ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ – ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Πρόγραμμα Μεταπτυχιακών Σπουδών

«ΠΜΣ ΠΛΗΡΟΦΟΡΙΚΗ»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	**Μοντελοποίηση Ακολουθιών με Βαθιά Νευρωνικά Δίκτυα** **Sequence Modelling with Deep Neural Networks**
Ονοματεπώνυμο Φοιτητή	**Κωνσταντάρας Θεοδόσιος**
Πατρώνυμο	**Παναγιώτης**
Αριθμός Μητρώου	**ΜΠΠΛ15034**
Επιβλέπων	**Γεώργιος Τσιχριντζής, Καθηγητής**

Ημερομηνία Παράδοσης **Σεπτέμβριος 2022**

Τριμελής Εξεταστική Επιτροπή

Γεώργιος Τσιχριντζής Καθηγητής	Ευάγγελος Σακκόπουλος Αναπληρωτής Καθηγητής	Διονύσιος Σωτηρόπουλος Επίκουρος Καθηγητής

Θα ήθελα να ευχαριστήσω τον Καθηγητή, κύριο Γεώργιο Τσιχριντζή, και τον Επίκουρο Καθηγητή, κύριο Σωτηρόπουλο Διονύσιο, του τμήματος Πληροφορικής του Πανεπιστημίου Πειραιώς, για την άμεση υποστήριξή και βοήθεια τους ώστε να ολοκληρώσω τις σπουδές μου.

Περίληψη

Στην παρούσα μεταπτυχιακή διατριβή επιδιώκεται η παρουσίαση της σύγχρονης βιβλιογραφίας που σχετίζεται με το αντικείμενο της Μοντελοποίησης Ακολουθιών μέ Βαθιά Νευρωνικά Δίκτυα (Deep Neural Networks). Η βασική ανάλυση αφορά τις αρχιτεκτονικές και τα μοντέλα Αναδρομικών Νευρωνικών Δικτύων (Recurrent Neural Networks - RNN), τα οποία βρίσκονται στο επίκεντρο της σύγχρονης επιστημονικής έρευνας και φαίνεται να παρουσιάζουν τα καλύτερα αποτελέσματα σε θέματα μοντελοποίησης ακολουθιών. Προσπάθειες για τη μοντελοποίηση ακολουθιών έχουν γίνει και με διαφορετικά μοντέλα νευρωνικών δικτύων, όπως είναι τα Συνελικτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks - CNN), τα οποία, όμως, δεν εξετάζονται στην παρούσα εργασία. Στην εργασία παρουσιάζονται οι περιορισμοί και τα μειονεκτήματά των κλασσικών δομών RNN για τη μοντελοποίηση ακολουθιών καθώς επίσης και η αρχιτεκτονική LSTM, που επεκτείνει τις δυνατότητες των RNN δικτύων. Τέλος, αναφέρονται οι λόγοι που οδήγησαν στην ανάπτυξη των μοντέλων μετασχηματιστή (Transformers), και παρουσιάζεται συνοπτικά η αρχιτεκτονική τους.

Abstract

The purpose of the present postgraduate dissertation is to present modern bibliography on the field of sequence modeling with deep neural networks. The main body of the analysis is about the Recurrent Neural Network architectures (RNN) which are at the heart of modern scientific research and seem to present the best results in the field. Attempts to model sequences have been made with different models of neural networks, such as the Convolutional Neural Networks (CNN), which, however, are not considered in this thesis. The dissertation presents the limitations and disadvantages of classical RNN structures for sequence modeling as well as the LSTM architecture, which extends the capabilities of RNN networks. Finally, the dissertation mentions the reasons that led to the development of Transformers models and their architecture is briefly presented.

