

Μηχανική Μάθηση και Τεχνητή Νοημοσύνη στην επιστήμη του Μηχανικού: Εφαρμογή στην εκτίμηση της θεμελιώδους ιδιοπεριόδου κτιρίων

Αριστοτέλης Χαραλαμπίδης¹

ΠΕΡΙΛΗΨΗ

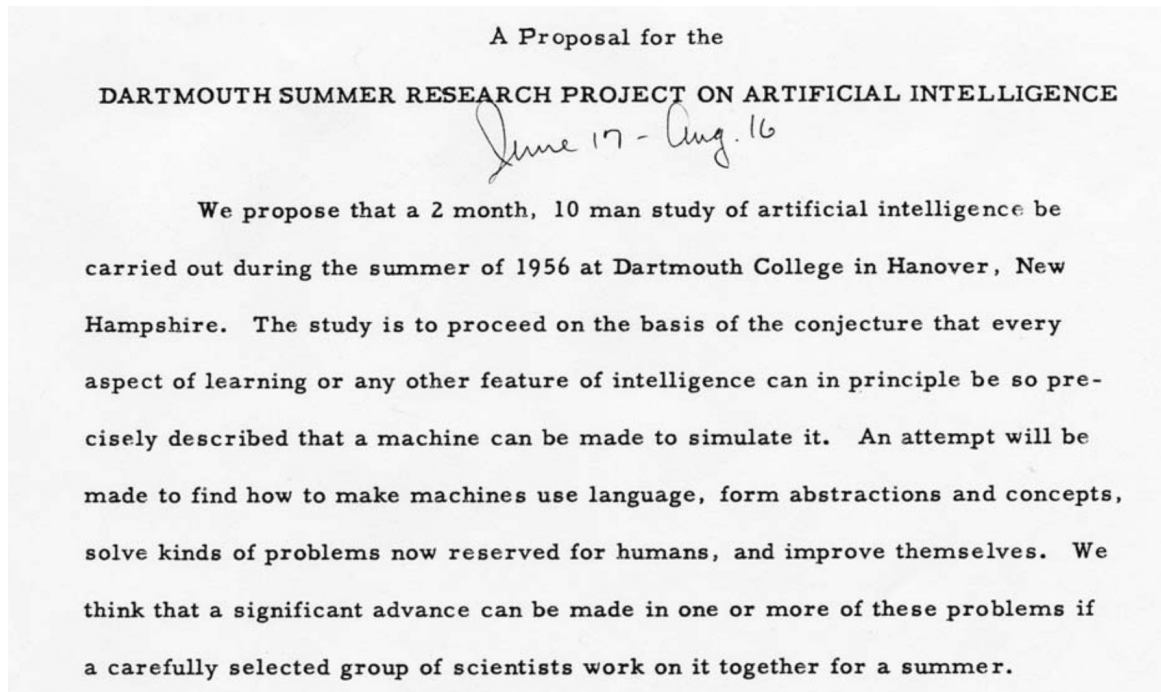
Η Μηχανική Μάθηση (Machine Learning – ML) και η Τεχνητή Νοημοσύνη – TN (Artificial Intelligence – AI) αναδύονται δυναμικά ως εργαλεία-κλειδιά στην επιστήμη του Μηχανικού, μετασχηματίζοντας τόσο την έρευνα όσο και την καθημερινή πράξη. Στην εργασία αυτή παρουσιάζεται μια ιστορική αναδρομή των τεχνολογιών αυτών, αναδεικνύοντας πώς αυτές έχουν ενσωματωθεί και προσαρμοστεί για να καλύψουν τις ανάγκες του Μηχανικού. Η ιστορική αναδρομή ξεκινά από τα πρώιμα rule-based expert systems και την αλγοριθμική εποχή, φτάνοντας στη ριζική τομή που επέφερε η εκπαίδευση βαθιών Τεχνητών Νευρωνικών Δικτύων – ΤΝΔ (deep learning of Artificial Neural Networks – ANNs), με χρήση τεράστιων ποσοτήτων δεδομένων και αυξημένων υπολογιστικών πόρων. Στο δεύτερο μέρος της εργασίας γίνεται μια εφαρμογή μεθόδων Μηχανικής Μάθησης σε ένα απλό πρόβλημα εκτίμησης της θεμελιώδους ιδιοπεριόδου μιας κατασκευής. Παρουσιάζεται ένα νέο ΤΝΔ το οποίο συνδυάζει εξαιρετική απόδοση και μικρό μέγεθος. Σημαντικό πλεονέκτημα του είναι και η απλότητα χρήσης του, αφού έχει μετατραπεί αλγοριθμικά σε συνάρτηση και έχει ενσωματωθεί σε φύλλο εργασίας. Η εργασία ολοκληρώνεται με επισημάνσεις γύρω από τις μεθοδολογικές παγίδες και τους περιορισμούς που πρέπει να λαμβάνονται υπόψη σε εφαρμογές Μηχανικής Μάθησης.

Λέξεις Κλειδιά: Μηχανική Μάθηση, Τεχνητή Νοημοσύνη, Τεχνητά Νευρωνικά Δίκτυα, Θεμελιώδης ιδιοπερίοδος, Τοιχοπληρώσεις.

1 ΙΣΤΟΡΙΚΗ ΑΝΑΣΚΟΠΗΣΗ

Ο όρος Τεχνητή Νοημοσύνη – TN (Artificial Intelligence – AI) εμφανίζεται για πρώτη φορά την 31^η Αυγούστου 1955, στο πλαίσιο μιας πρότασης ενός καλοκαιρινού ερευνητικού προγράμματος στο Dartmouth College με τίτλο “A PROPOSAL FOR THE DARTMOUTH SUMMER RESEARCH PROJECT ON ARTIFICIAL INTELLIGENCE”, από τους ερευνητές John McCarthy, Marvin Minsky, Nathaniel Rochester, and Claude Shannon (βλ. Σχήμα 1) [1].

¹ Επίκουρος Καθηγητής ΠΑ.Δ.Α., achar@uniwa.gr



Σχήμα 1: Η αρχή της πρότασης του καλοκαιρινού ερευνητικού προγράμματος στο Dartmouth College [1].

Ήδη από τότε, οι ερευνητές αυτοί εξέφρασαν φιλοδοξίες να αναπτύξουν υπολογιστικά συστήματα ικανά για γλώσσα, σκέψη και λήψη αποφάσεων. Παρόλη την τεράστια τεχνολογική απόσταση που μας χωρίζει από τότε (σημειώνεται εκείνη την εποχή δεν υπήρχαν καν δορυφόροι!), προκαλεί τεράστια έκπληξη και θαυμασμό η ακρίβεια και η σαφήνεια με την οποία είχαν οριοθετήσει το πρόβλημα. Ενδεικτικά:

- “**The study is to proceed on the basis of the conjecture** that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it.” δηλαδή **“Η μελέτη θα προχωρήσει με βάση την υπόθεση** ότι κάθε πτυχή της μάθησης ή οποιοδήποτε άλλο χαρακτηριστικό της νοημοσύνης μπορεί, κατ’ αρχήν, να περιγραφεί με τέτοια ακρίβεια ώστε να μπορεί να προσομοιωθεί από μια μηχανή.”. Συνεπώς, αν και δεν υπήρχε η βεβαιότητα, υπήρχε εξαρχής η πεποίθηση ότι μια μηχανή μπορεί να μάθει και να αναπαράγει οποιαδήποτε έκφανση αυτού που ονομάζουμε νοημοσύνη.
- “An attempt will be made to find how to make machines **use language**, form abstractions and concepts, solve kinds of problems now reserved for humans, and **improve themselves**.” δηλαδή «Θα γίνει μια προσπάθεια να βρεθεί τρόπος ώστε οι μηχανές **να χρησιμοποιούν τη γλώσσα**, να σχηματίζουν γενικεύσεις και έννοιες, να επιλύουν προβλήματα που σήμερα είναι προνόμιο των ανθρώπων και **να αυτοβελτιώνονται**». Αναγνωρίζεται συνεπώς η σημασία της χρήσης της γλώσσας και της αυτοβελτίωσης των μηχανών ως πυλώνες προόδου στην ΤΝ.

- “The speeds and memory capacities of present computers may be insufficient to simulate many of the higher functions of the human brain, but **the major obstacle is not lack of machine capacity, but our inability to write programs taking full advantage of what we have.**” δηλαδή «Οι ταχύτητες και οι χωρητικότητες μνήμης των σημερινών υπολογιστών μπορεί να είναι ανεπαρκείς για την προσομοίωση πολλών από τις ανώτερες λειτουργίες του ανθρώπινου εγκεφάλου, αλλά **το κύριο εμπόδιο δεν είναι η έλλειψη χωρητικότητας των μηχανών, αλλά η αδυναμία μας να γράψουμε προγράμματα που να αξιοποιούν πλήρως τις δυνατότητες που έχουμε.**». Αναγνωρίζεται το γεγονός ότι, εκείνη τη χρονική στιγμή, η πρόοδος πάνω στη ΤΝ δεν εμποδιζόταν μόνο από το υλισμικό (hardware) αλλά κυρίως από το λογισμικό (software).
- Σχετικά με τη χρήση της γλώσσας: “From this point of view, forming a generalization consists of **admitting a new word** and some rules whereby sentences containing it imply and are implied by others” δηλαδή «Από αυτή την άποψη, η διαμόρφωση μιας γενίκευσης συνίσταται **στην αποδοχή μιας νέας λέξης** και ορισμένων κανόνων, σύμφωνα με τους οποίους οι προτάσεις που την περιέχουν υπονοούν και υπονοούνται από άλλες.». Ο τρόπος με τον οποίο δουλεύουν τα σύγχρονα Μεγάλα Γλωσσικά Μοντέλα (Large Language Models – LLMs) όπως το GPT, είναι ακριβώς αυτός: συμπληρώνουν την επόμενη λέξη στο κείμενο. Ακριβέστερα, συμπληρώνουν το επόμενο λεκτικό (token), το οποίο μπορεί να είναι μία λέξη ή ένα τμήμα λέξης, ένα σημείο στίξης, το κενό κ.λπ.. Για να επιτευχθεί αυτό, δημιουργούν μια λίστα από πιθανές επιλογές για το επόμενο token. Κάθε ένα από αυτά είναι συσχετισμένο με μια πιθανότητα εμφάνισης, εξαρτώμενη από το υπόλοιπο κείμενο και τις έννοιες που αυτό περιέχει. Αφού επιλεγεί στοχαστικά το επόμενο token, η διαδικασία επαναλαμβάνεται έως ότου να ολοκληρωθεί το κείμενο.
- Σχετικά με την τυχαιότητα και τη δημιουργικότητα (randomness and creativity): “A fairly attractive and yet clearly incomplete conjecture is **that the difference between creative thinking and unimaginative competent thinking lies in the injection of a some randomness. The randomness must be guided by intuition to be efficient.**” δηλαδή «Μια αρκετά ελκυστική αλλά σαφώς ατελής υπόθεση είναι ότι **η διαφορά μεταξύ δημιουργικής σκέψης και ικανής σκέψης χωρίς φαντασία έγκειται στην εισαγωγή ενός στοιχείου τυχαιότητας. Η τυχαιότητα πρέπει να καθοδηγείται από την διαίσθηση για να είναι αποτελεσματική.**». Ο τρόπος με τον οποίο επιλέγεται το επόμενο token στο κείμενο από τα LLM, όπως περιεγράφηκε παραπάνω, είναι στοχαστικός. Εξαρτάται όμως από μια παράμετρο η οποία ονομάζεται «θερμοκρασία» (temperature). Όσο πιο μικρή είναι η θερμοκρασία, τόσο πιο ντετερμινιστικό γίνεται το σύστημα, δηλαδή, τόσο πιο συχνά επιλέγεται ως επόμενο token αυτό με την αυστηρά μεγαλύτερη πιθανότητα επιλογής στη λίστα. Όταν η θερμοκρασία πάρει τιμές μεγαλύτερες από την μονάδα, το κείμενο γίνεται δημιουργικό, απρόβλεπτο και ενδεχομένως να στερείται λογικής. Επιπλέον, το αποτέλεσμα διαφοροποιείται κάθε φορά που ζητάμε κάτι από το LLM, χρησιμοποιώντας τις ίδιες αρχικές οδηγίες (prompt). Αναγνωρίζεται συνεπώς η σημασία της τυχαιότητας για την δημιουργικότητα, αλλά επίσης αναγνωρίζεται ότι η τυχαιότητα για να είναι αποτελεσματική θα πρέπει να «καθοδηγείται από τη διαίσθηση». Όντως, η τυχαιότητα εφαρμόζεται στις υπολογισθείσες πιθανότητες εμφάνισης του επόμενου token και δεν σημαίνει τυχαία επιλογή του επόμενου token, κάτι που θα οδηγούσε σαφώς σε ακατάληπτο κείμενο.

