGNN) (Li et al., 2016) υιοθετεί μια Gated Recurrent Unit (GRU) στο βήμα ενημέρωσης,

$$\text{Aggregation: } n_v^{(l)} = \frac{1}{|\mathcal{N}_v|} \sum_{j \in \mathcal{N}_v} h_j^{(l)}$$

$$\text{Update: } h_v^{(l+1)} = \text{GRU}\left(h_v^{(l)}, n_v^{(l)}\right)$$

Το GGNN εκτελεί την επαναλαμβανόμενη συνάρτηση αρκετές φορές σε όλους τους κόμβους, γεγονός που ενδέχεται να αντιμετωπίσει το πρόβλημα της κλιμάκωσης όταν εφαρμόζεται σε μεγάλους γράφους.

Το Hypergraph Neural Networks (HGNN) (Feng et al., 2019) είναι ένα τυπικό νευρωνικό δίκτυο υπεργράφων, το οποίο κωδικοποιεί τη συσχέτιση δεδομένων υψηλής τάξης σε μια δομή υπεργράφων. Το επίπεδο συνελκτικού υπεργράφου έχει την ακόλουθη μαθηματική διατύπωση:

$$\text{Aggregation: } N^{(l)} = D_v^{-\frac{1}{2}} E W^0 \widetilde{D}_e^{-1} E^T D_v^{-\frac{1}{2}} H^{(l)}$$

$$\text{Update: } H^{(l+1)} = \delta\left(W^{(l)} N^{(l)}\right)$$

όπου $\delta(\cdot)$ είναι η μη γραμμική συνάρτηση ενεργοποίησης, όπως η ReLU, $W^{(l)}$ είναι ο μαθησιακός πίνακας μετασχηματισμού για το επίπεδο l , E είναι ο πίνακας γειννίας του υπεργράφου και D_e και D_v υποδηλώνουν τους διαγώνιους πίνακες των βαθμών ακμής και των βαθμών κορυφής, αντίστοιχα.

2.4.5 Νευρωνικά Δίκτυα Γράφων για ταξινόμηση κόμβων

Τα νευρωνικά δίκτυα γράφων (GNN) λειτουργούν σε δεδομένα δομημένα ως γράφους και είναι ιδιαίτερα χρήσιμα για εργασίες όπως η ταξινόμηση κόμβων όπου θέλουμε να προβλέψουμε την κλάση ή τις ιδιότητες των κόμβων σε ένα γράφο. Μια τέτοια διαδικασία ξεκινάει από την αναπαράσταση γράφου εισόδου. Σύμφωνα με τον J. Tang (2022) η κύρια ιδέα των νευρωνικών δικτύων γράφων είναι να ενημερώνουν επαναληπτικά τις αναπαραστάσεις των κόμβων, συνδυάζοντας τις αναπαραστάσεις των γειτονικών τους κόμβων και τις δικές τους αναπαραστάσεις. Σε κάθε επίπεδο του GNN, κάθε κόμβος ανταλλάσσει πληροφορίες με τους γειτονικούς κόμβους του και αυτές οι πληροφορίες χρησιμοποιούνται για την ενημέρωση της αναπαράστασης του ίδιου του κόμβου. Αυτή είναι συνήθως μια απλή λειτουργία, ακολουθούμενη από έναν μη γραμμικό μετασχηματισμό. Ο στόχος είναι να κωδικοποιηθούν όχι μόνο τα χαρακτηριστικά του κόμβου, αλλά και η δομή του γράφου γύρω από τον κόμβο.

Μετά τη συνάθροιση, η αναπαράσταση του κόμβου ενημερώνεται ή μετασχηματίζεται συνήθως μέσω μιας συνάρτησης όπως ένα επίπεδο νευρωνικού δικτύου. Αυτό το επίπεδο λαμβάνει τις συγκεντρωτικές πληροφορίες και την τρέχουσα αναπαράσταση του κόμβου και δημιουργεί μια νέα αναπαράσταση. Η διαδικασία επαναλαμβάνεται σε πολλά επίπεδα, κάθε φορά που συγκεντρώνει πληροφορίες από μια μεγαλύτερη γειτονιά. Μετά τα επίπεδα L , η αναπαράσταση κάθε κόμβου είναι συνάρτηση των χαρακτηριστικών όλων των κόμβων εντός του άλματος L (L -hop neighborhood) στο γράφο. Μετά τον επιθυμητό αριθμό επιπέδων, εφαρμόζεται μια συνάρτηση ανάγνωσης (readout function) για τη δημιουργία των τελικών ενσωματώσεων κόμβων. Η συνάρτηση ανάγνωσης πρέπει να είναι μεταθετική ως προς τη διάταξη (δηλαδή, η σειρά των κόμβων δεν πρέπει να έχει σημασία) και συχνά περιλαμβάνει τη λήψη του αθροίσματος, του μέσου όρου ή του μέγιστου των αναπαραστάσεων του κόμβου. Οι τελικές ενσωματώσεις κόμβων χρησιμοποιούνται στη συνέχεια για ταξινόμηση κόμβων. Αυτό γίνεται συνήθως με την εφαρμογή ενός τελικού (συχνά πλήρως

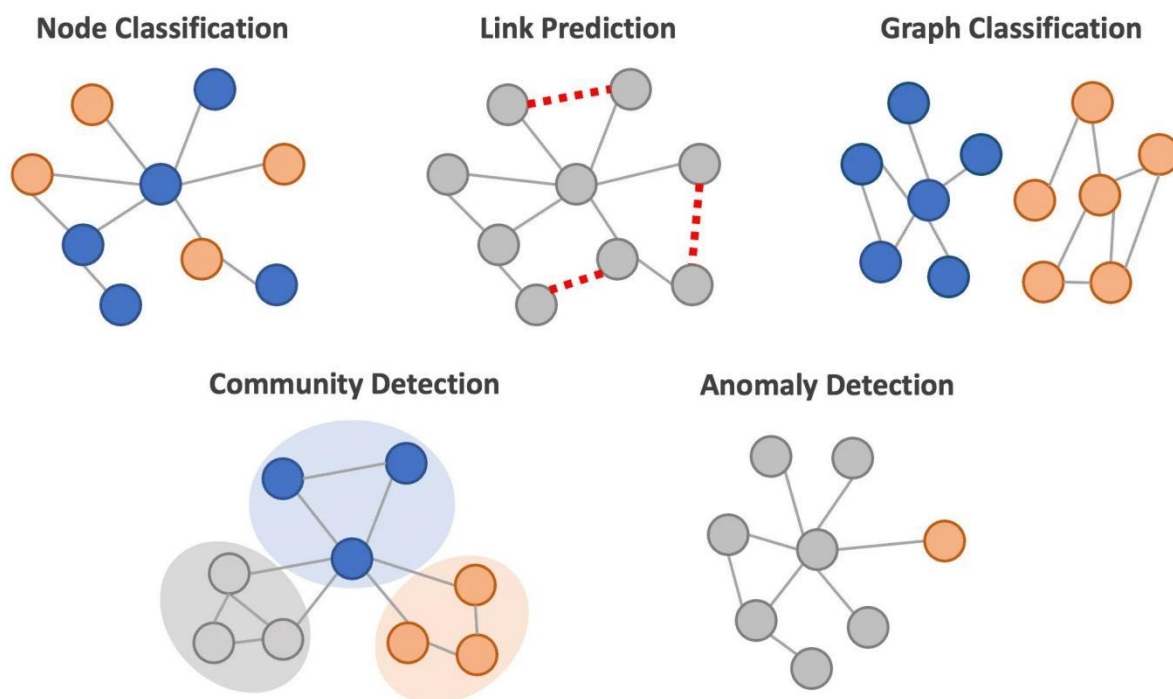
συνδεδεμένου) επιπέδου που χαρτογραφείται από τον χώρο ενσωμάτωσης στις κλάσεις εξόδου, ακολουθούμενο από μια συνάρτηση softmax για να ληφθούν οι πιθανότητες των κλάσεων. Στη συνέχεια, το μοντέλο εκπαιδεύεται βελτιστοποιώντας μια συνάρτηση απώλειας που συγκρίνει αυτές τις προβλεπόμενες πιθανότητες με τις πραγματικές κλάσεις, χρησιμοποιώντας μια μέθοδο όπως η κατάβαση κλίσης.