Πίνακας Περιεχομένων

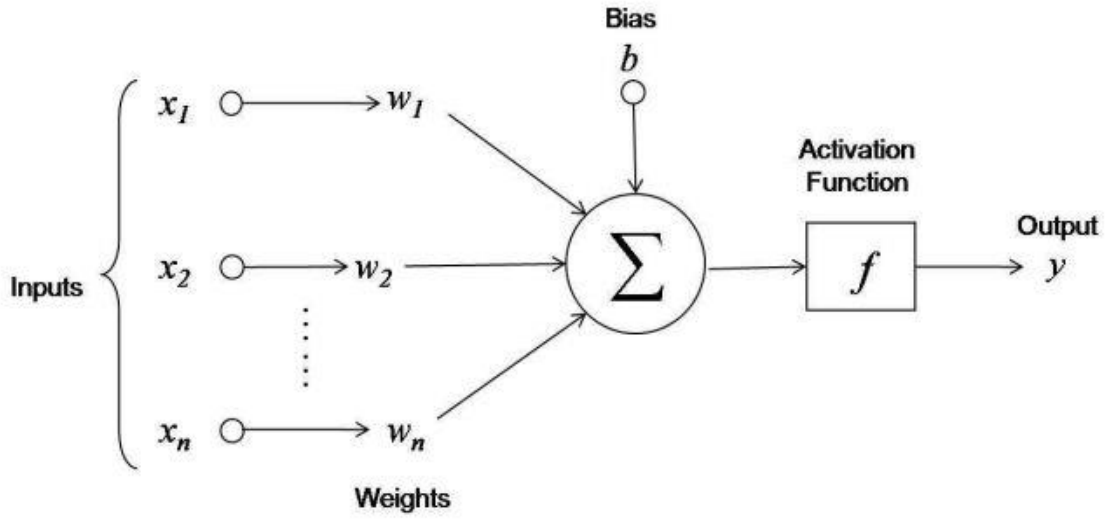
Αρχές Νευρωνικών Δίκτύων

Κλασσική Αρχιτεκτονική Νευρωνικών Δικτύων

Ο κορμός των νευρωνικών δικτύων είναι τα λεγόμενα feedforward μοντέλα και συγκεκριμένα το μοντέλο perceptron. Το perceptron γεννήθηκε την δεκαετία του ’50 από τον ψυχολόγο Frank Rosenblatt, ο οποίος προσπαθούσε να μοντελοποιήσει την λειτουργία με την οποία ο ανθρώπινος εγκέφαλος αντιλαμβάνεται, διατηρεί και επεξεργάζεται τις ποικίλες πληροφορίες που λαμβάνει συνεχώς από τον φυσικό κόσμο. Η δομή και η λειτουργία του perceptron είναι οι βάσεις των σύγχρονων νευρωνικών δικτύων και για το λόγο αυτό παρουσιάζονται συνοπτικά στις ακόλουθες παραγράφους.

Μοντέλο Perceptron

Το θεμελιώδες κομμάτι της αρχιτεκτονικής ενός νευρωνικού δικτύου είναι ο νευρώνας. Ο νευρώνας ορίζεται ως μια μαθηματική συνάρτηση η οποία παίρνει σαν δεδομένα εισόδου μια ή περισσότερες τιμές και υπολογίζει μια τιμή εξόδου. Είναι το κομμάτι του νευρωνικού δικτύου το οποίο καλείται να επεξεργαστεί την όλη πληροφορία που έχει λάβει σαν είσοδο και να παράγει το αποτέλεσμα αυτής της επεξεργασίας.