Ο ορισμός της νοημοσύνης έχει δοθεί εύστοχα από τον John McCarthy, σπουδαίο ερευνητή με τεράστια συνεισφορά στην τεχνολογία υπολογιστών και μέλος της ερευνητικής ομάδας του Dartmouth College, όπως φαίνεται στο Σχήμα 2.

*“Intelligence is the computational part of
the ability to achieve goals in the world.”*

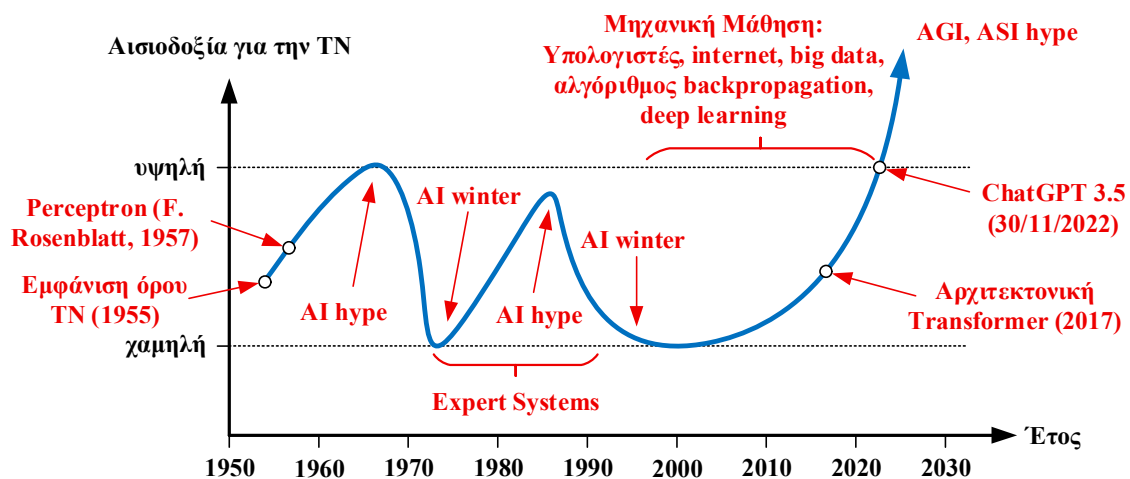
δηλαδή

*“Νοημοσύνη είναι το υπολογιστικό μέρος
της ικανότητας επίτευξης στόχων στον
πραγματικό κόσμο.”*



Σχήμα 2: Ο ορισμός της Νοημοσύνης κατά τον John McCarthy (1927–2011) [2].

Μετά την επινόηση του όρου το 1955, η αισιοδοξία σχετικά με την έκβαση της αναζήτησης της Τεχνητής Νοημοσύνης σημείωσε κατακόρυφη άνοδο (βλ. Σχήμα 3, προσαρμοσμένο από ομιλία της Melanie Mitchell [3]).



Σχήμα 3: Σχηματική αναπαράσταση της αισιοδοξίας και των προσδοκιών για την ΤΝ (προσαρμοσμένη από [3]).

Το 1957 παρουσιάστηκε από τον F. Rosenblatt το “perceptron”, μια μηχανή με χιλιάδες καλώδια η οποία μπορεί να θεωρηθεί ως η απαρχή των Τεχνητών Νευρωνικών Δικτύων – ΤΝΔ (Artificial Neural Networks – ANNs). Δημοσίευμα στην εφημερίδα NY Times τον Ιούλιο του 1958 αναφέρει “Το Πολεμικό Ναυτικό αποκάλυψε σήμερα το έμβρυο ενός ηλεκτρονικού υπολογιστή που, όπως αναμένει, θα είναι σε θέση να περπατά, να μιλά, να βλέπει, να γράφει, να αναπαράγεται και να έχει συνείδηση της ύπαρξής του.”. Στα μέσα της δεκαετίας του 1960, σημαντικοί ερευνητές,

όπως ο Claude Shannon (ιδρυτής της Θεωρίας της Πληροφορίας – Information Theory), ο Herbert Simon, ο Marvin Minsky και άλλοι είχαν προβλέψει ότι, χονδρικά, εντός 15-20 ετών θα είχε δημιουργηθεί η Τεχνητή Νοημοσύνη. Οι προσδοκίες ήταν πολύ υψηλές και διαψεύστηκαν, οπότε η έρευνα για την ΤΝ μπήκε σε «χειμώνα» (AI winter) υπό την έννοια ότι η χρηματοδότηση ήταν δύσκολη και λίγοι ερευνητές ασχολούνταν πλέον με αυτήν.

Την δεκαετία του 1980 υπήρξε μία ανάκαμψη, η οποία οφειλόταν στην ανάπτυξη των υπολογιστών καθώς και των λεγόμενων expert systems, τα οποία θεωρήθηκαν το επόμενο μεγάλο βήμα στην ΤΝ (βλ. Σχήμα 3). Πρόκειται για προγράμματα που κωδικοποιούσαν τη γνώση ειδικών σε ένα συγκεκριμένο θέμα σε ένα σύνολο κανόνων (if – then), με στόχο να μιμούνται τη διαδικασία λήψης αποφάσεων. Το πλέον χαρακτηριστικό παράδειγμα είναι το MYCIN, ένα από τα πρώτα expert systems που αναπτύχθηκαν για τη διάγνωση και τη θεραπεία λοιμώξεων του αίματος, κυρίως βακτηριακών λοιμώξεων και μηνιγγίτιδας. Αναπτύχθηκε στο πανεπιστήμιο Stanford τη δεκαετία του 1970, ήταν γραμμένο σε LISP (γλώσσα που ανέπτυξε ο John McCarthy) και χρησιμοποιούσε περίπου 600 κανόνες για να αναλύσει τα συμπτώματα των ασθενών, τα αποτελέσματα των εργαστηριακών εξετάσεων και άλλες πληροφορίες για να προτείνει θεραπεία. Σε δοκιμές, το MYCIN είχε ένα ποσοστό διάγνωσης περίπου 65%, που συχνά ξεπερνούσε αυτό των ανθρώπινων ειδικών, αλλά δεν εφαρμόστηκε ποτέ κλινικά λόγω νομικών ανησυχιών, δυσχρηστίας και τεχνολογικών περιορισμών της εποχής.

Εν τέλει, οι υπερβολικές προσδοκίες για τα expert systems διαψεύστηκαν: η ανάπτυξη και συντήρηση των κανόνων ήταν εξαιρετικά δαπανηρή και χρονοβόρα, η κλιμάκωση (scaling) σε πιο περίπλοκα ή δυναμικά προβλήματα ήταν δύσκολη, και η απόδοσή τους μειωνόταν δραματικά εκτός του προκαθορισμένου πεδίου γνώσης. Αυτό οδήγησε σε έναν δεύτερο «χειμώνα» για την ΤΝ. **Στη δεκαετία του 1990, οι ερευνητές απέφευγαν συστηματικά τον όρο «Τεχνητή Νοημοσύνη», καθώς η αναφορά του θεωρούνταν ανασταλτικός παράγοντας για τη χρηματοδότηση ερευνητικών έργων [3]. Για τον λόγο αυτό, η έρευνα στον χώρο συνέχισε να αναπτύσσεται κυρίως υπό τον λιγότερο πομπώδη όρο Μηχανική Μάθηση (Machine Learning – ML), ο οποίος εστίαζε με μετρήσιμους όρους στον μηχανισμό με τον οποίο τα συστήματα αποκτούν γνώση. Ο όρος «Μηχανική Μάθηση» είχε εισαχθεί ήδη από το 1959 από τον Arthur L. Samuel, ο οποίος παρουσίασε ένα πρόγραμμα που μάθαινε να παίζει ντάμα (βλ. Σχήμα 4) [4].**

Συνεπώς, **η Μηχανική Μάθηση αποτελεί ένα από τα βασικά εργαλεία για την επίτευξη της Τεχνητής Νοημοσύνης.** Ως όρος, υιοθετήθηκε ευρέως τόσο για να αποστασιοποιηθεί η έρευνα από το στίγμα των πρώιμων αποτυχιών της Τεχνητής Νοημοσύνης, όσο και επειδή περιέγραφε πιο συγκεκριμένες και εφαρμόσιμες μεθόδους.

Η αναβίωση ήρθε σταδιακά μετά το 2000 (βλ. Σχήμα 3), και σε αυτό συνέβαλαν αρκετοί παράγοντες συνεργατικά:

- Η έλευση του διαδικτύου και η διαθεσιμότητα μεγάλης ποσότητας οργανωμένων δεδομένων (big data), όπως για παράδειγμα το ImageNet [5], μία βάση δεδομένων με εικόνες αντικειμένων με ετικέτα (labeled data), η οποία σήμερα περιλαμβάνει πάνω από 14 εκατομμύρια εγγραφές σε 20.000 κατηγορίες.
- Το γεγονός ότι είχαν ήδη βρεθεί αποτελεσματικοί αλγόριθμοι για την διαχείριση και εκπαίδευση βαθιών ΤΝΔ, ιδιαίτερα ο αλγόριθμος οπισθοδιάδοσης του σφάλματος

(backpropagation algorithm), ο οποίος προτάθηκε από τους Linnainmaa (1970) [6] και Werbos (1981) [7], ενώ καθιερώθηκε και υλοποιήθηκε αποτελεσματικά από τον G. L. Hinton (βραβείο Νόμπελ Φυσικής 2024 “*for foundational discoveries and inventions that enable machine learning with artificial neural networks*”) και συνεργάτες του (1986) [8]. Σημειώνεται ότι ο αλγόριθμος backpropagation είναι μια έξυπνη εφαρμογή του «κανόνα της αλυσίδας» του διαφορικού λογισμού.

- Η ανάπτυξη πιο σύγχρονων υπολογιστικών συστημάτων, με αυξημένες αποθηκευτικές και επεξεργαστικές ικανότητες (CPUs, GPUs, κατανεμημένοι υπολογισμοί, εξειδικευμένοι επεξεργαστές).