Οι Kipf και Welling (2017) εισήγαγαν τα Graph Convolutional Networks (GCN) και απέδειξαν τη χρησιμότητά τους για ημι-εποπτευόμενη ταξινόμηση στα κοινωνικά δίκτυα. Η προσέγγισή τους αξιοποιεί τη δομή του γράφου και τα χαρακτηριστικά που σχετίζονται με κάθε κόμβο για την πρόβλεψη ετικετών κόμβων, όπως τον προσδιορισμό των ρόλων των χρηστών σε ένα κοινωνικό δίκτυο. Οι Velickovic et al. (2018) πρότειναν το Graph Attention Networks (GAT), έναν τύπο GNN που χρησιμοποιεί μηχανισμούς προσοχής για να σταθμίσει την επιρροή των κόμβων κατά τη συγκέντρωση πληροφοριών γειτόνων. Οι ερευνητές εφάρμοσαν GAT για την ταξινόμηση κόμβων σε δίκτυα παραπομπών, όπου οι κόμβοι αντιπροσωπεύουν έγγραφα, οι ακμές αντιπροσωπεύουν αναφορές και ο στόχος ήταν να ταξινομήσουν τα έγγραφα σε μία από τις πολλές κατηγορίες με βάση το περιεχόμενό τους. Επίσης, οι Zitnik et al. (2018) χρησιμοποίησαν GCN για την πρόβλεψη παρενεργειών σε συνδυασμούς φαρμάκων. Ο γράφος αντιπροσωπεύει ένα δίκτυο φαρμάκων, όπου οι κόμβοι είναι τα φάρμακα και οι ακμές είναι οι γνωστές αλληλεπιδράσεις μεταξύ τους. Το μοντέλο GCN ήταν σε θέση να ταξινομήσει τους κόμβους (φάρμακα) ως προς την τάση τους να προκαλούν ορισμένες παρενέργειες όταν συνδυάζονται με άλλες.

2.4.6 Νευρωνικά δίκτυα γράφων για πρόβλεψη συνδέσμου

Η πρόβλεψη συνδέσμου (ύπαρξη ακμής) είναι μια θεμελιώδης εργασία στην ανάλυση γράφου που περιλαμβάνει την πρόβλεψη εάν ένας σύνδεσμος (μια ακμή) υπάρχει μεταξύ δύο κόμβων στο γράφο. Αυτή η διαδικασία είναι κρίσιμη σε μια ποικιλία εφαρμογών, όπως συστήματα συστάσεων, ανάλυση κοινωνικών δικτύων, ανάλυση βιολογικών δικτύων κ.α. (Zhang, M. 2022). Στο πλαίσιο των GNN, η πρόβλεψη συνδέσμου εκτελείται με εκμάθηση ενσωματώσεων κόμβων που μπορούν να αναπαραστήσουν αποτελεσματικά τη δομή και τις ιδιότητες του γράφου. Οι μαθημένες

ενσωματώσεις κόμβων χρησιμοποιούνται για να συμπεράνουμε εάν πρέπει να υπάρχει σύνδεση μεταξύ δύο κόμβων.



Εικόνα 20 Εφαρμογές Νευρωνικών Δικτύων Γράφων: Ταξινόμηση κόμβων, πρόβλεψη ακμής, ταξινόμηση γράφου, ανίχνευση κοινότητας, ανίχνευση ανωμαλίας.
Πηγή: (Masui, 2022)

Η διαδικασία ξεκινάει με την αναπαράσταση γράφου εισόδου. Παρόμοια με την εργασία ταξινόμησης κόμβων, η είσοδος στο GNN είναι ένας γράφος $G(V,E)$, όπου το V αντιπροσωπεύει το σύνολο των κόμβων και το E αντιπροσωπεύει το σύνολο των ακμών. Κάθε κόμβος μπορεί επίσης να συσχετιστεί με ένα διάνυσμα χαρακτηριστικών. Το βασικό συστατικό ενός GNN είναι η συγκέντρωση ή το βήμα μετάδοσης μηνυμάτων, όπου κάθε κόμβος συγκεντρώνει πληροφορίες από τους γειτονικούς κόμβους του για να ενημερώσει τη δική του αναπαράσταση. Αυτή η διαδικασία συνήθως γίνεται μέσω πολλαπλών επιπέδων, καθένα από τα οποία συγκεντρώνει πληροφορίες από μια μεγαλύτερη γειτονιά. Μετά τον επιθυμητό αριθμό επιπέδων, εφαρμόζεται μια συνάρτηση ανάγνωσης για τη δημιουργία των τελικών ενσωματώσεων κόμβων. Αυτές οι ενσωματώσεις θα πρέπει ιδανικά να αντιπροσωπεύουν τους κόμβους με τέτοιο τρόπο ώστε οι κόμβοι με μεγάλη πιθανότητα σχηματισμού ακμής να έχουν παρόμοιες ή σχετικές αναπαραστάσεις. Μόλις δημιουργηθούν οι ενσωματώσεις κόμβων, χρησιμοποιούνται για την πρόβλεψη συνδέσμων. Αυτή η διαδικασία συνήθως περιλαμβάνει τη δημιουργία μιας

βαθμολογίας για κάθε δυνητικό ζεύγος κόμβων για να αντιπροσωπεύει την πιθανότητα μιας ακμής. Υπάρχουν διάφοροι τρόποι υπολογισμού αυτής της βαθμολογίας. Μια κοινή προσέγγιση είναι ο υπολογισμός μιας βαθμολογίας ομοιότητας μεταξύ των ενσωματώσεων δύο κόμβων. Εναλλακτικά, μπορεί να εκπαιδευτεί ένας δυαδικός ταξινομητής που παίρνει τις ενσωματώσεις δύο κόμβων ως είσοδο και εξάγει μια πιθανότητα ύπαρξης μιας σύνδεσης μεταξύ τους.