Εικόνα 1- Αρχιτεκτονική του Perceptron

Στην παραπάνω εικόνα αναπαρίσταται ενας νευρώνας μαζί με τα δεδομένα εισόδου του, την συνάρτηση ενεργοποίησης και τα δεδομένα εξόδου καθώς επίσης και τον όρο μεροληψίας (bias term). Το μοντέλο της εικόνας είναι το πιο απλό νευρωνικό δίκτυο που υπάρχει και στην

βιβλιογραφία απαντάται συχνά ως Perceptron. Αυτό το υπολογιστικό μοντέλο χρησιμοποιείται ως γραμμικός ταξινομητής και μπορεί να εφαρμοστεί σε περιπτώσεις όπου υπάρχει γραμμικός διαχωρισμός μεταξύ των δεδομένων. Η συνάρτηση του Perceptron δέχεται τα δεδομένα εισόδου, τα οποία συνήθως είναι ενα διάνυσμα πραγματικών τιμών, και τα αντιστοιχίζει σε μια και μοναδική τιμή εξόδου. Με πολύ απλά λόγια, ο αλγόριθμος λειτουργεί ως εξής:

- Αθροίζει όλες τις τιμές εισόδου ($x_1, x_2, \dots, x_n$) σε συνδυασμό με τα αντίστοιχα βάρη ($w_1, w_2, \dots, w_3$) τα οποία έχουμε προσαρτήσει σε κάθε τιμή εισόδου.
- Προσθέτει τον όρο μεροληψίας (b).
- Εφαρμόζει μια γραμμική συνάρτηση ενεργοποίησης (f) στους προηγούμενους υπολογισμούς.
- Σε περίπτωση που το αποτέλεσμα της συνάρτησης είναι μεγαλύτερο ή μικρότερο από κάποια οριακή τιμή, τότε κατηγοριοποιεί το αποτέλεσμα στην αντίστοιχη κλάση (y) (τιμές 0 ή 1).

Μπορούμε να παρουσιάσουμε την παραπάνω διαδικασία υπολογισμού της εξόδου του perceptron με τη μαθηματική της μορφή ως εξής:

Άθροισμα των τιμών εισόδου σε συνδυασμό με τα αντίστοιχα βάρη

$$z = w_1 x_1 + w_2 x_2 + \cdots + w_n x_n = \sum_{i=1}^{n} w_i x_i = \boldsymbol{W}^T \boldsymbol{X} \tag{1.1}$$

Προσθήκη όρου μεροληψίας

$$z = w_1 x_1 + w_2 x_2 + \cdots + w_n x_n + b = \sum_{i=1}^{n} w_i x_i + b = \boldsymbol{W}^T \boldsymbol{X} + b \tag{1.2}$$

Εφαρμογή συνάρτησης ενεργοποίησης

$$y = f(z) = f(\boldsymbol{W}^T \boldsymbol{X} + b) \tag{1.3}$$

όπου τα σύμβολα $\boldsymbol{W}$ και $\boldsymbol{X}$ εκφράζουν διανύσματα στήλες και η έκφραση $\boldsymbol{W}^T \boldsymbol{X}$ συμβολίζει το εσωτερικό τους γινόμενο. Συγκεκριμένα, τα διανύσματα έχουν την παρακάτω μορφή:

$$\boldsymbol{X} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} \qquad \boldsymbol{W} = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \vdots \\ w_n \end{bmatrix}$$

Η απόφαση για την κατηγοριοποίηση της τιμής λαμβάνεται στο βήμα της εφαρμογής της συνάρτησης ενεργοποίησης. Η τελική τιμή y προκύπτει ως εξής:

$$y = f(z) = \begin{cases} 1 \; if \; z \geq \theta \\ 0 \; if \; z < \theta \end{cases} \tag{1.4}$$

Η παραπάνω διαδικασία περιγράφει τον τρόπο λειτουργίας του αλγόριθμου Perceptron. Στη συνέχεια, θα περιγράψουμε ενα πιο αποτελεσματικό μοντέλο νευρωνικού δικτύου, το οποίο βασίζεται στο Perceptron, όμως έχει πολύ περισσότερες υπολογιστικές δυνατότητες: το πολυεπίπεδο Perceptron.