A. L. Samuel

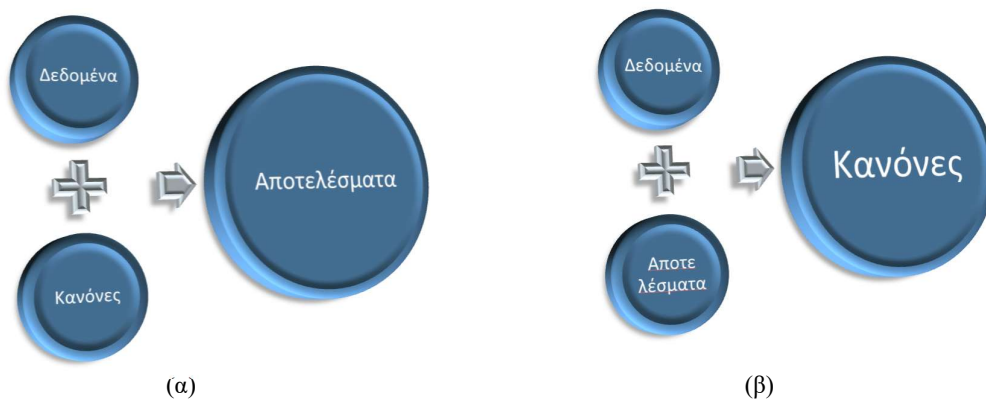
Some Studies in Machine Learning Using the Game of Checkers

Abstract: Two machine-learning procedures have been investigated in some detail using the game of checkers. Enough work has been done to verify the fact that a computer can be programmed so that it will learn to play a better game of checkers than can be played by the person who wrote the program. Furthermore, it can learn to do this in a remarkably short period of time (8 or 10 hours of machine-playing time) when given only the rules of the game, a sense of direction, and a redundant and incomplete list of parameters which are thought to have something to do with the game, but whose correct signs and relative weights are unknown and unspecified. The principles of machine learning verified by these experiments are, of course, applicable to many other situations.

Σχήμα 4: Η περίληψη του άρθρου του A. L. Samuel που εισήγαγε τον όρο της Μηχανικής Μάθησης το 1959 [4].

Εδώ αξίζει να σημειωθεί ότι ο όρος Μηχανική Μάθηση δεν αναφέρεται αποκλειστικά στα Τεχνητά Νευρωνικά Δίκτυα. Υπάρχει πλήθος μεθόδων που μπορούν να χρησιμοποιηθούν, αναλόγως του προβλήματος και των διαθέσιμων δεδομένων, από την απλή γραμμική παλινδρόμηση μίας ή πολλών μεταβλητών (univariate or multivariate linear regression) [9], μέχρι τις Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machines – SVM) [10], τα Δέντρα Απόφασης (Decision Trees) [11], τα Τυχαία Δάση (Random Forests) [12], τις Μεθόδους Ενίσχυσης (Boosting [13] και Bagging [14]), τους k-πλησιέστερους γείτονες (k-Nearest Neighbors – k-NN) [15] και τον αλγόριθμο K-means [16] για ομαδοποίηση δεδομένων (clustering). Επιπλέον των μεθόδων, διάφορες τεχνικές όπως η μείωση των διαστάσεων του προβλήματος (dimensionality reduction) με χρήση της Principal Component Analysis (PCA) [17], η κανονικοποίηση (regularization) [18] για την απλούστευση του μοντέλου μέσω της συνάρτησης κόστους (cost function), οι καμπύλες εκπαίδευσης (learning curves) [9] για την ανίχνευση φαινομένων underfitting ή overfitting κ.α. μπορούν να χρησιμοποιηθούν για την βελτιστοποίηση του τελικού συστήματος.

Σε κάθε περίπτωση, η προσέγγιση της Μηχανικής Μάθησης διαφοροποιείται ριζικά από τη συμβατική αλγοριθμική διαδικασία επίλυσης προβλημάτων, αφού οι κανόνες και τα αποτελέσματα έχουν αλλάξει αμοιβαία θέση (βλ. Σχήμα 5). Στη συμβατική αλγοριθμική προσέγγιση, τα δεδομένα (π.χ. αριθμός ορόφων, ποσοστό τοιχοπλήρωσης κ.α.) σε συνδυασμό με προκαθορισμένους κανόνες (διαμόρφωση μοντέλου, υπολογισμός μητρώου δυσκαμψίας κ.ά.) οδηγούν στο τελικό αποτέλεσμα (την εκτίμηση της θεμελιώδους ιδιοπεριόδου της κατασκευής). Αντίθετα, στη Μηχανική Μάθηση, τα δεδομένα σε συνδυασμό με τα τελικά αποτελέσματα από μεγάλο πλήθος ήδη ορθά επιλυμένων παραδειγμάτων χρησιμοποιούνται για την αυτόματη εξαγωγή των κανόνων που τα συνδέουν. Οι κανόνες αυτοί συνιστούν συνήθως ένα «μαύρο κουτί» (black box), καθώς δεν προκύπτουν άμεσα από τη φυσική ερμηνεία του προβλήματος το οποίο αντιπροσωπεύουν.



Σχήμα 5: (α) Συμβατική αλγοριθμική προσέγγιση και (β) προσέγγιση Μηχανικής Μάθησης.

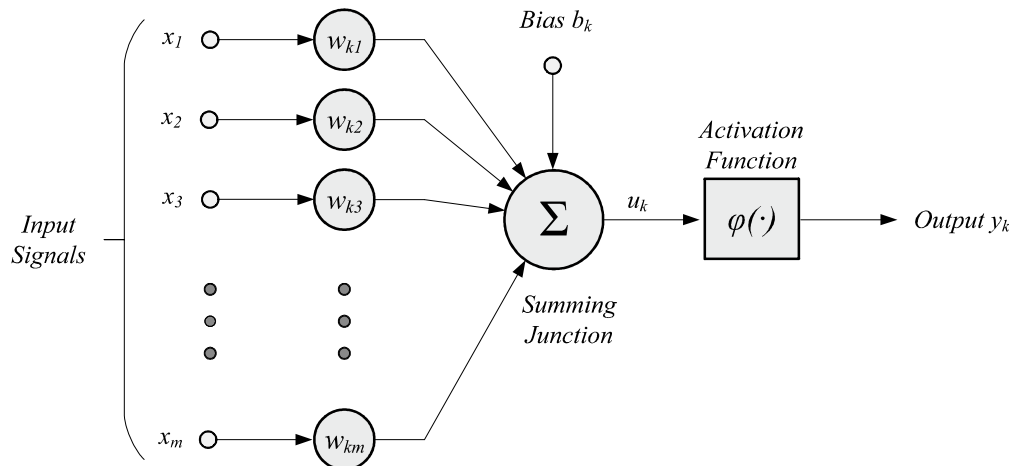
Γεγονός όμως είναι, ότι τα ΤΝΔ έχουν συγκεντρώσει το μεγαλύτερο ενδιαφέρον των ερευνητών. Τα ΤΝΔ είναι υπολογιστικά συστήματα που μιμούνται τα βιολογικά νευρωνικά δίκτυα που αποτελούν τον εγκέφαλο των ζώων. Μοιάζουν με τον εγκέφαλο από δύο απόψεις [19]:

1. Η γνώση αποκτάται από το δίκτυο από το περιβάλλον του μέσω μιας διαδικασίας μάθησης.
2. Η ισχύς των συνδέσεων μεταξύ των νευρώνων (neurons), γνωστή ως συναπτικό βάρος (synaptic weight), χρησιμοποιείται για την αποθήκευση της αποκτηθείσας γνώσης.

Τα βιολογικά νευρωνικά δίκτυα είναι ικανά να «μαθαίνουν» να εκτελούν εργασίες με βάση παραδείγματα, χωρίς να βασίζονται σε συγκεκριμένους κανόνες ή εξειδικευμένο προγραμματισμό. Για παράδειγμα, πειράματα «επανακαλωδίωσης» (“rewiring” experiments) σε χάμστερ έχουν δείξει ότι το τμήμα του εγκεφάλου που κανονικά είναι ο ακουστικός φλοιός μπορεί στην πραγματικότητα να μάθει να βλέπει, σχεδόν με τον ίδιο τρόπο που το κάνει ο πρωτογενής οπτικός φλοιός, όταν χειρουργικά δημιουργημένες (στο στάδιο της ανάπτυξης) νευρωνικές συνδέσεις στέλνουν εκεί τις πληροφορίες εισόδου οι οποίες προέρχονται από τον αμφιβληστροειδή. Αυτό υποστηρίζει την υπόθεση ότι δεν υπάρχει καμία εγγενώς εξειδικευμένη ιδιότητα του ακουστικού φλοιού εκτός από το ότι αυτό το συγκεκριμένο τμήμα του εγκεφάλου λαμβάνει πληροφορίες από τα αυτιά [20].

Ένα ΤΝΔ αποτελείται από έναν αριθμό τεχνητών νευρώνων, οι οποίοι συχνά αποκαλούνται κόμβοι (nodes), οργανωμένους σε διαδοχικές στρώσεις. Η στρώση εισόδου (input layer) αντιστοιχεί στα δεδομένα εισόδου του υπό εξέταση προβλήματος, ενώ η στρώση εξόδου (output layer) αντιστοιχεί στο αποτέλεσμα (ή τα αποτελέσματα) του προβλήματος. Ανάμεσα σε αυτές μπορεί να παρεμβάλλονται μία ή περισσότερες κρυφές στρώσεις (hidden layers), οι οποίες περιέχουν ενδιάμεσους κόμβους και προσδίδουν στο δίκτυο τη δυνατότητα να προσεγγίζει σύνθετες, έντονα μη γραμμικές σχέσεις.

Σε κάθε κόμβο, όλα τα σήματα εισόδου (x_i), πολλαπλασιασμένα με τα αντίστοιχα βάρη τους (w_{ki}), αθροίζονται μαζί με έναν όρο μετατόπισης (bias) του κόμβου (b_k). Το άθροισμα τροφοδοτείται σε μια συνάρτηση ενεργοποίησης (φ), π.χ. μια σιγμοειδή συνάρτηση, η οποία με τη σειρά της παράγει την έξοδο του κόμβου (y_k) (βλ. Σχήμα 6). Ένας αλγόριθμος οπισθοδιάδοσης του σφάλματος (backpropagation algorithm) προσαρμόζει συνεχώς τα βάρη και τους όρους μετατόπισης κάθε κόμβου κατά τη διάρκεια της εκπαίδευσης του δικτύου, σαρώνοντας το σύνολο δεδομένων εκπαίδευσης πολλές φορές. Κάθε πλήρης σάρωση ονομάζεται εποχή (epoch). Η εκπαίδευση συνεχίζεται έως ότου το δίκτυο φτάσει σε μια σταθερή κατάσταση όπου δεν υπάρχουν περαιτέρω σημαντικές αλλαγές στα συναπτικά βάρη και τους όρους μετατόπισης. Για περισσότερες πληροφορίες, ο αναγνώστης μπορεί να ανατρέξει σε σχετικές πηγές που είναι άφθονες στη βιβλιογραφία.



Σχήμα 6: Μη γραμμικό προσομοίωμα ενός τεχνητού νευρώνα (artificial neuron) [19].

2 ΣΥΓΧΡΟΝΕΣ ΕΞΕΛΙΞΕΙΣ

Η πιο πρόσφατη επανάσταση στην ΤΝ προήλθε από την παρουσίαση της αρχιτεκτονικής Transformer στα ΤΝΔ με το άρθρο “Attention is all you need” (2017) [21] από ερευνητές της Google (βλ. Σχήμα 4).

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com	Noam Shazeer* Google Brain noam@google.com	Niki Parmar* Google Research nikip@google.com	Jakob Uszkoreit* Google Research usz@google.com
Llion Jones* Google Research llion@google.com	Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu	Lukasz Kaiser* Google Brain lukaszkaizer@google.com	
Illia Polosukhin* ‡ illia.polosukhin@gmail.com			

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

*Equal contribution. Listing order is random. Jakob proposed replacing RNNs with self-attention and started the effort to evaluate this idea. Ashish, with Illia, designed and implemented the first Transformer models and has been crucially involved in every aspect of this work. Noam proposed scaled dot-product attention, multi-head attention and the parameter-free position representation and became the other person involved in nearly every detail. Niki designed, implemented, tuned and evaluated countless model variants in our original codebase and tensor2tensor. Llion also experimented with novel model variants, was responsible for our initial codebase, and efficient inference and visualizations. Lukasz and Aidan spent countless long days designing various parts of and implementing tensor2tensor, replacing our earlier codebase, greatly improving results and massively accelerating our research.

†Work performed while at Google Brain.

‡Work performed while at Google Research.