Στη συνέχεια το μοντέλο εκπαιδεύεται βελτιστοποιώντας μια συνάρτηση απώλειας που ενθαρρύνει τις προβλεπόμενες βαθμολογίες να ταιριάζουν με την πραγματική ύπαρξη άκρων στο γράφο. Για παράδειγμα, τα ζεύγη κόμβων που έχουν μια άκρη μεταξύ τους στο πραγματικό γράφο θα πρέπει να έχουν υψηλή βαθμολογία, ενώ τα ζεύγη κόμβων που δεν έχουν ακμή θα πρέπει να έχουν χαμηλή βαθμολογία. Οι παράμετροι του GNN ενημερώνονται χρησιμοποιώντας μεθόδους βελτιστοποίησης που βασίζονται σε κλίση.

Μετά την εκπαίδευση, το μοντέλο μπορεί να χρησιμοποιηθεί για την πρόβλεψη συνδέσμων που λείπουν στο γράφο ή για την πρόβλεψη μελλοντικών συνδέσμων εάν ο γράφος αντιπροσωπεύει ένα δυναμικό σύστημα που εξελίσσεται με την πάροδο του χρόνου. Ορισμένες προσεγγίσεις εξετάζουν επίσης το πρόβλημα της πρόβλεψης συνδέσμων που περιλαμβάνει αόρατους κόμβους κατά τη διάρκεια της εκπαίδευσης. Αυτή η ρύθμιση απαιτεί συχνά πρόσθετες στρατηγικές, όπως τη χρήση χαρακτηριστικών κόμβων ή δομής υπογράφων για τη δημιουργία ενσωματώσεων κόμβων.

Οι Wang et al. (2019) παρουσιάζουν ένα μοντέλο σύστασης που βασίζεται σε γράφο, όπου οι κόμβοι αντιπροσωπεύουν χρήστες και στοιχεία και οι ακμές αντιπροσωπεύουν αλληλεπιδράσεις. Χρησιμοποιώντας τα νευρωνικά δίκτυα γράφων, μαθαίνουν ενσωματώσεις κόμβων για την πρόβλεψη συνδέσμων, προβλέποντας εάν ένας χρήστης θα αλληλεπιδράσει με ένα αντικείμενο στο μέλλον. Οι Zitnik και Leskovec (2017) που ασχολούνται με τον τομέα της βιοπληροφορικής, χρησιμοποιούν τα GNN για την πρόβλεψη των αλληλεπιδράσεων φαρμάκων-φαρμάκων. Οι κόμβοι αντιπροσωπεύουν φάρμακα και οι ακμές αντιπροσωπεύουν γνωστές αλληλεπιδράσεις μεταξύ τους. Οι συγγραφείς χρησιμοποίησαν GNN για να προβλέψουν την πιθανότητα μη παρατηρημένων αλληλεπιδράσεων, ένα κρίσιμο πρόβλημα στη φαρμακολογία. Οι Zhou et al. (2018) παρουσιάζουν μια ολοκληρωμένη ανασκόπηση των μεθόδων και

εφαρμογών GNN. Σε μία από τις επισημασμένες εφαρμογές, τα GNN χρησιμοποιούνται για την πρόβλεψη συνδέσμων στα κοινωνικά δίκτυα, προβλέποντας πιθανές μελλοντικές συνδέσεις. Η μελέτη των Schlichtkrull et al. (2018) παρουσιάζει μια εφαρμογή των GNN στο πλαίσιο δικτύων ακαδημαϊκής συνεργασίας. Οι κόμβοι στο γράφο αντιπροσωπεύουν τους συγγραφείς και μια ακμή μεταξύ δύο συγγραφέων υποδηλώνει ότι έχουν συνυπογράψει μια εργασία. Οι συγγραφείς χρησιμοποιούν μια παραλλαγή των GNN, Relational Graph Convolutional Networks (R-GCNs), για την πρόβλεψη μελλοντικών συνεργασιών.

2.5 Μάθηση με αναπαράσταση - Representation Learning

Η μάθηση με αναπαράσταση είναι μια λειτουργία των Νευρωνικών Δικτύων, βασική πτυχή της βαθιάς μάθησης. Ο στόχος της εκμάθησης αναπαράστασης είναι η αυτόματη εκμάθηση αναπαραστάσεων - δηλαδή χαρακτηριστικών ή ιδιοτήτων - από ακατέργαστα δεδομένα εισόδου. Αυτές οι μαθημένες αναπαραστάσεις μπορούν στη συνέχεια να χρησιμοποιηθούν για διάφορες εργασίες (π.χ., ταξινόμηση, παλινδρόμηση ή ομαδοποίηση). Τα νευρωνικά δίκτυα, είναι ιδιαίτερα αποτελεσματικά στη μάθηση αναπαράστασης. Μπορούν να μάθουν αυτόματα σύνθετες αναπαραστάσεις δεδομένων εκπαιδεύοντας σε μεγάλο όγκο δεδομένων με κατάλληλη αντικειμενική συνάρτηση. Οι αναπαραστάσεις συνήθως μαθαίνονται στα κρυφά επίπεδα του νευρωνικού δικτύου.

Μια διαδικασία λειτουργίας αναπαράστασης ξεκινάει με τη μετατροπή των δεδομένων εισόδου σε ένα υψηλότερο επίπεδο αφαίρεσης (LeCun et al., 2015; Goodfellow et al. 2016). Τα αρχικά επίπεδα μπορεί να μάθουν να ανιχνεύουν απλά μοτίβα (όπως ακμές σε μια εικόνα), ενώ τα βαθύτερα επίπεδα μαθαίνουν να αντιπροσωπεύουν πιο σύνθετα χαρακτηριστικά (όπως σχήματα ή αντικείμενα). Τα νευρωνικά δίκτυα μαθαίνουν αναπαραστάσεις μέσω της διαδικασίας backpropagation και gradient descent, όπου το δίκτυο κάνει προβλέψεις με βάση τις τρέχουσες αναπαραστάσεις του. Η διαφορά μεταξύ της πρόβλεψης του δικτύου και της πραγματικής αλήθειας υπολογίζεται χρησιμοποιώντας μια συνάρτηση απώλειας. Αυτό το σφάλμα στη συνέχεια διαδίδεται ξανά μέσω του δικτύου, προσαρμόζοντας τα βάρη για να ελαχιστοποιηθεί η απώλεια. Αυτό έχει ως αποτέλεσμα το δίκτυο να μαθαίνει πιο ακριβείς αναπαραστάσεις των δεδομένων.