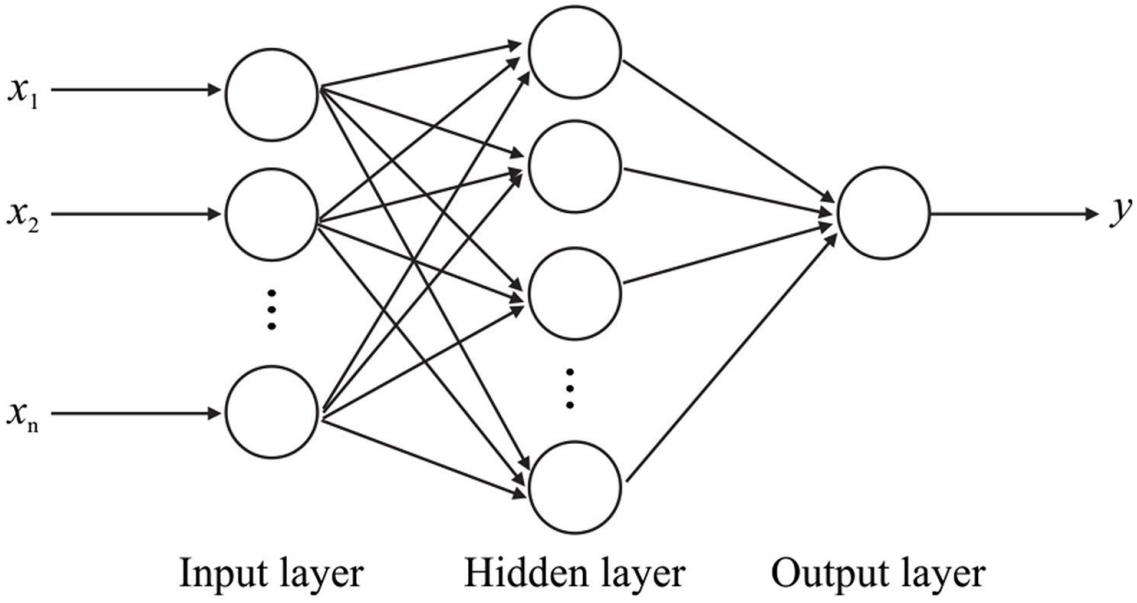
Πολυεπίπεδο Perceptron (Multilayer Perceptron - MLP)

Ένα δίκτυο αποτελούμενο από ενα και μόνο νευρώνα, δηλαδή το μοντέλο Perceptron που παρουσιάστηκε στην προηγούμενη υπο-ενότητα, έχει τον περιορισμό οτι δε μπορεί να εφαρμοστεί σε δεδομένα τα οποία έχουν μη γραμμική σχέση μεταξύ τους και, συνεπώς, να μάθει πιο περίπλοκες, μη-γραμμικές συναρτήσεις. Για το λόγο αυτό, προτάθηκε ενα διαφορετικό είδος αρχιτεκτονικής, το πολυεπίπεδο perceptron.

Όπως προδίδει και το όνομά του, ενα πολυεπίπεδο Perceptron έχει ως δομικό του στοιχείο τον νευρώνα Perceptron. Ενα MLP αποτελείται από τουλάχιστον τρία επίπεδα κόμβων (nodes):

- Ενα επίπεδο εισαγωγής δεδομένων (Input Layer),
- Ενα επίπεδο εξαγωγής δεδομένων (Output Layer)
- Ενα ή περισσότερα κρυφά επίπεδα (Hidden Layers) τα οποία συνθέτονται, συνήθως, από αρκετούς στοιβαγμένους (stacked) κόμβους/νευρώνες Perceptron.

Στην εικόνα που ακολουθεί, φαίνεται ενα πολυεπίπεδο Perceptron, ή αλλιώς, ενα πολυεπίπεδο νευρωνικό δίκτυο.