Σχήμα 7: Η πρώτη σελίδα του άρθρου με τίτλο “Attention is all you need” (v.7) [21]. Η σειρά των συγγραφέων είναι τυχαία.

Η καινοτομία της αρχιτεκτονικής αυτής ήταν ότι εγκατέλειψε τις παλιές προσεγγίσεις που βασίζονταν σε Επαναληπτικά Νευρωνικά Δίκτυα (Recurrent Neural Networks – RNNs) ή Συνελκτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks – CNNs) για επεξεργασία ακολουθιών, και χρησιμοποίησε αποκλειστικά έναν μηχανισμό που ονομάζεται *self-attention*.

Με το *self-attention*, το μοντέλο μπορεί να «προσέχει» (δηλαδή να δίνει διαφορετικό βάρος) σε πολλαπλά σημεία της εισόδου ταυτόχρονα, επιτρέποντας:

- Παράλληλη επεξεργασία δεδομένων (άρα ταχύτερη εκπαίδευση).
- Καλύτερη κατανόηση μακρινών σχέσεων μέσα σε κείμενο ή μέσα σε άλλες ακολουθίες (άρα καλύτερη αντίληψη μεγάλων κειμένων).

Αποτέλεσμα ήταν η δραματική βελτίωση των παραπάνω σημείων σε σχέση με τα RNNs και τα CNNs. Αυτή η ιδέα έγινε η βάση για σχεδόν όλα τα σύγχρονα LLMs, όπως τα BERT, GPT, Llama, Claude, Grok, DeepSeek, Kimi κ.α., με δισεκατομμύρια παραμέτρους, καθιστώντας δυνατή τη δημιουργία συστημάτων με εντυπωσιακές δυνατότητες επικοινωνίας, κατανόησης και δημιουργίας περιεχομένου. Ειδικά η κυκλοφορία του ChatGPT 3.5 στις 30/11/2022 ανέδειξε στο ευρύ κοινό την τεράστια πρόοδο που είχε γίνει στον τομέα.

Τα σύγχρονα LLMs έχουν εξελιχθεί σε πολυτροπικά (multimodal) μοντέλα, ικανά να επεξεργάζονται και να παράγουν όχι μόνο κείμενο, αλλά και εικόνες, ήχο και βίντεο. Είναι ικανά να λειτουργούν σε πραγματικό χρόνο και να προσφέρουν αλληλεπίδραση σε πολλές γλώσσες και μορφές. Μπορούν να παράγουν απολύτως ρεαλιστικά βίντεο, όπως για παράδειγμα αυτοκίνητα που κινούνται γρήγορα σε πλημμυρισμένους δρόμους, χωρίς να λύνουν πολύπλοκες εξισώσεις Computational Fluid Dynamics (CFD) ή να γνωρίζουν οτιδήποτε για Fluid-Structure Interaction (FSI). Πέρα από τέτοιου τύπου εφαρμογές, κάποια μοντέλα έχουν ήδη ή θα έχουν τεράστια επίπτωση στην ανθρώπινη ζωή και εμπειρία. Για παράδειγμα, το AlphaFold 3 (το οποίο προβλέπει όχι μόνο τις τρισδιάστατες δομές πρωτεϊνών, αλλά και τις αλληλεπιδράσεις μεταξύ μορίων) έχει φέρει επανάσταση στην ανάπτυξη νέων φαρμάκων, η οποία θα γίνει αισθητή τα επόμενα χρόνια. Για την δημιουργία του μοντέλου αυτού τιμήθηκε ο Demis Hassabis με το Νόμπελ Χημείας το 2024, ως συνιδρυτής και διευθύνων σύμβουλος της Google DeepMind.

Τονίζεται ότι, παρόλο που ενδεχομένως πολλοί έχουν την εντύπωση ότι τα μοντέλα μπορούν να λειτουργούν μόνο εντός των ορίων της ήδη αποκτηθείσας γνώσης που έχουν κατά την εκπαίδευσή τους, πολλά από αυτά έχουν ήδη αποδείξει ότι μπορούν να κάνουν δημιουργικές και απρόβλεπτες επιλογές, οι οποίες ξεπερνούν την ανθρώπινη διαίσθηση ακόμη και των πλέον ειδικών. Πλέον υπάρχει έκφραση για αυτό ακριβώς: ονομάζεται «κίνηση 37» (move 37), και αναφέρεται στην 37^η κίνηση που έκανε το πρόγραμμα AlphaGo στον δεύτερο αγώνα του ιστορικού ματς εναντίον του Lee Sedol τον Μάρτιο του 2016. Η κίνηση ξεχώρισε γιατί ήταν εντελώς απρόσμενη ακόμη και για επαγγελματίες παίκτες του Go.

3 ΜΕΛΛΟΝΤΙΚΕΣ ΚΑΤΕΥΘΥΝΣΕΙΣ

Ο όρος Μεγάλο Γλωσσικό Μοντέλο (LLM) πλέον τείνει να μην χρησιμοποιείται. Αντί αυτού χρησιμοποιείται ο πιο γενικός όρος Μοντέλο Θεμελίωσης (Foundation Model), επειδή τα μεγάλα μοντέλα πλέον δεν είναι απλώς “γλωσσικά” αλλά αποτελούν γενικές βάσεις πάνω στις οποίες μπορούν να θεμελιωθούν πολλά διαφορετικά συστήματα και εφαρμογές, τα οποία διαχειρίζονται εικόνα, ήχο, βίντεο κ.λπ. Τα κυριότερα προβλήματα που αντιμετωπίζουν και χρήζουν βελτίωσης έχουν να κάνουν με τον σχεδιασμό λογικών βημάτων σε σύνθετα προβλήματα, τις παραισθήσεις (hallucinations, δηλαδή την ιδιαίτερα συχνή κατάσταση που φαντάζονται πληροφορίες οι οποίες δεν υπάρχουν στην πραγματικότητα), την τρισδιάστατη αντίληψη του κόσμου και την αριθμητική ακρίβεια σε υπολογισμούς.

Για το μέλλον, έχουν διαμορφωθεί δύο βασικές κατευθύνσεις ανάπτυξης:

- Non-agentic workflow: τα μοντέλα απαντούν άμεσα σε εντολές με μία ενιαία έξοδο, χωρίς εσωτερικό σχεδιασμό ή κλήση εξωτερικών εργαλείων (π.χ. παραγωγή κειμένου ή απλή μετάφραση σε ένα βήμα).
- Agentic workflow: το μοντέλο (ή συνεργαζόμενα μοντέλα) σπάνε το πρόβλημα σε στάδια, ακολουθούν διαδοχικά βήματα, και ενδεχομένως χρησιμοποιούν εξωτερικά εργαλεία (tool use) όπως APIs, βάσεις δεδομένων, ή εξειδικευμένα συστήματα (π.χ. Wolfram Alpha, OpenWeatherMap, GitHub).

Τα περισσότερα νέα μοντέλα ακολουθούν πλέον το agentic workflow, καθώς αυτό επιτρέπει την επίλυση πιο σύνθετων προβλημάτων.

Ενώ η Στενή Τεχνητή Νοημοσύνη (Narrow AI ή Artificial Narrow Intelligence – ANI) έχει ήδη επιτευχθεί (για παράδειγμα, εξειδικευμένα chatbots σε ιστοτόπους, μοντέλα αναγνώρισης εικόνας από ακτινογραφίες, συστήματα μετάφρασης κειμένου κ.λπ.), επόμενος στόχος θεωρείται η Τεχνητή Γενική Νοημοσύνη (General AI ή Artificial General Intelligence – AGI), δηλαδή το θεωρητικό επίπεδο τεχνητής νοημοσύνης που μπορεί να κατανοεί, να μαθαίνει και να εφαρμόζει γνώσεις σε οποιοδήποτε πεδίο, όπως ένας άνθρωπος. Ο απώτερος στόχος όμως είναι πλέον η Τεχνητή Υπερνοημοσύνη (Artificial Super-Intelligence – ASI), το ιερό δισκοπότηρο, η οποία ξεπερνά κατά πολύ την ανθρώπινη νοημοσύνη σε όλους τους τομείς, συμπεριλαμβανομένης της δημιουργικότητας, της επίλυσης προβλημάτων και της επιστημονικής καινοτομίας. Τη στιγμή συγγραφής του παρόντος άρθρου, η επίτευξη του AGI θεωρείται ότι θα συμβεί οσονούπω, ενώ το κυνήγι για την ASI έχει οδηγήσει σε μία ανεπανάληπτη κούρσα εξοπλισμών μεταξύ κρατών και εταιρειών-κολοσσών, με τεράστια ποσά να δαπανώνται προς πάσα κατεύθυνση (βλ. Σχήμα 3).

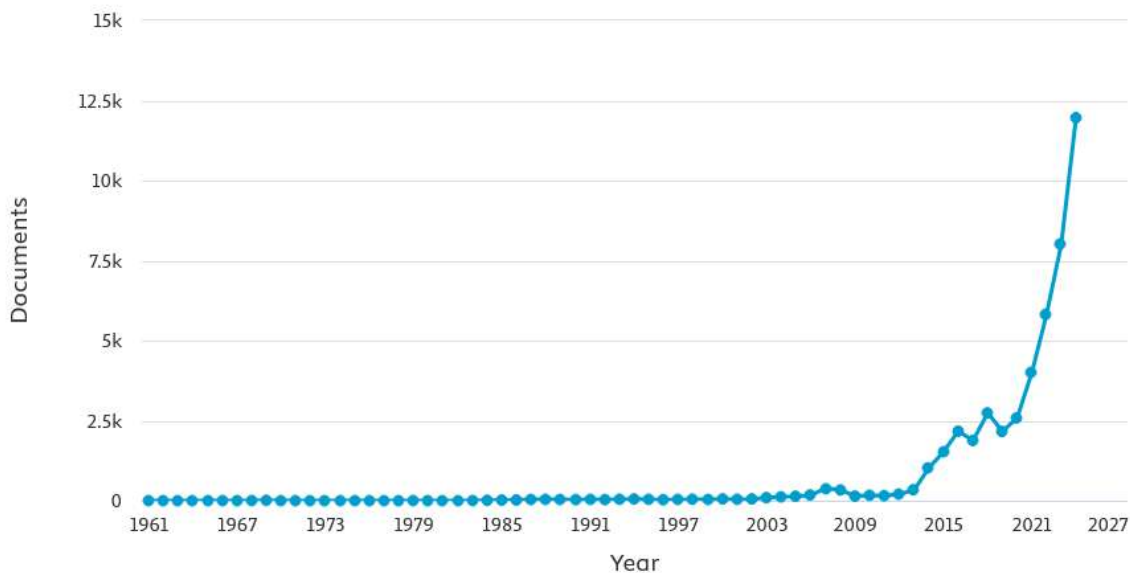
4 ΕΦΑΡΜΟΓΕΣ ΣΤΗΝ ΕΠΙΣΤΗΜΗ ΤΟΥ ΜΗΧΑΝΙΚΟΥ

Η επιστήμη του Μηχανικού δεν θα μπορούσε να παραμείνει ανεπηρέαστη από αυτές τις τεχνολογικές εξελίξεις. Τα τελευταία χρόνια, παρατηρείται μια ραγδαία και συστηματική ενσωμάτωση μεθόδων Μηχανικής Μάθησης και Τεχνητής Νοημοσύνης στην ανάλυση, τον σχεδιασμό και την πρόβλεψη πολύπλοκων φαινομένων, γεγονός που αποτυπώνεται στον ολοένα αυξανόμενο αριθμό σχετικών επιστημονικών δημοσιεύσεων. Παραδείγματα περιλαμβάνουν την αναγνώριση φθορών σε κατασκευές μέσω εικόνων [22], την πρόβλεψη σεισμικής απόκρισης [23], τον υπολογισμό του οπλισμού σε στοιχεία οπλισμένου σκυροδέματος [24], τον υπολογισμό της σκληρότητας κεραμικών μικρο-προβόλων [25], την επίλυση προβλημάτων μεταφοράς θερμότητας [26], την προσομοίωση ενδομυϊκών βλαβών [27] και πάρα πολλά άλλα. Μία απλή αναζήτηση στο scopus.com επαληθεύει ότι ο ρυθμός δημοσίευσης σχετικών άρθρων στην επιστήμη του Μηχανικού αυξάνεται ραγδαία (βλ. Σχήμα 8), ενώ στο σύνολο των σχετικών δημοσιευμένων άρθρων, μόνο 1 στα 6 περίπου άρθρα ανήκουν στην επιστήμη του Μηχανικού, (όπως αυτή ορίζεται από το scopus) (βλ. Σχήμα 9).