Ένα βασικό πλεονέκτημα της εκμάθησης αναπαράστασης στα νευρωνικά δίκτυα είναι ότι επιτρέπει τη μάθηση από άκρο σε άκρο. Όπως αναφέρουν οι Pan and Yang (2020), το δίκτυο μαθαίνει τόσο τις αναπαραστάσεις όσο και τη λειτουργία πρόβλεψης συγκεκριμένης εργασίας μαζί, που συχνά οδηγούν σε καλύτερη απόδοση από ό,τι αν οι αναπαραστάσεις και η συνάρτηση πρόβλεψης είχαν μαθευτεί ξεχωριστά. Οι μαθημένες αναπαραστάσεις μπορούν συχνά να επαναχρησιμοποιηθούν σε όλες τις εργασίες. Αυτό είναι γνωστό ως μάθηση μεταφοράς. Για παράδειγμα, οι Wu et al. (2020) αναφέρουν ότι ένα νευρωνικό δίκτυο που έχει εκπαιδευτεί να αναγνωρίζει αντικείμενα σε εικόνες μπορεί να μάθει αναπαραστάσεις που είναι χρήσιμες για μια άλλη εργασία, όπως η δημιουργία λεζάντας εικόνων.

2.5.1 Μάθηση με αναπαράσταση για επεξεργασία φυσικής γλώσσας

Η μάθηση με αναπαράσταση είναι ένα κρίσιμο συστατικό πολλών εργασιών επεξεργασίας φυσικής γλώσσας (NLP). Συνήθως περιλαμβάνει την αντιστοίχιση λέξεων, φράσεων, προτάσεων ή ακόμα και ολόκληρων εγγράφων σε διανυσματικά κενά, έτσι ώστε οι σημασιολογικές σχέσεις μεταξύ των γλωσσικών στοιχείων να αντικατοπτρίζονται στις γεωμετρικές σχέσεις μεταξύ των ενσωματώσεών τους. Κάποιες από τις πιο συνηθισμένες τεχνικές ως προς την εκμάθηση αναπαράστασης στην επεξεργασία φυσικής γλώσσας είναι οι:

- Ενσωματώσεις λέξεων: Τεχνικές όπως το Word2Vec (Mikolov et al., 2013) και το GloVe (Pennington et al., 2014) μαθαίνουν διανυσματικές αναπαραστάσεις λέξεων από μεγάλες ποσότητες δεδομένων κειμένου. Αυτές οι αναπαραστάσεις αποτυπώνουν σημασιολογικές και συντακτικές ομοιότητες μεταξύ των λέξεων.
- Ενσωματώσεις λέξεων με βάση τα συμφραζόμενα: Σε αντίθεση με το Word2Vec και το GloVe που δημιουργούν ένα μόνο διάνυσμα για κάθε λέξη ανεξάρτητα από το περιεχόμενό της, ενσωματώσεις με βάση τα συμφραζόμενα όπως ELMo (Peters et al., 2018), BERT (Devlin et al., 2018) και GPT (Radford et al., 2018) δημιουργούν διαφορετικές ενσωματώσεις για μια λέξη ανάλογα με το περιεχόμενό της. Αυτό τους δίνει τη δυνατότητα να συλλάβουν πιο λεπτές έννοιες και χρήσεις λέξεων.

- Ενσωματώσεις πρότασης/έγγραφου: Τεχνικές όπως το Doc2Vec (Le and Mikolov, 2014) επεκτείνουν το Word2Vec για να δημιουργήσει αναπαραστάσεις ολόκληρων προτάσεων ή εγγράφων. Πιο πρόσφατες προσεγγίσεις αξιοποιούν μοντέλα που βασίζονται σε μετασχηματιστές (όπως BERT ή GPT) για να δημιουργήσουν αναπαραστάσεις προτάσεων ή εγγράφων συγκεντρώνοντας τις ενσωματώσεις των λέξεων που περιέχουν.
- Προεκπαιδευμένα μοντέλα γλώσσας: Μεγάλης κλίμακας προεκπαιδευμένα μοντέλα όπως BERT, GPT-2/3 (Radford et al., 2020) και RoBERTa (Liu et al., 2019) έχουν εκπαιδευτεί να μαθαίνουν πλούσιες αναπαραστάσεις της γλώσσας από μεγάλες ποσότητες των δεδομένων. Αυτές οι αναπαραστάσεις μπορούν να ρυθμιστούν με ακρίβεια για συγκεκριμένες εργασίες NLP, επιτυγχάνοντας συχνά επιδόσεις αιχμής.

Οι παραπάνω αναπαραστάσεις χρησιμοποιούνται για ένα ευρύ φάσμα εργασιών NLP, συμπεριλαμβανομένης της ταξινόμησης κειμένου, της αναγνώρισης ονομαστικών οντοτήτων, της απάντησης ερωτήσεων και της αυτόματης μετάφρασης.

2.5.2 Αλγόριθμος Word2vec για εκμάθηση αναπαραστάσεων λέξεων

Όπως αναφέρθηκε προηγουμένως, το Word2Vec είναι ένας δημοφιλής αλγόριθμος για την εκμάθηση πυκνών διανυσματικών αναπαραστάσεων λέξεων, γνωστός ως ενσωματώσεις λέξεων. Αυτές οι ενσωματώσεις έχουν σχεδιαστεί για να καταγράφουν σημασιολογικές και συντακτικές σχέσεις μεταξύ των λέξεων, έτσι ώστε λέξεις με παρόμοια σημασία ή ρόλους σε μια πρόταση να έχουν παρόμοιες ενσωματώσεις. Ο αλγόριθμος εισήχθη από τους Mikolov et al. (2013), για εργασία με τίτλο «Αποτελεσματική εκτίμηση των αναπαραστάσεων λέξεων σε διανυσματικό χώρο» για λογαριασμό της Google. Περιλαμβάνει δύο κύριες αρχιτεκτονικές: Continuous Bag of Words (CBOW) και Skip-Gram.

- CBOW: Σε αυτό το μοντέλο, το νευρωνικό δίκτυο προβλέπει την τρέχουσα λέξη με βάση το πλαίσιο. Η είσοδος στο μοντέλο είναι ένα παράθυρο λέξεων περιβάλλοντος περιβάλλοντος και η έξοδος είναι η ίδια η κεντρική λέξη.

- Skip-Gram: Αυτό το μοντέλο λειτουργεί αντίστροφα, προβλέποντας λέξεις περιβάλλοντος με δεδομένη την τρέχουσα λέξη. Η είσοδος στο μοντέλο είναι μία λέξη και η έξοδος είναι οι λέξεις που το περιβάλλουν μέσα σε ένα συγκεκριμένο παράθυρο.