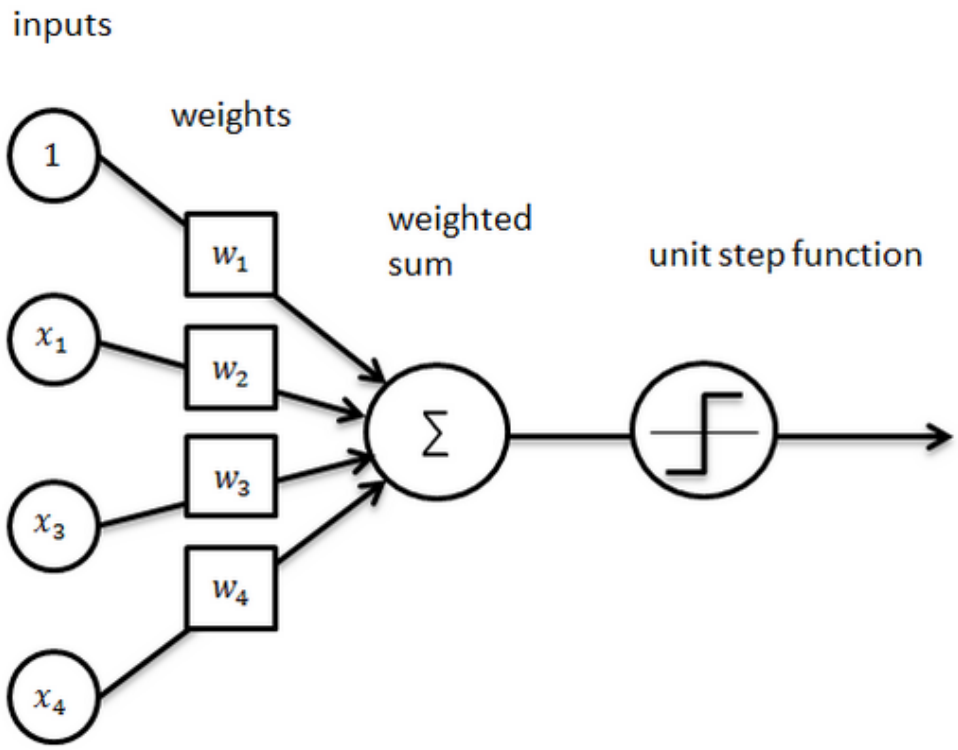


Εικόνα 2 - Αρχιτεκτονική MLP

Στο παραπάνω γράφημα μπορούμε να διακρίνουμε ενα MLP δίκτυο με τα τρία επίπεδα που αναφέρθηκαν νωρίτερα. Συγκεκριμένα:

- Στο επίπεδο εισαγωγής δεδομένων (Input Layer) ο κάθε κόμβος λαμβάνει τα δεδομένα και τα προωθεί στα επόμενα επίπεδα του δικτύου.
- Το κρυφό επίπεδο (Hidden Layer) είναι ο βασικός υπολογιστικός κορμός του δικτύου. Ενα νευρωνικό δίκτυο μπορεί να περιέχει παραπάνω από ενα κρυφά επίπεδα.
- Το επίπεδο εξαγωγής είναι αυτό στο οποίο καταλήγουν οι υπολογισμοί των προηγούμενων επιπέδων και παράγει το τελικό αποτέλεσμα.

Τα αποτελέσματα από κάθε κόμβο του νευρωνικού δικτύου προωθούνται στο επόμενο επίπεδο σαν δεδομένα εισαγωγής. Όπως και στο Perceptron, ο κάθε κόμβος λαμβάνει τα δεδομένα, τα επεξεργάζεται και παράγει ενα αποτέλεσμα. Αν μεγενθύνουμε εναν μεμονωμένο κόμβο του παραπάνω δικτύου, θα δούμε οτι έχει την ακόλουθη δομή:



Εικόνα 3 - Κόμβος ενός MLP

Όπως είπαμε και νωρίτερα, ενα MLP αποτελέιται από πολλά στοιβαγμένα Perceptron, γι'αυτό και η δομή ενός κόμβου του MLP είναι ίδια με την δομή του νευρώνα Perceptron. Η συνάρτηση ενεργοποίησης του παραπάνω σχήματος είναι η μοναδιαία βηματική συνάρτηση, ωστόσο μπορεί να επιλεγεί οποιαδήποτε συνάρτηση ενεργοποίησης ταιριάζει καλύτερα ανάλογα με τις ανάγκες του νευρωνικού δικτύου. Ενα MLP είναι συνώνυμο με τα Βαθιά Νευρωνικα Δίκτυα Εμπρόσθιας Τροφοδότησης (Deep Feedforward Networks) τα οποία αποτελούν τα κυρίαρχα μοντέλα βαθιάς μάθησης (Deep Learning). Το όνομα των μοντέλων αυτών προκύπτει από το γεγονός οτι η πληροφορία προωθείται από το επίπεδο εισαγωγής, διερχόμενη από τα κρυφά επίπεδα στα οποία γίνονται οι βασικοί υπολογισμοί, πρός το επίπεδο εξαγωγής, χωρίς να υπάρχουν συνδέσεις ανατροφοδότησεις (feedback connections) των παραγόμενων δεδομένων πρός το δίκτυο. Επίσης, ο όρος βαθιά μάθηση εξάγεται από την ίδια την αρχιτεκτονική των μοντέλων αυτών, η οποία ουσιαστικά συνδέει διάφορες συναρτήσεις μεταξύ τους. Θεωρούμε πως κάθε κρυφό επίπεδο ενός νευρωνικού δικτύου αναπαριστά μια συνάρτηση $f^{(i)}(\boldsymbol{x})$, ώπου i είναι το κρυφό επίπεδο του δικτύου. Οπότε, στην περίπτωση ενός νευρωνικού δικτύου με τα τρια επίπεδα (3 hidden layers), μπορούμε να το αναπαραστήσουμε με μια αλυσιδωτή δομή η οποία συνδέει τις συναρτήσεις κάθε επιπέδου ως εξής:

$$f(\boldsymbol{x}) = f^{(3)}\left(f^{(2)}\left(f^{(1)}(\boldsymbol{x})\right)\right) \quad (1.4)$$

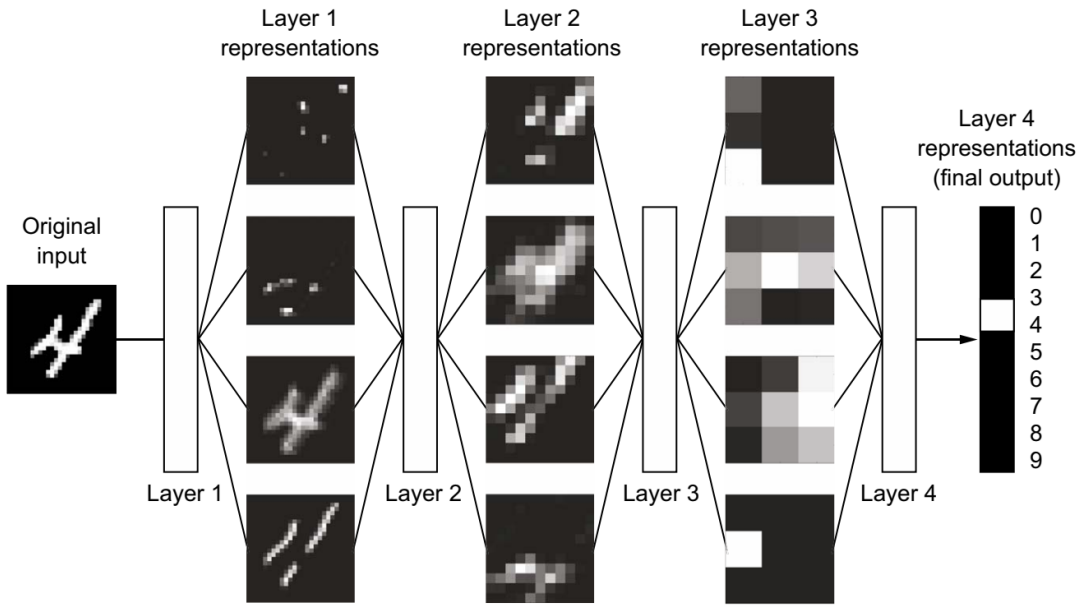
Σύμφωνα με την προηγούμενη περιγραφή, οι παραπάνω συναρτήσεις αντιστοιχούν στα διάφορα κρυφά επίπεδα του νευρωνικού δικτύου: $f^{(1)}$ είναι το πρώτο κρυφό επίπεδο, η $f^{(2)}$ το δευτερο και η $f^{(3)}$ το τρίτο κρυφό επίπεδο του δικτύου και ανάλογα με το μήκος αυτής της αλυσίδας προκύπτει και το βάθος του νευρωνικού δικτύου. Τέλος, ο όρος κρυφό επίπεδο προκύπτει από το γεγονός οτι δεν υπάρχει πρόσβαση από το επίπεδο διεπαφής του νευρωνικού δικτύου, δηλαδή τα επίπεδα εισαγωγής και εξαγωγής δεδομένων σε αυτά τα ενδιάμεσα κρυφά επίπεδα. Η ύπαρξη και η λειτουργεία των κρυφών επιπέδων είναι γνωστή μόνο στο νευρωνικό δίκτυο που τα υλοποιεί.