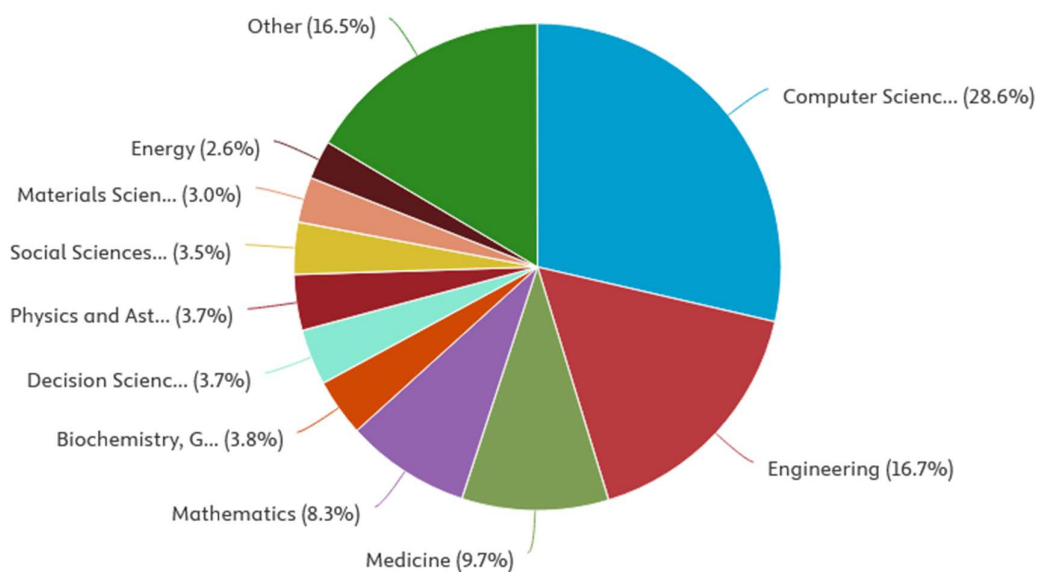
Όσον αφορά τις εφαρμογές, αυτές μπορούν πλέον να διακριθούν σε δύο βασικές κατηγορίες:

- Εφαρμογές βασιζόμενες σε Foundation Models, δηλαδή μεγάλα ή μικρότερα προ-εκπαιδευμένα μοντέλα που μπορούν να «προσαρμοστούν λεπτομερώς» (δηλαδή να γίνουν fine-tuned) ώστε να εκτελούν με ακρίβεια συγκεκριμένες εργασίες.

- Εφαρμογές Μηχανικής Μάθησης απευθείας σε πρωτογενή δεδομένα, οι οποίες οδηγούν σε εξειδικευμένα και προσεκτικά ελεγμένα συστήματα. Αυτά τα συστήματα μπορούν να ενσωματωθούν, για παράδειγμα μέσω API, σε μεγαλύτερες λύσεις της πρώτης κατηγορίας οπότε να προκύψουν υβριδικά συστήματα με καλύτερη απόδοση.



Σχήμα 8: Ρυθμός δημοσίευσης άρθρων (άρθρα/έτος) σχετικών με Μηχανική Μάθηση και Τεχνητή Νοημοσύνη στην επιστήμη του Μηχανικού (αναζήτηση με όρους “Machine Learning” και “Artificial Intelligence” στα πεδία «Τίτλος», «Περίληψη» και «Λέξεις-κλειδιά», με περιορισμό των πηγών σε “Engineering”, από το scopus την 13/08/2025).



Σχήμα 9: Κατανομή συνόλου δημοσιευμένων άρθρων σχετικών με Μηχανική Μάθηση και Τεχνητή Νοημοσύνη (αναζήτηση με όρους “Machine Learning” και “Artificial Intelligence” στα πεδία «Τίτλος», «Περίληψη» και «Λέξεις-κλειδιά», από το scopus την 13/08/2025).

Σημαντικό είναι να σημειωθεί ότι η εκπαίδευση ενός Foundation Model από το μηδέν είναι αδύνατη σε προσωπικούς υπολογιστές. Απαιτούνται τεράστιοι υπολογιστικοί πόροι, οι οποίοι μπορούν να διατεθούν μόνο από μεγάλα ιδρύματα ή εταιρείες. Για παράδειγμα, το κόστος για την εκπαίδευση του (πεπαλαιωμένου πλέον) GPT-4 εκτιμάται σε περισσότερο από 100 εκατομμύρια δολάρια. Ακόμη και ο απλός συμπερασμός (inference), δηλαδή η απλή χρήση ήδη εκπαιδευμένων, μικρότερων ανοιχτών μοντέλων με λίγα δισεκατομμύρια παραμέτρους παραμένει ιδιαίτερα απαιτητική, ακόμη και για πολύ ισχυρά συστήματα προσωπικών υπολογιστών. Οι περιορισμοί προκύπτουν είτε από την ανεπαρκή μνήμη, είτε – ακόμη κι όταν η μνήμη επαρκεί – από την μεγάλη κατανάλωση ενέργειας και τον χαμηλό ρυθμό παραγωγής tokens, που καθιστά την εμπειρία μη χρηστική. Τη στιγμή συγγραφής αυτού του άρθρου, ένα κορυφαίο σύστημα με GPU που διαθέτει 24 GB VRAM μπορεί να εκτελεί αποδοτικά inference σε μοντέλα περίπου 13 δισ. παραμέτρων με χρήση κβαντοποίησης (quantization) 8-bit, η οποία μειώνει σημαντικά τις απαιτήσεις μνήμης, μετατρέποντας τα βάρη των μοντέλων από 32-bit σε 8-bit. Το πλήρες fine-tuning τέτοιων μοντέλων παραμένει μη ρεαλιστικό όταν υπερβαίνουν τις 7 δισ. παραμέτρους. Ωστόσο, με την τεχνική QLoRA (Quantization 4-bit + Low-Rank Adaptation), όπου η εκπαίδευση εφαρμόζεται μόνο σε επιλεγμένα layers μέσω LoRA adapters, καθίσταται εφικτό το fine-tuning ακόμη και σε μοντέλα των 13 δισ. παραμέτρων. Τα κορυφαία μοντέλα αυτή τη στιγμή διαθέτουν όμως πλέον του 1 τρις. παραμέτρους. Ρεαλιστικά λοιπόν, η χρήση αυτών των μοντέλων έχει νόημα μόνο σε εξειδικευμένους servers με υπηρεσίες cloud, είτε μέσω των εταιρειών που διαθέτουν τα μοντέλα κλειστού κώδικα, είτε μέσω τρίτων εταιρειών που εξυπηρετούν ανοιχτά μοντέλα.

Όσον αφορά τις εφαρμογές Μηχανικής Μάθησης της δεύτερης κατηγορίας, αυτές έχουν ιδιαίτερο ενδιαφέρον διότι το πρόβλημα είναι σαφώς πιο διαχειρίσιμο και οριοθετημένο, ενώ το τελικό μοντέλο μπορεί να ελεγχθεί διεξοδικά. Ωστόσο, αυτή η προσέγγιση δεν αποτελεί πανάκεια. Για να είναι επιτυχής, απαιτούνται δύο βασικές προϋποθέσεις:

- **Η χρήση μεθόδων Μηχανικής Μάθησης να προσφέρει σαφή πλεονεκτήματα σε απόδοση, ακρίβεια, σταθερότητα ή ικανότητα γενίκευσης, σε σχέση με τις καθιερωμένες μεθόδους.** Για παράδειγμα, δεν έχει νόημα η εκπαίδευση νευρωνικών δικτύων για την εκτίμηση της παραμόρφωσης σε μικρά δικτυώματα, όταν με εφαρμογή της μητρικής στατικής μπορεί να προκύψει πολύ πιο γρήγορα το «ακριβές» αποτέλεσμα.
- **Να υπάρχουν επαρκή δεδομένα για την εκπαίδευση των μοντέλων και την εκτίμηση της ποιότητάς τους.** Για παράδειγμα, τα TNA, τα οποία αποτελούν την πλέον διαδεδομένη μέθοδο Μηχανικής Μάθησης, είναι πολύ πιθανόν να «υπερ-εκπαιδευτούν», δηλαδή να κάνουν overfit, όταν τα δεδομένα δεν είναι επαρκή. Ένα τέτοιο σύστημα δεν έχει καλές δυνατότητες γενίκευσης.

Σχετικά με την δεύτερη προϋπόθεση, αυτή της επάρκειας δεδομένων, οι ίδιοι οι φυσικοί νόμοι παρέχουν πολύτιμη πληροφορία η οποία μπορεί να ενσωματωθεί απευθείας μέσα στο πλαίσιο της εκπαίδευσης των TNA. Για το σκοπό αυτό, προτάθηκαν σχετικά πρόσφατα τα Physics-Informed Neural Networks (PINNs) [28,29], τα οποία άνοιξαν νέους δρόμους στην επίλυση προβλημάτων που διέπονται από φυσικούς νόμους, μειώνοντας σημαντικά την ανάγκη συλλογής μεγάλου όγκου δεδομένων και παράγοντας καλύτερα τελικά συστήματα. Σε αυτά, η συνάρτηση κόστους (loss function), η οποία χρησιμοποιείται στην εκπαίδευση του TNA, δεν περιλαμβάνει

μόνο τις διαφορές μεταξύ προβλέψεων και μετρήσεων, αλλά και όρους που εκφράζουν το σφάλμα ως προς την ικανοποίηση των μερικών διαφορικών εξισώσεων που γνωρίζουμε ότι (πρέπει να) διέπουν το πρόβλημα.

5 ΕΦΑΡΜΟΓΗ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ ΓΙΑ ΤΗΝ ΕΚΤΙΜΗΣΗ ΤΗΣ ΘΕΜΕΛΙΩΔΟΥΣ ΙΔΙΟΠΕΡΙΟΔΟΥ ΚΤΙΡΙΩΝ

Η θεμελιώδης ιδιοπερίοδος είναι μια σημαντική παράμετρος για τον αντισεισμικό σχεδιασμό, ειδικά για τις κατασκευές από οπλισμένο σκυρόδεμα που γενικά συνδέονται με μεγάλη μάζα. Ωστόσο, η ακριβής εκτίμησή της δεν είναι εύκολη, καθώς εξαρτάται από διάφορους παράγοντες, όπως το ύψος της κατασκευής, την παρουσία (ή απουσία) τοίχων πλήρωσης, την ακαμψία αυτών των τοίχων, το μήκος των ανοιγμάτων κ.λπ. [30,31]. Οι κανονισμοί βασίζονται σε απλοποιημένους εμπειρικούς τύπους για την εκτίμηση της θεμελιώδους περιόδου των κατασκευών. Μια δημοφιλής κατηγορία αυτών των τύπων έχει τη μορφή $T \cong cH^a$ όπου, H είναι ένα μέτρο της κατακόρυφης διάστασης της κατασκευής (πραγματικό ύψος ή αριθμός ορόφων) ενώ a και c είναι κατάλληλες σταθερές. Επομένως, αναγνωρίζεται εξ αρχής ότι το ύψος είναι ο πιο κρίσιμος παράγοντας. Για παράδειγμα, η σχέση $T \cong C_t H^{3/4}$ εισήχθη για πρώτη φορά στο ATC3-06 (1978) για κατασκευές με σκελετό από οπλισμένο σκυρόδεμα. Η τιμή $C_t = 0,075$ προέκυψε με βάση τα πειραματικά δεδομένα που συλλέχθηκαν από τον σεισμό του San Fernando το 1971. Η ίδια έκφραση έχει υιοθετηθεί από τον Ευρωκώδικα 8 (2004) και τον Uniform Building Code (1997). Άλλοι κανονισμοί χρησιμοποιούν διαφορετικές τιμές για το C_t ανάλογα με τον τύπο της κατασκευής (οπλισμένο σκυρόδεμα, χάλυβας ή άλλος) και την παρουσία (ή απουσία) τοίχων πλήρωσης. Χρησιμοποιούνται επίσης πιο απλοϊκές σχέσεις (όπως η σχέση $T = 0,1N$, όπου N είναι ο αριθμός των ορόφων, στον Καναδικό κανονισμό του 1995) αλλά και πιο περίπλοκες εκφράσεις. Μια πιο πλήρης λίστα μπορεί να αναζητηθεί στις αναφορές [30,31].