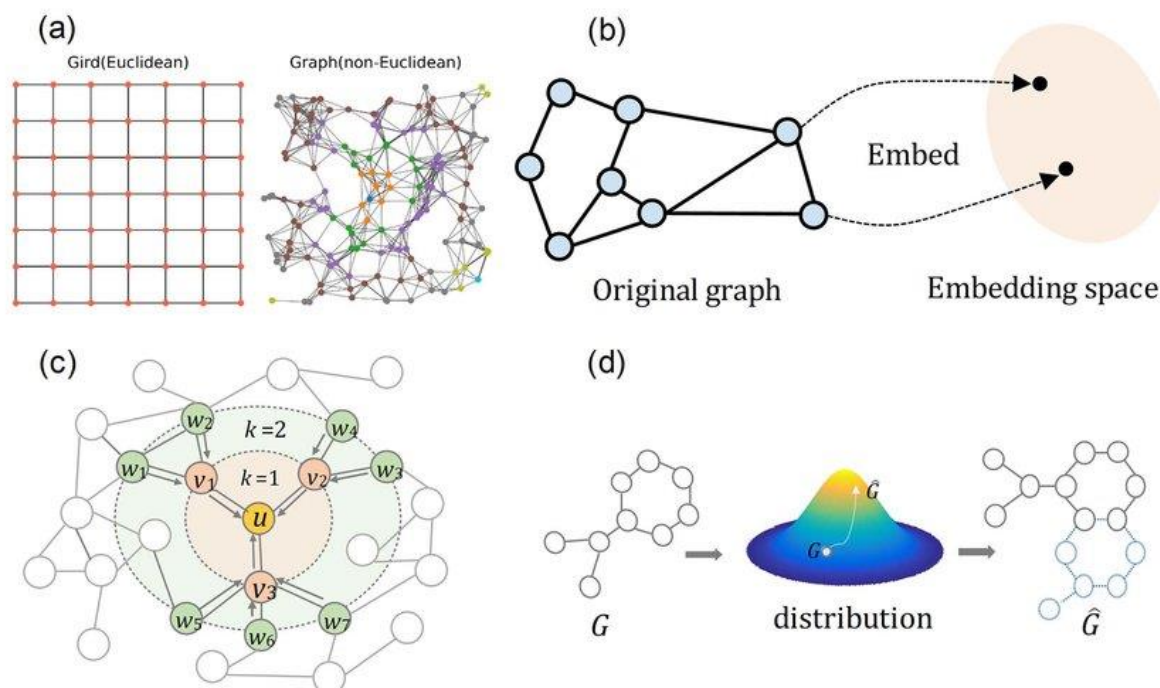
Τα παραπάνω μοντέλα εκπαιδεύονται σε μεγάλα σώματα κειμένου χρησιμοποιώντας ένα απλό νευρωνικό δίκτυο μιας στρώσης, τα βάρη του οποίου σχηματίζουν τις μαθημένες ενσωματώσεις λέξεων μετά την εκπαίδευση. Το Word2Vec έχει μερικές ενδιαφέρουσες ιδιότητες. Ενώ το Word2Vec ήταν πρωτοποριακό στην εισαγωγή του και εξακολουθεί να χρησιμοποιείται ευρέως, αν και υπήρξαν περαιτέρω εξελίξεις στον τομέα, όπως GloVe, FastText και πιο πρόσφατη ενσωμάτωση λέξεων με βάση τα μοντέλα όπως τα ELMo, BERT και GPT (Radford et al., 2018).

2.6 Μάθηση αναπαραστάσεων γράφων – Graph representation learning

Η Μάθηση γράφων, αναφέρεται σε μια κατηγορία τεχνικών μηχανικής εκμάθησης που χρησιμοποιούνται σε περιπτώσεις που τα δεδομένα είναι γράφοι. Οι γράφοι, με αυτή την έννοια, είναι μαθηματικές δομές που αποτελούνται από κόμβους (ή κορυφές) και ακμές. Οι κόμβοι αντιπροσωπεύουν οντότητες και οι ακμές αντιπροσωπεύουν σχέσεις ή αλληλεπιδράσεις μεταξύ αυτών των οντοτήτων. Οι αλγόριθμοι εκμάθησης γράφων έχουν σχεδιαστεί για να μαθαίνουν από και να κάνουν προβλέψεις σχετικά με δεδομένα που έχουν δομή γράφου. Είναι κατάλληλα για την αντιμετώπιση σύνθετων, ακανόνιστων δεδομένων και έχουν εφαρμοστεί σε πολλούς τομείς, όπως η ανάλυση κοινωνικών δικτύων, η βιοπληροφορική, τα συστήματα συστάσεων και η επεξεργασία φυσικής γλώσσας (Kipf and Welling, 2017).

Παρακάτω παρουσιάζεται η μάθηση αναπαράστασης γράφων, μια τεχνική που συνδυάζει μη-Ευκλείδεια δεδομένα γράφων με σύγχρονες τεχνικές μηχανικής μάθησης. εικόνα περιλαμβάνει ένα σύνολο γραφικών μεθόδων αναπαράστασης, καθώς και δίκτυα νευρώνων γράφων, χωρισμένα στις ακόλουθες ενότητες:

- (a) Απεικόνιση δεδομένων παρόμοιων με πλέγματα Ευκλείδειων ορθογωνικών συντεταγμένων έναντι μη-Ευκλείδειων γράφων (μη κατάλληλα για μάθηση).
- (b) Παρουσίαση μεθόδων αναπαράστασης γράφων, που επιτρέπουν τη μετατροπή των κόμβων ενός γράφου σε χαμηλοδιάστατους χώρους αναπαράστασης.
- (c) Στο επόμενο στάδιο, γίνεται αναφορά στα νευρωνικά δίκτυα γράφων, τα οποία μαθαίνουν αναπαραστάσεις γράφων μέσω διάφορων τεχνικών, όπως η συγκέντρωση και η διάδοση μηνυμάτων.
- (d) Τέλος, παρουσιάζονται τα γεννητικά μοντέλα γράφων, τα οποία μαθαίνουν την κατανομή των εισαγόμενων δειγμάτων για τη δημιουργία μοριακών γράφων με επιθυμητές ιδιότητες.



Εικόνα 21 Παραδείγματα μάθησης αναπαράστασης γράφων.
Πηγή: (Yi et al., 2021)

Κάποιες βασικές έννοιες και τεχνικές για τη μάθηση γράφων είναι τα Νευρωνικά Δίκτυα Γράφων που αναλύθηκαν παραπάνω, η ταξινόμηση κόμβων, της οποίας στόχος είναι η πρόβλεψη ετικετών για κόμβους σε ένα γράφο, η πρόβλεψη συνδέσμου που περιλαμβάνει την πρόβλεψη της ύπαρξης ακμής μεταξύ δύο κόμβων σε ένα γράφο, η ταξινόμηση γράφου που είναι η πρόβλεψη μιας ετικέτας για ένα ολόκληρο γράφο (Wu et al., 2020).