Σκοπός των παραπάνω μοντέλων είναι να μάθουν, μέσα από την έκθεσή τους σε εναν μεγάλο όγκο δεδομένων, να προβλέπουν και να κατηγοριοποιούν νεα δεδομένα, τα οποία δεν έχουν υπάρξει ξανά μέσα στο δίκτυο. Στην παρακάτω ενότητα γίνεται μια περιγραφή της διαδικασίας μάθησης των πολυεπίπεδων νευρωνικών δικτύων.

Διαδικασία Εκπαίδευσης Νευρωνικών Δικτύων

Στην προηγούμενη ενότητα παρουσιάστηκε η πιο πολυχρησιμοποιημένη και ευρέως γνωστή αρχιτεκτονική πολυεπίπεδων νευρωνικών δικτύων, το MLP. Στην παρούσα ενότητα, θα παρουσιάσουμε, χωρίς να εμβαθύνουμε στην ανάλυση των μαθηματικών που διέπουν τις έννοιες που θα ακολουθήσουν, τον τρόπο με τον οποίο καταφέρνουν τα πολυεπίπεδα δίκτυα εμπρόσθιας τροφοδότησης να αποκτήσουν την γνώση που χρειάζεται ώστε να επιτυγχάνουν τις πιο ακριβείς προβλέψεις.

Όσο πιο μεγάλη είναι η έκθεση των πολυεπίπεδων νευρωνικών δικτύων σε δεδομένα, τόσο περισσότερη γνώση αποκτάνε στο να μπορούν να κατηγοριοποιούν νεα δεδομένα. Ενα απλό παράδειγμα λειτουργίας ενός δικτύου μπορεί να είναι η κατηγοριοποίηση μιας εικόνας ενός χειρόγραφου αριθμού. Αρχικά παρέχουμε στο δίκτυο μια πληθώρα από εικόνες, μαζί με την αντίστοιχη κατηγορία (Label) της κάθε εικόνας, και το εκπαιδεύουμε. Τα δεδομένα αυτά ονομάζονται δεδομένα εκπαίδευσης (Training Data) και αποτελούν την πηγή γνώσης του νευρωνικού δικτύου. Αφού το δίκτυο εκπαιδευθεί σε ικανοποιητικό βαθμό, του παρέχουμε δεδομένα τα οποία δεν έχει «ξαναδεί» (Validation Data), όπως στο παράδειγμα του χειρόγραφου αριθμού, μια καινούρια εικόνα ενός χειρόγραφου αριθμού, με σκοπό να μας δώσει σαν αποτελέσμα την κατηγορία του αριθμού, δηλαδή ποιός αριθμός είναι και να υπολογίσουμε την ικανότητα του δικτύου να εξάγει σωστά αποτελέσματα για νεα δεδόμενα. Στην παρακάτω εικόνα μπορούμε να δούμε ενα εκπαιδευμένο νευρωνικό δίκτυο, το οποίο αντιστοιχεί στην παραπάνω περιγραφή.