Είναι εύκολο να δείχθεί ότι, γενικά, αυτές οι εκφράσεις παρουσιάζουν πολύ διαφορετικές εκτιμήσεις της θεμελιώδους ιδιοπεριόδου για την ίδια κατασκευή. Αυτό μπορεί να αποδοθεί σε δύο κυρίως λόγους: (α) ορισμένες από τις εκφράσεις είναι υπερβολικά απλοϊκές και αγνοούν σημαντικούς παράγοντες και (β) τα δεδομένα που χρησιμοποιούνται για την παραγωγή των σχέσεων είναι περιορισμένα.

Για την εφαρμογή μεθόδων Μηχανικής Μάθησης στο πρόβλημα αυτό, η έλλειψη δεδομένων καλύφθηκε σε κάποιο βαθμό με την δημοσιοποίηση της βάσης δεδομένων FP4026 [32], η οποία περιλαμβάνει δεδομένα για 4026 κτίρια από οπλισμένο σκυρόδεμα με τοιχοπληρώσεις, τα οποία έχουν σχεδιαστεί σύμφωνα με τον Ευρωκώδικα 2 [33] και τον Ευρωκώδικα 8 [34]. Οι τοιχοπληρώσεις έχουν προσομοιωθεί χρησιμοποιώντας ένα ισοδύναμο μη γραμμικό μοντέλο που προτάθηκε από τον Crisafulli [35]. Η βάση δεδομένων [32] περιλαμβάνει κατασκευές με διάφορες διαμορφώσεις, οι οποίες εξαρτώνται από πέντε παραμέτρους. Συγκεκριμένα,

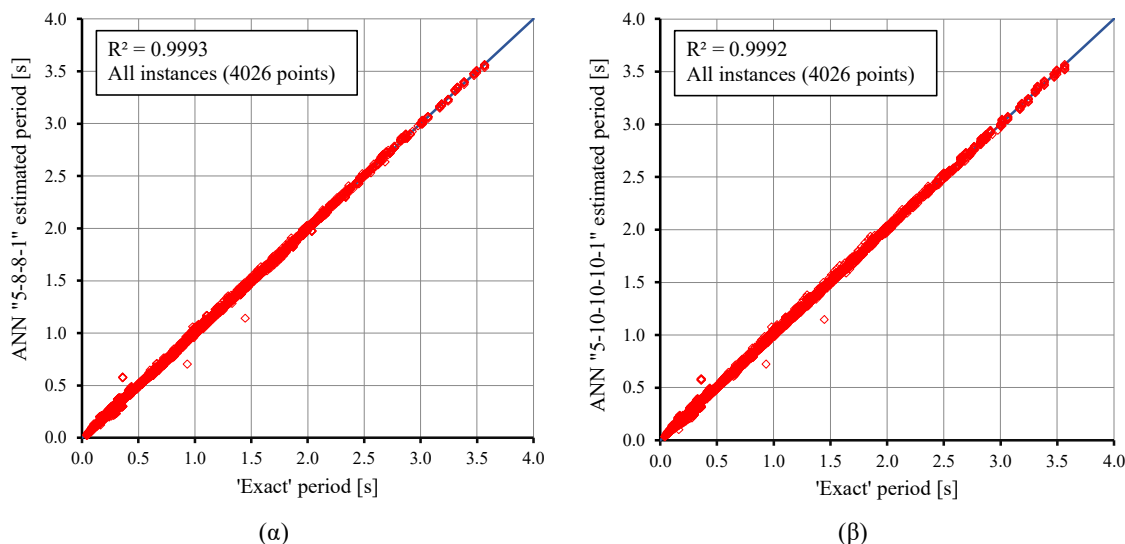
1. ο αριθμός των ορόφων κυμαίνεται από 1 έως 22 ορόφους, με βήμα ίσο με τη μονάδα,
2. ο αριθμός των ανοιγμάτων έχει τρεις διαφορετικές τιμές (2, 4, 6),
3. το μήκος των ανοιγμάτων έχει τέσσερις διαφορετικές τιμές (3,0, 4,5, 6,0 και 7,5 m),
4. το ποσοστό ανοίγματος παίρνει πέντε διαφορετικές τιμές (0%, 25%, 50%, 75% και 100%), με το 0% να αντιστοιχεί σε πλήρως τοιχοπληρωμένα ανοίγματα και το 100% σε γυμνά πλαίσια, και, τέλος,

5. η αξονική στιβαρότητα του τοίχου παίρνει επτά διαφορετικές τιμές (2,25, 4,5, 7,5, 11,25, 15,0, 20,0 και 25,0 x 10⁵ kN/m).

Μετά την δημοσίευση της βάσης [32], αυτή χρησιμοποιήθηκε σε πλήθος άρθρων για την εκτίμηση της θεμελιώδους ιδιοπεριόδου πολυώροφων κτιρίων (βλ. ενδεικτικά [36–41]). Στο συγκεκριμένο πρόβλημα, η χρήση ΤΝΔ μπορεί να αποδώσει συντελεστή προσδιορισμού (coefficient of determination) R^2 που υπερβαίνει το 0.999, όπως έχει δειχθεί στη βιβλιογραφία. Δεδομένου ότι έχει επιτευχθεί ένα τέτοιο επίπεδο ακρίβειας, η περαιτέρω αύξησή του μέσω ολοένα και πιο πολύπλοκων μοντέλων καθίσταται πρακτικά άνευ ουσίας.

Έτσι, στην παρούσα εργασία, παρουσιάζεται ένα απλό ΤΝΔ με αρχιτεκτονική 5-8-8-1, το οποίο συνδυάζει απλότητα και εξαιρετική απόδοση στο συγκεκριμένο πρόβλημα. Η αρχιτεκτονική 5-8-8-1 αναλύεται ως εξής:

- Στρώση εισόδου (input layer) με πέντε νευρώνες (neurons) ή αλλιώς κόμβους (nodes), όσες και οι παράμετροι της βάσης [32].
- Πρώτη κρυφή στρώση (hidden layer) με οκτώ κόμβους.
- Δεύτερη κρυφή στρώση με οκτώ κόμβους.
- Στρώση εξόδου (output layer) με έναν μόνο κόμβο, ο οποίος αντιστοιχεί στην εκτιμώμενη θεμελιώδη ιδιοπερίοδο της κατασκευής.



Σχήμα 10: (α) Απόδοση του προτεινόμενου ΤΝΔ με αρχιτεκτονική 5-8-8-1 στην εκτίμηση της θεμελιώδους ιδιοπεριόδου με χρήση της βάσης δεδομένων [32] και του PyTorch [42] και (β) απόδοση στο ίδιο πρόβλημα από ΤΝΔ με αρχιτεκτονική 5-10-10-10-1 [36] και χρήση του WEKA [43].

Η εκπαίδευσή του προτεινόμενου ΤΝΔ έγινε με το λογισμικό PyTorch [42]. Συγκρινόμενο με ένα σαφώς μεγαλύτερο ΤΝΔ με αρχιτεκτονική 5-10-10-10-1 [36], το οποίο εκπαιδεύτηκε με χρήση του λογισμικού WEKA [43], το προτεινόμενο ΤΝΔ πετυχαίνει ελαφρώς καλύτερη τιμή του R^2 , όπως φαίνεται στο Σχήμα 10, έχοντας σχεδόν τους μισούς κόμβους στις ενδιάμεσες κρυφές στρώσεις.