Ως προς τη διάκρισή τους, υπάρχουν διάφορες κατηγορίες για την εκμάθηση αναπαράστασης γράφων, η καθεμία με τις μοναδικές της μεθοδολογίες και εφαρμογές. Οι πιο βασικές κατηγορίες είναι οι εξής:

- Ενσωματώσεις κόμβων (Node Embeddings): Σε αυτήν την κατηγορία, οι αλγόριθμοι στοχεύουν να μάθουν μια διανυσματική αναπαράσταση (ή ενσωμάτωση) για κάθε κόμβο σε ένα γράφο. Αυτές οι ενσωματώσεις έχουν σχεδιαστεί για να καταγράφουν τον ρόλο και τη θέση του κόμβου στη δομή του γράφου. Οι κλασικοί αλγόριθμοι αυτής της κατηγορίας περιλαμβάνουν τον DeepWalk (Perozzi et al., 2014) και τον node2vec (Grover & Leskovec, 2016). Τα νευρωνικά δίκτυα γράφων όπως το Graph Convolutional Networks (GCN) και το GraphSAGE μαθαίνουν επίσης ενσωματώσεις κόμβων.

- Ενσωματώσεις άκρων (Edge Embeddings): Αυτοί οι αλγόριθμοι μαθαίνουν αναπαραστάσεις για τις ακμές σε ένα γράφο, οι οποίες μπορούν να είναι χρήσιμες για εργασίες όπως η πρόβλεψη ακμών. Οι μέθοδοι για την εκμάθηση των ενσωματώσεων ακμών συχνά περιλαμβάνουν μετασχηματισμό και συνδυασμό των ενσωματώσεων των κόμβων που συνδέει η ακμή.
- Ενσωματώσεις γράφου (Graph Embeddings): Αυτοί οι αλγόριθμοι στοχεύουν να μάθουν μια ενσωμάτωση για ένα ολόκληρο γράφο, το οποίο μπορεί να είναι χρήσιμο για εργασίες όπως η ταξινόμηση γράφων. Η εκμάθηση αναπαράστασης σε επίπεδο γράφου συχνά περιλαμβάνει κάποια μορφή συνάθροισης ενσωματώσεων κόμβων (ή μερικές φορές ακμών). Τα νευρωνικά δίκτυα γράφων για εκμάθηση σε επίπεδο γράφου περιλαμβάνουν τα δίκτυα ισομορφισμού γράφων (GIN) και τα δίκτυα προσοχής γράφου (GAT) (Velickovic et al., 2018).
- Υβριδικά μοντέλα (Hybrid Models): Ορισμένα μοντέλα εκμάθησης αναπαράστασης γράφων μαθαίνουν ενσωματώσεις για κόμβους, ακμές και ολόκληρους γράφους, μαζί. Συχνά έχουν σχεδιαστεί για να χειρίζονται πολυτροπικά δεδομένα, πολυσχεσιακά δεδομένα ή ετερογενή γράφους (γράφους με πολλαπλούς τύπους κόμβων ή ακμών).

Σύνοψη

Στη διατριβή αυτή αναλύσαμε τους βασικούς πυλώνες της Μηχανικής Μάθησης, των Νευρωνικών Δικτύων και των Νευρωνικών δικτύων Γράφων, αλλά ταυτόχρονα καλύψαμε και νέες τεχνολογικές εφαρμογές τους. Στο πρώτο κεφάλαιο, επισημάναμε την σημαντικότητα της τεχνητής νοημοσύνης στον σύγχρονο κόσμο. Αναδείξαμε τις διάφορες προσεγγίσεις στην μηχανική μάθηση, εξετάζοντας τα χαρακτηριστικά και τις δυνατότητες της εποπτευόμενης, της μη εποπτευόμενης και της ενισχυτικής μάθησης. Αναφερθήκαμε στην βασική συλλογή αλγορίθμων που συνιστούν την εποπτευόμενη και μη εποπτευόμενη μηχανική μάθηση, όπως τα Decision Trees, K-MEANS, Random Forests, K-Nearest Neighbors, DBSCAN και αποτελούν την μαθηματική βάση για τις νεότερες εξελίξεις της μηχανικής μάθησης.

Στο δεύτερο κεφάλαιο, δώσαμε έμφαση στα Νευρωνικά Δίκτυα. Από το απλό δίκτυο Perceptron μέχρι τα πολυεπίπεδα NN, παρατηρήσαμε την διαδικασία εκπαίδευσης δικτύων μέσω της τεχνικής backpropagation και επισημάναμε τα διάφορα προβλήματα που μπορεί να αντιμετωπίσει κανείς κατά την εκπαίδευση. Το κεφάλαιο κορυφώνεται με την ανάλυση των νευρωνικών δικτύων γράφων και των πρωτοποριακών εφαρμογών τους στην εποχή των ραγδαίων τεχνολογικών εξελίξεων. Τέλος, εξετάσαμε την πρωτοποριακή περιοχή της μάθησης με αναπαράσταση, εστιάζοντας στην επεξεργασία φυσικής γλώσσας και την αναπαράσταση γράφων, αναγνωρίζοντας την ολοένα αυξανόμενη σημασία της στον τομέα της τεχνητής νοημοσύνης.

Συνολικά, αυτή η διατριβή αποτελεί μια συνολική επισκόπηση των τεχνολογιών Μηχανικής Μάθησης και Νευρωνικών Δικτύων, υπογραμμίζοντας τη σημασία και την αναγκαιότητα συνεχούς έρευνας στον τομέα αυτό.

Βιβλιογραφία

1. Abiodun M. Ikotun, Absalom E. Ezugwu, Laith Abualigah, Belal Abuhaija, Jia Heming, K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data, Information Sciences, Volume 622, 2023, Pages 178-210, ISSN 0020-0255, <https://doi.org/10.1016/j.ins.2022.11.139>.
2. Aggarwal, C. C. (2023). Neural Networks and deep learning a textbook. Cham: Springer International Publishing AG.
3. Aha, D. W., Kibler, D., & Albert, M. K. (1991). Instance-based learning algorithms. Machine Learning, 6(1), 37-66.
4. Alessandri, A., & Gaggero, M. (2017). Fast Moving Horizon State Estimation for Discrete-Time Systems Using Single and Multi Iteration Descent Methods.
5. Alzubi, J., Nayyar, A., & Kumar, A. (2018). Machine Learning from Theory to Algorithms: An Overview. Journal of Physics: Conference Series, 1142, 012012. Second National Conference on Computational Intelligence (NCCI 2018), Bangalore, India. <https://doi.org/10.1088/1742-6596/1142/1/012012>
6. Arthur, D., & Vassilvitskii, S. (2007). K-means++: The advantages of careful seeding. In Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms (pp. 1027-1035).
7. Battaglia, P. W., Pascanu, R., Lai, M., Rezende, D., & Kavukcuoglu, K. (2016). Interaction Networks for Learning about Objects, Relations and Physics. arXiv preprint arXiv:1612.00222.
8. Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer.
9. Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In Proceedings of the Fifth Annual Workshop on Computational Learning Theory (pp. 144-152).
10. Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

11. Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. A. (1984). Classification and Regression Trees. CRC press.
12. Cartis, C., Fiala, J., Marteau, B., & Roberts, L. (2018). Improving the Flexibility and Robustness of Model-based Derivative-free Optimization Solvers.
13. Chen, C. (2020). Machine learning for finance: a comprehensive overview. Journal of Financial Data Science, 2(2), 1-14.
14. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.
15. Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine learning, 20(3), 273-297.
16. Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. IEEE Transactions on Information Theory, 13(1), 21-27.
17. Cui, M. (2020). Introduction to the K-Means Clustering Algorithm Based on the Elbow Method. Accounting, Auditing and Finance, 1, 5-8. Clausius Scientific Press, Canada. DOI: 10.23977/accaf.2020.010102. ISSN 2616-3136.
18. Das, R., & Soylu, M. (2023). A key review on graph data science: The power of graphs in scientific studies. *Chemometrics and Intelligent Laboratory Systems*, 240, 104896. <https://doi.org/0169-7439>
19. Dauphin, Y. N., Pascanu, R., Gulcehre, C., Cho, K., Ganguli, S., & Bengio, Y. (2014). Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. Advances in Neural Information Processing Systems, 27.
20. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.
21. Dietterich, T. G. (2000). Ensemble methods in machine learning. In International Workshop on Multiple Classifier Systems (pp. 1-15). Springer.

22. Djordjevic, I. B. (2021). Quantum Machine Learning. In Quantum Information Processing, Quantum Computing, and Quantum Error Correction (Second Edition). Elsevier.
23. Draper, N. R., & Smith, H. (1998). Applied Regression Analysis (3rd ed.). John Wiley & Sons.
24. Drucker, H., Burges, C. J., Kaufman, L., Smola, A. J., & Vapnik, V. (1997). Support vector regression machines. In Advances in Neural Information Processing Systems (pp. 155-161).
25. Eliasy, Ashkan & Przychodzen, Justyna. (2020). The role of AI in capital structure to enhance corporate funding strategies. Array. 6. 100017. 10.1016/j.array.2020.100017.
26. Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96) (pp. 226-231).
27. Fang, Xianjin & Yu, Fangchao & Yang, Gaoming & Qu, Youyang. (2019). Regression Analysis With Differential Privacy Preserving. IEEE Access. PP. 1-1. 10.1109/ACCESS.2019.2940714.
28. Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. Annals of statistics, 1189-1232.
29. García Cabello, J. (2019). A model towards global demographics: An application—a universal bank branch geolocator based on branch size. *Soft Computing*, 23, 7193-7205. <https://doi.org/10.1007/s00500-018-3362-z>
30. Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to forget: Continual prediction with LSTM. Neural Computation, 12(10), 2451-2471.
31. Gilpin, L. H., Bau, D., Yuan, B. Z., Bajwa, A., Specter, M., & Kagal, L. (2018). Explaining explanations: An overview of interpretability of machine learning. In 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA) (pp. 80-89). IEEE.
32. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.

33. Graves, A., & Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 18(5-6), 602-610.
34. Grover, A., & Leskovec, J. (2016). node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 855-864).
35. Hamilton, W. L., Ying, R., & Leskovec, J. (2018). Inductive Representation Learning on Large Graphs. *arXiv preprint arXiv:1706.02216*. (cs.SI).
36. Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer.
37. Hawkins, D. M. (2004). The problem of overfitting. *Journal of Chemical Information and Computer Sciences*, 44(1), 1-12.
38. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770-778).
39. Hinton, G. E., & Sejnowski, T. J. (1999). *Unsupervised Learning: Foundations of Neural Computation*. MIT Press.
40. Hochreiter, S., Bengio, Y., Frasconi, P., & Schmidhuber, J. (2001). Gradient flow in recurrent nets: the difficulty of learning long-term dependencies. In *A field guide to dynamical recurrent networks* (pp. 237-244). IEEE Press.
41. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735-1780.
42. Hosmer Jr, D. W., Lemeshow, S., & Sturdivant, R. X. (2013). *Applied Logistic Regression* (3rd ed.). John Wiley & Sons.
43. Hubel, D. H., & Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology*, 160(1), 106-154.

44. Jain, A., Liu, I., Sarda, A., & Molino, P. (2019, December 4). Food Discovery with Uber Eats: Using Graph Learning to Power Recommendations. Uber Blog. Retrieved from <https://www.uber.com/en-GR/blog/uber-eats-graph-learning/>
45. Johnson, S. C. (1967). Hierarchical clustering schemes. *Psychometrika*, 32(3), 241-254.
46. Jumper, J., Evans, R., Pritzel, A. et al. Highly accurate protein structure prediction with AlphaFold. *Nature* 596, 583–589 (2021). <https://doi.org/10.1038/s41586-021-03819-2>
47. Kim, Yonsoo & Kim, Soojun & Kang, Narae & Kim, Taegyun & Kim, Hungsoo. (2014). Estimation of Frequency Based Snowfall Depth Considering Climate Change Using Neural Network. *Journal of Korean Society of Hazard Mitigation*. 14. 10.9798/KOSHAM.2014.14.1.93.
48. Kipf, T. N., & Welling, M. (2017). Semi-Supervised Classification with Graph Convolutional Networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*.
49. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems* (pp. 1097-1105).
50. Kumar, V., & Rajput, D. S. (Eds.). (2023). *Concepts and techniques of graph neural network*. Engineering Science Reference. <https://lccn.loc.gov/2022052611>
51. Le, Q., & Mikolov, T. (2014). Distributed Representations of Sentences and Documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*.
52. LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2324.
53. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436-444.
54. Liao, Y., Desaraju, V. R., & Seidmann, A. (2007). An integrated assortment and inventory model for perishable products. *Management Science*, 53(12), 1863-1878.
55. Lin, M., Chen, Q., & Yan, S. (2013). Network in network. *arXiv preprint arXiv:1312.4400*.

56. Linderman, S. W., Johnson, M. J., & Adams, R. P. (2017). Scalable Bayesian inference for excitatory point process networks. In International Conference on Machine Learning (pp. 2088-2097).
57. Manning, C. D., Raghavan, P., & Schütze, H. (2008). Introduction to Information Retrieval. Cambridge University Press.
58. Masui, T. (2022, October 17). Graph neural networks with PYG on node classification, link prediction, and anomaly detection. Medium. <https://towardsdatascience.com/graph-neural-networks-with-pyg-on-node-classification-link-prediction-and-anomaly-detection-14aa38fe1275>
59. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.
60. Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems (pp. 3111-3119).
61. Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2013). Playing Atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602.
62. Murphy, K. P. (2012). Machine Learning: A Probabilistic Perspective. MIT Press.
63. Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted Boltzmann machines. In Proceedings of the 27th international conference on international conference on machine learning (pp. 807-814).
64. Olah, C. (2015). Understanding LSTM networks. <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>
65. Pande, A., Cui, A., Dabak, A., Ansari, S., Luo, X., Cui, B., ... & Zaidi, Q. (2018). Probabilistic framework for instantaneous adaptation in visual processing. Proceedings of the National Academy of Sciences, 115(12), 2986-2991.

66. Pearl, J. (1988). Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann.
67. Pearl, J. (2000). Causality: Models, Reasoning, and Inference. Cambridge University Press.
68. Pimentel, J. M., Clifton, D. A., Clifton, L., & Tarassenko, L. (2014). A review of novelty detection. *Signal Processing*, 99, 215-249.
69. Rendle, S., Freudenthaler, C., Gantner, Z., & Schmidt-Thieme, L. (2009). BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence* (pp. 452-461).
70. Resnik, P. (1999). Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 11, 95-130.
71. Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention* (pp. 234-241). Springer.
72. Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386–408. <https://doi.org/10.1037/h0042519>
73. Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088), 533-536.
74. Salakhutdinov, R., & Hinton, G. E. (2009). Semantic hashing. *International Journal of Approximate Reasoning*, 50(7), 969-978.
75. Sankar, A., & Karlapalem, K. (2003). An efficient algorithm for top-k high-dimensional pattern search. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 614-619).
76. Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2000). Analysis of recommendation algorithms for e-commerce. In *Proceedings of the 2nd ACM conference on Electronic commerce* (pp. 158-167).

77. Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, 61, 85-117.
78. Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J., & Williamson, R. C. (2001). Estimating the Support of a High-Dimensional Distribution. *Neural Computation*, 13(7), 1443-1471.
79. Schölkopf, B., & Smola, A. J. (2002). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press.
80. Sejdinovic, D., Sriperumbudur, B., Gretton, A., & Fukumizu, K. (2013). Equivalence of distance-based and RKHS-based statistics in hypothesis testing. *The Annals of Statistics*, 41(5), 2263-2291.
81. Shalev-Shwartz, S., & Ben-David, S. (2014). *Understanding machine learning: From theory to algorithms*. Cambridge University Press.
82. Siddharth Misra, Hao Li Noninvasive fracture characterization based on the classification of sonic wave travel times, *Machine Learning for Subsurface Characterization*, Gulf Professional Publishing, 2020
83. Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., ... & Hassabis, D. (2017). Mastering the game of Go without human knowledge. *Nature*, 550(7676), 354-359.
84. Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
85. Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1), 1929-1958.
86. Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press Cambridge.

87. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1-9).
88. Tadepalli, P., & Ok, D. (1996). Scaling up reinforcement learning by factored value functions. In Machine Learning Proceedings 1996 (pp. 511-519). Elsevier.
89. Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., & Mei, Q. (2015). Line: Large-scale information network embedding. In Proceedings of the 24th international conference on world wide web (pp. 1067-1077).
90. Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society: Series B (Methodological), 58(1), 267-288.
91. Tommasi, T., Quadrianto, N., Caputo, B., & Lampert, C. H. (2014). Beyond Dataset Bias: Multi-level Contextual Categorization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 1753-1760).
92. Ullman, S. (1980). Against direct perception. The Behavioral and Brain Sciences, 3(1), 1-22.
93. Vapnik, V. (1995). The Nature of Statistical Learning Theory. Springer.
94. Vapnik, V. (1998). Statistical Learning Theory. Wiley.
95. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems (pp. 30-31).
96. Wang, D., Cui, P., Zhu, W., & Yang, S. (2016). Structural deep network embedding. In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (pp. 1225-1234).
97. Wang, Y., Ma, J., & Ye, J. (2017). A fast parallel SGD for matrix factorization in shared memory systems. In Proceedings of the Tenth ACM International Conference on Web Search and Data Mining (pp. 767-775).

98. Weng, L., & Perona, P. (2013). Conditional models for contextual human motion prediction. In 2013 IEEE Conference on Computer Vision and Pattern Recognition (pp. 3023-3030). IEEE.
99. Weston, J., Ratle, F., Mobahi, H., & Collobert, R. (2011). Deep learning via semi-supervised embedding. In Proceedings of the 28th International Conference on International Conference on Machine Learning (ICML'11) (pp. 1168-1175).
100. Witten, I. H., Frank, E., & Hall, M. A. (2011). Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann.
101. Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., ... & Klingner, J. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144.
102. Xie, L., & Yuille, A. (2016). Holistically-nested edge detection. In Proceedings of the IEEE international conference on computer vision (pp. 1395-1403).
103. Xu, J., Zhang, L., & Wang, S. (2017). A survey of network embedding methods. arXiv preprint arXiv:1711.08752.
104. Yan, S., Xiong, Y., & Lin, D. (2007). Spatial pyramid pooling in deep convolutional networks for visual recognition. In Proceedings of the IEEE international conference on computer vision (pp. 3464-3471).
105. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., & Hovy, E. (2016). Hierarchical attention networks for document classification. In Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Human language technologies (pp. 1480-1489).
106. Yi, H. -C., You, Z. -H., Huang, D. -S., & Kwoh, C. K. (2021). Graph representation learning in bioinformatics: trends, methods and applications. Briefings in Bioinformatics, 23(1), bbab340. <https://doi.org/10.1093/bib/bbab340>
107. Zhang, W., Du, S., Wang, X., Li, L., Yang, Y., & Wang, J. (2018). Scalable large graph learning: Stochastic gradient descent with a feasible subsample size. In Proceedings of the

- 24th ACM SIGKDD international conference on knowledge discovery & data mining (pp. 298-307).
108. Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level convolutional networks for text classification. In *Advances in neural information processing systems* (pp. 649-657).
109. Zhang, X., Zhao, J., & LeCun, Y. (2016). Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics* (pp. 288-296).
110. Zhao, W. X., Ma, H., & He, Q. (2015). Recommender systems based on user reviews: the state of the art. *User Modeling and User-Adapted Interaction*, 25(2), 99-154.
111. Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C., & Sun, M. (2020). Graph neural networks: A review of methods and applications. *AI Open*, 1, 57-81. <https://doi.org/10.1016/j.aiopen.2021.01.001>
112. Zhou, P., Qi, Z., Zheng, S., Xu, K., Bao, H., & Xu, B. (2017). EAST: An efficient and accurate scene text detector. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2642-2651).