Για την αξιοποίηση του εκπαιδευμένου μοντέλου σε ένα ευρύτερο πλαίσιο εφαρμογών, είναι απαραίτητη η διάθεση τόσο της αρχιτεκτονικής όσο και των βαρών του. Αυτό μπορεί να πραγματοποιηθεί με διάφορους τρόπους, ανάλογα με την πλατφόρμα χρήσης (π.χ. αρχεία .onnx για φορητότητα, .pt/.pth για PyTorch, .h5 για TensorFlow κ.ά.), απαιτώντας τη χρήση του αντίστοιχου εξειδικευμένου λογισμικού. Στην παρούσα εργασία ακολουθήθηκε μια εναλλακτική προσέγγιση: αναπτύχθηκε κώδικας σε Python, ο οποίος παράγει αυτόματα βελτιστοποιημένο κώδικα Python για την υλοποίηση της προωθητικής διάδοσης (forward pass) του εκπαιδευμένου ΤΝΔ. Ο παραγόμενος κώδικας μπορεί εύκολα να μετασχηματιστεί σε οποιοδήποτε επιθυμητή μορφή και να χρησιμοποιηθεί για συμπερασμό (inference) χωρίς εξειδικευμένο λογισμικό. Ενδεικτικά, στο Σχήμα 11 παρουσιάζεται η εκδοχή του σε VBA (Visual Basic for Applications) για χρήση απευθείας στο Microsoft Excel.

```
Public Function Period5881(nsto As Double, nspan As Double, sl As Double, oratio As Double, ws As Double)
    As Double

    'variable      range used  units
    '              in training
    nsto_n = (nsto - 1) / 21    'number of stories 1 .. 22    (-)
    nspan_n = (nspan - 2) / 4   'number of spans   2 .. 6      (-)
    sl_n = (sl - 3#) / 4.5      'span length       3 .. 7.5    (m)
    oratio_n = oratio / 1#      'opening ratio      0 .. 1      (-)
    ws_n = (ws - 2.25) / 22.75 'wall stiffness     2.25 .. 25   (x 10^5 kN/m)

    n_1_1 = 1 / (1 + Exp(-(2.001794 * nsto_n - 0.133324 * nspan_n + 0.15308 * sl_n - 1.776044 * oratio_n
    + 0.099713 * ws_n + 0.132783)))
    n_1_2 = 1 / (1 + Exp(-(-0.845942 * nsto_n + 0.16936 * nspan_n + 2.120253 * sl_n - 2.613696 * oratio_n
    + 0.336213 * ws_n + 1.633914)))
    n_1_3 = 1 / (1 + Exp(-(0.366092 * nsto_n + 0.263646 * nspan_n + 0.023239 * sl_n + 4.401766 * oratio_n
    - 0.508016 * ws_n - 1.244036)))
    n_1_4 = 1 / (1 + Exp(-(-1.475332 * nsto_n + 2.169023 * nspan_n + 2.851159 * sl_n - 1.22201 * oratio_n
    + 0.008523 * ws_n + 4.524468)))
    n_1_5 = 1 / (1 + Exp(-(-0.327358 * nsto_n + 0.191917 * nspan_n + 0.001362 * sl_n + 3.875104 * oratio_n
    + 6.403768 * ws_n + 1.614252)))
    n_1_6 = 1 / (1 + Exp(-(0.311194 * nsto_n - 0.056633 * nspan_n + 0.408408 * sl_n + 2.588574 * oratio_n
    - 1.435539 * ws_n - 1.282778)))
    n_1_7 = 1 / (1 + Exp(-(-4.090371 * nsto_n - 0.077611 * nspan_n - 0.300995 * sl_n + 3.390639 * oratio_n
    - 0.056882 * ws_n - 2.372193)))
    n_1_8 = 1 / (1 + Exp(-(1.131201 * nsto_n + 0.129222 * nspan_n - 0.10719 * sl_n + 8.388197 * oratio_n
    + 0.078591 * ws_n - 6.380926)))

    n_2_1 = 1 / (1 + Exp(-(3.11317 * n_1_1 + 1.894734 * n_1_2 + 1.900744 * n_1_3 - 4.257499 * n_1_4 -
    3.066481 * n_1_5 + 0.567246 * n_1_6 - 1.710043 * n_1_7 + 2.439602 * n_1_8 - 2.236251)))
    n_2_2 = 1 / (1 + Exp(-(0.266049 * n_1_1 - 4.193397 * n_1_2 + 1.466526 * n_1_3 - 1.230494 * n_1_4 -
    1.478064 * n_1_5 + 0.317588 * n_1_6 - 0.878624 * n_1_7 - 1.160025 * n_1_8 - 0.369043)))
    n_2_3 = 1 / (1 + Exp(-(1.906023 * n_1_1 + 0.96787 * n_1_2 + 1.862416 * n_1_3 - 3.691494 * n_1_4 -
    2.395166 * n_1_5 + 0.012186 * n_1_6 - 2.031888 * n_1_7 + 1.828438 * n_1_8 - 1.920391)))
    n_2_4 = 1 / (1 + Exp(-(2.923381 * n_1_1 - 3.617184 * n_1_2 + 2.7317 * n_1_3 - 1.479763 * n_1_4 -
    0.988765 * n_1_5 - 1.604322 * n_1_6 - 4.209823 * n_1_7 - 2.607512 * n_1_8 - 1.028801)))
    n_2_5 = 1 / (1 + Exp(-(-0.409394 * n_1_1 + 0.727772 * n_1_2 + 2.677315 * n_1_3 - 2.114131 * n_1_4 -
    1.389831 * n_1_5 - 1.831689 * n_1_6 - 1.121321 * n_1_7 - 0.980444 * n_1_8 - 1.255347)))
    n_2_6 = 1 / (1 + Exp(-(1.126737 * n_1_1 - 3.131869 * n_1_2 + 0.779841 * n_1_3 + 1.017341 * n_1_4 +
    2.664886 * n_1_5 - 1.239734 * n_1_6 - 1.626478 * n_1_7 - 0.248442 * n_1_8 + 0.880002)))
    n_2_7 = 1 / (1 + Exp(-(2.416051 * n_1_1 + 0.78108 * n_1_2 + 1.411709 * n_1_3 - 2.431481 * n_1_4 -
    3.168485 * n_1_5 + 5.052105 * n_1_6 - 7.795841 * n_1_7 - 3.37931 * n_1_8 - 2.24697)))
    n_2_8 = 1 / (1 + Exp(-(-1.804636 * n_1_1 - 0.83494 * n_1_2 - 2.523328 * n_1_3 + 1.895672 * n_1_4 +
    1.82664 * n_1_5 - 1.038444 * n_1_6 + 2.128415 * n_1_7 - 1.494182 * n_1_8 + 1.057407)))

    t_n = 3.629246 * n_2_1 - 2.159771 * n_2_2 + 1.62898 * n_2_3 + 1.145598 * n_2_4 + 0.881901 * n_2_5 +
    0.576079 * n_2_6 + 1.879723 * n_2_7 - 0.488354 * n_2_8 - 0.106698

    Period5881 = Application.Max(0, t_n * 3.526 + 0.04)

End Function
```

Σχήμα 11: Μετατροπή του forward pass του προτεινόμενου ΤΝΔ με αρχιτεκτονική 5-8-8-1 σε κώδικα VBA.

Εάν είναι επιθυμητή η ευρύτερη διάθεση του εκπαιδευμένου ΤΝΔ και η αξιοποίησή του τόσο από Μηχανικούς όσο και από τα ίδια τα Foundation Models που έχουν πρόσβαση στο διαδίκτυο, είναι δυνατή η ανάπτυξη αντίστοιχου RESTful API σε Python (π.χ. μέσω Flask) ή ακόμη και σε PHP, η οποία υποστηρίζεται εγγενώς σε περιβάλλοντα shared hosting. Με τον τρόπο αυτό, το μοντέλο καθίσταται άμεσα προσβάσιμο μέσω HTTP requests από οποιαδήποτε εφαρμογή.

Στα πλαίσια της παρούσας εργασίας υλοποιήθηκε το RESTful API του συγκεκριμένου εκπαιδευμένου ΤΝΔ σε PHP, το οποίο παρέχει endpoint για αιτήματα GET και POST, με υποστήριξη ανταλλαγής δεδομένων σε μορφή JSON, ενσωματωμένο μηχανισμό ελέγχου εγκυρότητας (validation) των εισερχόμενων παραμέτρων εντός των ορίων εκπαίδευσης, καθώς και CORS compatibility για cross-domain χρήση. Λεπτομέρειες για τη χρήση του παρουσιάζονται στο Σχήμα 12.

JSON για POST Request:

```
{
  "nsto": 5,
  "nspan": 3,
  "sl": 4.5,
  "oratio": 0.25,
  "ws": 15.5
}
```

Παράδειγμα με GET Request:

<https://charalampakis.com/api/period5881.php?nsto=5&nspan=3&sl=4.5&oratio=0.25&ws=15.5>

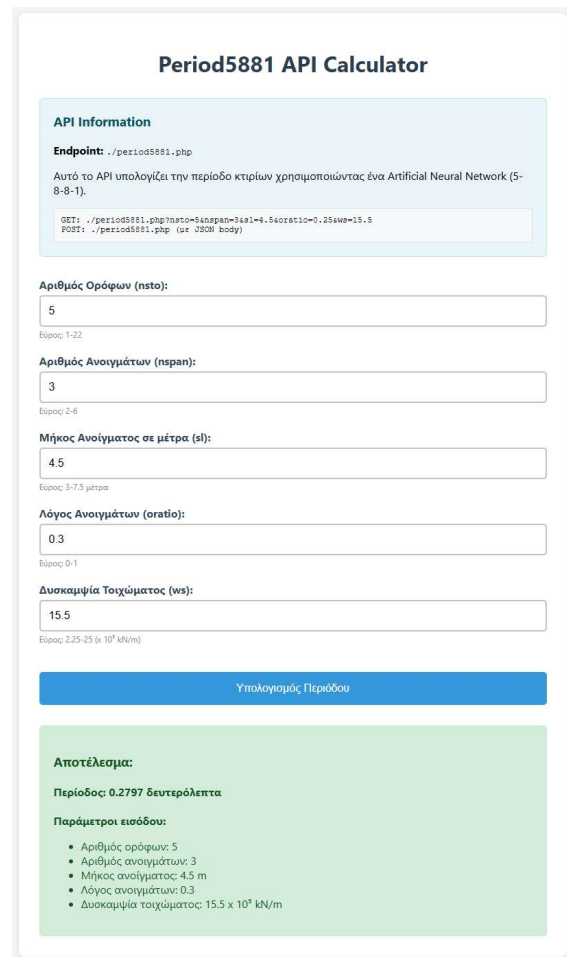
Παράδειγμα με cURL:

```
curl -X POST
https://charalampakis.com/api/period5881.php \
-H "Content-Type: application/json" \
-d '{
  "nsto": 5,
  "nspan": 3,
  "sl": 4.5,
  "oratio": 0.25,
  "ws": 15.5
}'
```

Αναμενόμενη απάντηση:

```
{
  "input": {
    "nsto": 5,
    "nspan": 3,
    "sl": 4.5,
    "oratio": 0.25,
    "ws": 15.5
  },
  "result": 0.25311561945223415,
  "status": "success",
  "timestamp": "2025-08-17T08:27:06+03:00"
}
```

(α)



Period5881 API Calculator

API Information

Endpoint: `./period5881.php`

Αυτό το API υπολογίζει την περίοδο κτιρίων χρησιμοποιώντας ένα Artificial Neural Network (5-8-8-1).

GET: `./period5881.php?nsto=5&nspan=3&sl=4.5&oratio=0.25&ws=15.5`
 POST: `./period5881.php` (use JSON body)

Αριθμός Ορόφων (nsto):

 Εύρος: 1-22

Αριθμός Ανοιγμάτων (nspan):

 Εύρος: 2-6

Μήκος Ανοίγματος σε μέτρα (sl):

 Εύρος: 3-7.5 μέτρα

Λόγος Ανοιγμάτων (oratio):

 Εύρος: 0-1

Δυσκαμψία Τοιχώματος (ws):

 Εύρος: 2.25-25 (x 10⁶ kN/m)

Υπολογισμός Περιόδου

Αποτέλεσμα:

Περίοδος: 0.2797 δευτερόλεπτα

Παράμετροι εισόδου:

- Αριθμός ορόφων: 5
- Αριθμός ανοιγμάτων: 3
- Μήκος ανοίγματος: 4.5 m
- Λόγος ανοιγμάτων: 0.3
- Δυσκαμψία τοιχώματος: 15.5 x 10⁶ kN/m

(β)

Σχήμα 12: API endpoint του εκπαιδευμένου ΤΝΔ στη διεύθυνση <https://charalampakis.com/api/period5881.php>

(α) λεπτομέρειες χρήσης (β) δοκιμαστική εφαρμογή στη διεύθυνση https://charalampakis.com/api/period5881_test_page.html

Λαμβάνοντας υπόψη την απόδοση μοντέλων που έχουν εκπαιδευτεί με την βάση [32] και έχουν παρουσιαστεί στη βιβλιογραφία, καθίσταται προφανές ότι πλέον δεν υπάρχει κάτι περαιτέρω προς αξιοποίηση. Η πρόοδος σε αυτό το πεδίο προϋποθέτει την ανάπτυξη μιας νέας βάσης δεδομένων, η οποία θα περιλαμβάνει πρόσθετες παραμέτρους (π.χ. χαρακτηριστικά εδάφους, λεπτομέρειες σχετικά με τη μορφολογία των ανοιγμάτων κ.ά.) και, κυρίως, σημαντικά μεγαλύτερο αριθμό εγγράφων.

6 ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΠΡΟΟΠΤΙΚΕΣ

Στο πρώτο μέρος της παρούσας εργασίας παρουσιάστηκε μια σύντομη αλλά περιεκτική ιστορική αναδρομή των τεχνολογιών της Μηχανικής Μάθησης και της Τεχνητής Νοημοσύνης, από τις απαρχές τους μέχρι τις πλέον πρόσφατες εξελίξεις. Δεδομένου ότι, κατά την περίοδο συγγραφής του άρθρου, η πρόοδος στον χώρο αυτό είναι καταγιστική, με νέα μοντέλα και μεθοδολογίες να εμφανίζονται σχεδόν καθημερινά, καθίσταται εξαιρετικά δύσκολο να διατυπωθούν ασφαλείς προβλέψεις για το μέλλον.

Σε κάθε περίπτωση, η επιστήμη του Μηχανικού δεν έμεινε ανεπηρέαστη από τις εξελίξεις. Ένας τεράστιος αριθμός ερευνητικών εργασιών έχει δημοσιευθεί τα τελευταία χρόνια, ενώ ο ρυθμός αύξησης των δημοσιεύσεων αυξάνεται συνεχώς. Ωστόσο, η άκριτη εφαρμογή των νέων τεχνολογιών δεν είναι πάντοτε δικαιολογημένη και, σε πολλές περιπτώσεις, τα παραγόμενα μοντέλα στερούνται πρακτικής χρησιμότητας. Δύο είναι οι βασικές προϋποθέσεις, όπως αναλύθηκε: (α) η χρήση των τεχνολογιών αυτών να οδηγεί σε κάποιο μοντέλο με σαφή πλεονεκτήματα έναντι των συμβατικών μεθόδων, και (β) να υπάρχει επάρκεια πληροφορίας, υπό την έννοια των αρκετών ορθά επιλυμένων παραδειγμάτων για την εκπαίδευση, ώστε τα μοντέλα να έχουν ικανοποιητική δυνατότητα γενίκευσης, δηλαδή να αποδίδουν σωστά όταν εκτίθενται σε δεδομένα που διαφέρουν από αυτά από τα οποία εκπαιδεύτηκαν.

Στο δεύτερο μέρος παρουσιάστηκε μια εφαρμογή Μηχανικής Μάθησης, υπό τη μορφή ενός απλού Τεχνητού Νευρωνικού Δικτύου (ΤΝΔ), η οποία μπορεί να χρησιμοποιηθεί για την εκτίμηση της θεμελιώδους ιδιοπεριόδου μιας κατασκευής. Το forward pass του ΤΝΔ μεταφράστηκε σε απλή συνάρτηση για χρήση σε οποιαδήποτε γλώσσα προγραμματισμού ή και απευθείας σε φύλλα εργασίας (Microsoft Excel). Επιπλέον, βασιζόμενο σε αυτή τη συνάρτηση, υλοποιήθηκε RESTful API endpoint, καθιστώντας δυνατή τη χρήση του μοντέλου τόσο από Μηχανικούς όσο και από Foundation Models με πρόσβαση στο διαδίκτυο.

Ως κατακλείδα, αξίζει να τονιστεί ότι οι εποχές που διανύουμε είναι μοναδικές από τεχνολογικής άποψης. Το τεχνολογικό τοπίο είναι σήμερα πλήρως ρευστό, ενώ παραδοσιακές σταθερές καταρρέουν. Το επάγγελμα του προγραμματιστή, για παράδειγμα, αντιμετωπίζει πλέον ένα αβέβαιο μέλλον, κάτι που θα φάνταζε αδιανόητο μόλις λίγα χρόνια πριν. Αντίστοιχα, η αναζήτηση πληροφορίας μέσω παραδοσιακών μηχανών αναζήτησης υποχωρεί ραγδαία, παρότι εκφράσεις όπως «google it» είχαν ενσωματωθεί στην καθημερινή γλώσσα. **Η ενδεχόμενη ανακάλυψη μιας Υπερνοημοσύνης συνιστά γεγονός ασύγκριτης σημασίας, ξεπερνώντας ακόμη και τη Βιομηχανική Επανάσταση.** Πέραν των βαθιών φιλοσοφικών ερωτημάτων που αναδύονται, υπάρχουν και πρακτικά ερωτήματα. Πώς θα διασφαλιστεί η ανθρώπινη φύση απέναντι στη χρήση μιας τέτοιας Υπερνοημοσύνης; Ποια θα είναι η πορεία και το μέλλον της ίδιας της ανθρώπινης ύπαρξης;

7 ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] McCarthy J, Minsky ML, Rochester N, Shannon CE. A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence, August 31, 1955. *AI Mag* 2006;27:12–12. <https://doi.org/10.1609/AIMAG.V27I4.1904>.
- [2] McCarthy J. What is Artificial Intelligence? . [Http://Www-FormalStanfordEdu/Jmc/Whatisai/WhatisaiHtml](http://Www-FormalStanfordEdu/Jmc/Whatisai/WhatisaiHtml) 2007.
- [3] Mitchell M. The future of Artificial Intelligence 2024. <https://www.youtube.com/watch?v=HAiXT1mGTXc> (accessed August 10, 2025).
- [4] Samuel AL. Some studies in machine learning using the game of checkers. *IBM J Res Dev* 1959;3:210–29.
- [5] Deng J, Dong W, Socher R, Li L-J, Kai Li, Li Fei-Fei. ImageNet: A large-scale hierarchical image database. 2009 IEEE Conference on Computer Vision and Pattern Recognition, IEEE; 2009, p. 248–55. <https://doi.org/10.1109/CVPR.2009.5206848>.
- [6] Linnainmaa S. The representation of the cumulative rounding error of an algorithm as a Taylor expansion of the local rounding errors. University of Helsinki, 1970.
- [7] Werbos PJ. Applications of advances in nonlinear sensitivity analysis. Proceedings of the 10th IFIP conference, NYC: 1981, p. 762–70.
- [8] Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors. *Nature* 1986;323:533–6. <https://doi.org/10.1038/323533a0>.
- [9] Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning 2009. <https://doi.org/10.1007/978-0-387-84858-7>.
- [10] Cortes C, Vapnik V, Saitta L. Support-vector networks. *Machine Learning* 1995 20:3 1995;20:273–97. <https://doi.org/10.1007/BF00994018>.
- [11] Quinlan JR. Induction of decision trees. *Machine Learning* 1986 1:1 1986;1:81–106. <https://doi.org/10.1007/BF00116251>.
- [12] Breiman L. Random forests. *Mach Learn* 2001;45:5–32. <https://doi.org/10.1023/A:1010933404324/METRICS>.
- [13] Freund Y, Schapire RE. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J Comput Syst Sci* 1997;55:119–39. <https://doi.org/10.1006/JCSS.1997.1504>.
- [14] Breiman L. Bagging predictors. *Mach Learn* 1996;24:123–40. <https://doi.org/10.1007/BF00058655/METRICS>.
- [15] Cover TM, Hart PE. Nearest Neighbor Pattern Classification. *IEEE Trans Inf Theory* 1967;13:21–7. <https://doi.org/10.1109/TIT.1967.1053964>.
- [16] MacQueen J. Some methods for classification and analysis of multivariate observations. Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, vol. 5.1, 1967, p. 297.
- [17] Jolliffe IT. Principal Component Analysis. New York: Springer-Verlag; 2002. <https://doi.org/10.1007/B98835>.
- [18] Tibshirani R. Regression Shrinkage and Selection Via the Lasso. *J R Stat Soc Series B Stat Methodol* 1996;58:267–88. <https://doi.org/10.1111/J.2517-6161.1996.TB02080.X>.
- [19] Haykin SS. Neural networks and learning machines. Prentice Hall/Pearson; 2009.
- [20] Ptito M, Giguère JF, Boire D, Frost DO, Casanova C. When the auditory cortex turns visual. *Prog Brain Res*, vol. 134, Elsevier; 2001, p. 447–58. [https://doi.org/10.1016/S0079-6123\(01\)34029-3](https://doi.org/10.1016/S0079-6123(01)34029-3).
- [21] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A, et al. Attention is all you need. *Adv Neural Inf Process Syst* 2017;30.
- [22] Mohan A, Poobal S. Crack detection using image processing: A critical review and analysis. *Alexandria Engineering Journal* 2018;57:787–98. <https://doi.org/10.1016/J.AEJ.2017.01.020>.
- [23] Zhou Y, Meng S, Lou Y, Kong Q. Physics-Informed Deep Learning-Based Real-Time Structural Response Prediction Method. *Engineering* 2024;35:140–57. <https://doi.org/10.1016/J.ENG.2023.08.011>.
- [24] Charalampakis AE, Papanikolaou VK. Machine learning design of R/C columns. *Eng Struct* 2021;226. <https://doi.org/10.1016/j.engstruct.2020.111412>.
- [25] Liu X, Athanasiou CE, Padture NP, Sheldon BW, Gao H. A machine learning approach to fracture mechanics problems. *Acta Mater* 2020;190:105–12. <https://doi.org/10.1016/J.ACTAMAT.2020.03.016>.
- [26] Cai S, Wang Z, Wang S, Perdikaris P, Karniadakis GE. Physics-informed neural networks for heat transfer problems. *J Heat Transfer* 2021;143. <https://doi.org/10.1115/1.4050542/1104439>.
- [27] Yin M, Ban E, Rego B V., Zhang E, Cavinato C, Humphrey JD, et al. Simulating progressive intramural damage leading to aortic dissection using DeepONet: An operator-regression neural network. *J R Soc Interface* 2022;19. <https://doi.org/10.1098/RSIF.2021.0670;PAGEGROUP:STRING:PUBLICATION>.

- [28] Raissi M, Perdikaris P, Karniadakis GE. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J Comput Phys* 2019;378:686–707. <https://doi.org/10.1016/J.JCP.2018.10.045>.
- [29] Raissi M, Perdikaris P, Karniadakis GE. Physics Informed Deep Learning (Part I): Data-driven Solutions of Nonlinear Partial Differential Equations 2017.
- [30] Asteris PG, Repapis CC, Tsaris AK, Di Trapani F, Cavaleri L. Parameters affecting the fundamental period of infilled RC frame structures. *Earthquake and Structures* 2015;9:999–1028. <https://doi.org/10.12989/eas.2015.9.5.999>.
- [31] Asteris PG, Repapis CC, Repapi E V., Cavaleri L. Fundamental period of infilled reinforced concrete frame structures. *Structure and Infrastructure Engineering* 2017;13:929–41. <https://doi.org/10.1080/15732479.2016.1227341>.
- [32] Asteris PG. The FP4026 Research Database on the fundamental period of RC infilled frame structures. *Data Brief* 2016;9:704–9. <https://doi.org/10.1016/j.dib.2016.10.002>.
- [33] CEN CE de Normalisation. Eurocode 2: Design of concrete structures—part 1-1: general rules and rules for buildings 2004:EN 1992-1-1.
- [34] CEN CE de Normalisation. Eurocode 8: Design of Structures for Earthquake Resistance—Part 1: General Rules. Seismic Actions and Rules for Buildings 2004:EN 1998-1.
- [35] Crisafulli FJ, Carr AJ. Proposed macro-model for the analysis of infilled frame structures. *Bulletin of the New Zealand Society for Earthquake Engineering* 2007;40.
- [36] Charalampakis AE, Tsiatas GC, Kotsiantis SB. Machine learning and nonlinear models for the estimation of fundamental period of vibration of masonry infilled RC frame structures. *Eng Struct* 2020;216:110765. <https://doi.org/10.1016/j.engstruct.2020.110765>.
- [37] Markou G, Bakas NP, Chatzichristofis SA, Papadrakakis M. A general framework of high-performance machine learning algorithms: application in structural mechanics. *Comput Mech* 2024;73:705–29. <https://doi.org/10.1007/s00466-023-02386-9>.
- [38] Thisovithan P, Aththanayake H, Meddage DPP, Ekanayake IU, Rathnayake U. A novel explainable AI-based approach to estimate the natural period of vibration of masonry infill reinforced concrete frame structures using different machine learning techniques. *Results in Engineering* 2023;19:101388. <https://doi.org/10.1016/J.RINENG.2023.101388>.
- [39] Yahiaoui A, Dorbani S, Yahiaoui L. Machine learning techniques to predict the fundamental period of infilled reinforced concrete frame buildings. *Structures* 2023;54:918–27. <https://doi.org/10.1016/J.ISTRUC.2023.05.052>.
- [40] Rahman T, Zheng P, Sultana S. Bayesian Optimized LightGBM model for predicting the fundamental vibrational period of masonry infilled RC frames. *Frontiers of Structural and Civil Engineering* 2024;18:1084–102. <https://doi.org/10.1007/S11709-024-1077-Z/METRICS>.
- [41] Asteris PG, Nikoo M. Artificial bee colony-based neural network for the prediction of the fundamental period of infilled frame structures. *Neural Comput Appl* 2019. <https://doi.org/10.1007/s00521-018-03965-1>.
- [42] Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *Adv Neural Inf Process Syst* 2019;32.
- [43] Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH. The WEKA data mining software. *ACM SIGKDD Explorations Newsletter* 2009;11:10–8. <https://doi.org/10.1145/1656274.1656278>.