Εικόνα 4 – Εκτίμηση εικόνας χειρόγραφου αριθμού από πολυεπίπεδο νευρωνικό δίκτυο

Όπως φαίνεται στην παραπάνω εικόνα, τα διαδοχικά κρυφά επίπεδα του δικτύου μεταμορφώνουν την αρχική εικόνα σε όλο και πιο αφηρημένες αναπαραστάσεις, ώσπου, τελικά, να καταφέρουν να βρούν χαρακτηριστικά (Features) παρόμοια με αυτά τα οποία έχουν ανακαλύψει μέσα από την διαδικασία εκμάθησης και να αντιστοιχίσουν την αρχική εικόνα στην σωστή κατηγορία.

Μέχρι στιγμής έχουμε δει τον τρόπο με τον οποίο τα νευρωνικά δίκτυα υπολογίζουν μέσα από την διαδικασία της εμπρόσθιας τροφοδότησης (Feed Forward Pass) μια εκτίμηση για τα δεδομένα που έλαβαν στο επίπεδο εισαγωγής. Αυτή η εκτίμηση, όμως, δεν παρέχει κάποια χρήσιμη πληροφορία προς το δίκτυο αν δεν αξιοποιηθεί κατάλληλα. Τα βασικά εργαλεία που χρησιμοποιούν τα νευρωνικά δίκτυα κατά την διαδικασία εκμάθησης είναι:

- Η διαδικασία της εμπρόσθιας τροφοδότησης
- Ο υπολογισμός του κόστους μέσω της συνάρτησης κόστους (Cost/Loss Function)
- Η ελαχιστοποίηση της συνάρτησης κόστους μέσω του αλγορίθμου αντίστροφης διάδοσης σφαλμάτων (Backpropagation Algorithm)

Συνάρτηση Κόστους

Για να μπορέσει ενα νευρωνικό δίκτυο να εξάγει εκτιμήσεις όσο το δυνατόν πιο κοντά στις τιμές που θα θέλαμε, ή αλλιώς, να μπορεί να προσομοιάζει όσο το δυνατόν καλυτερα την συνάρτηση που θα θέλαμε, θα πρέπει να τροφοδοτήσει την εκτίμηση/πρόβλεψη της διαδικασίας της εμπρόσθιας τροφοδότησης σε μια συνάρτηση κόστους, η οποία θα μας δώσει την απόκλιση της αναμενόμενης τιμής (η τιμή που θα θέλαμε να επιτύχει το δίκτυο) από την τιμή που δώσαμε

στην συνάρτηση κόστους. Καθώς ο σκοπός της εκπαίδευσης των νευρωνικών δικτύων είναι να μοντελοποιήσουν τον τρόπο με τον οποίο παράχθηκαν τα δεδομένα με τα οποία το εκπαιδεύουμε, προτεραιότητα του έιναι η ελαχιστοποίηση της διαφοράς μεταξύ της αναμενόμενης τιμής και της τιμής που προέβλεψε το δίκτυο. Δηλαδή η ελαχιστοποιήση της συνάρτησης κόστους, η οποία έχει την παρακάτω γενική μορφή:

$$\mathcal{L} = (\hat{y}_k, y_k) \tag{1.5}$$

Όπου $\hat{y}_k$ είναι η εκτίμηση του νευρωνικού δικτύου και y_k είναι η τιμή στόχος που θα θέλαμε να επιτύχει το δίκτυο.

Αλγόριθμος Αντίστροφης Διάδοσης Σφαλμάτων (Backpropagation Algorithm)

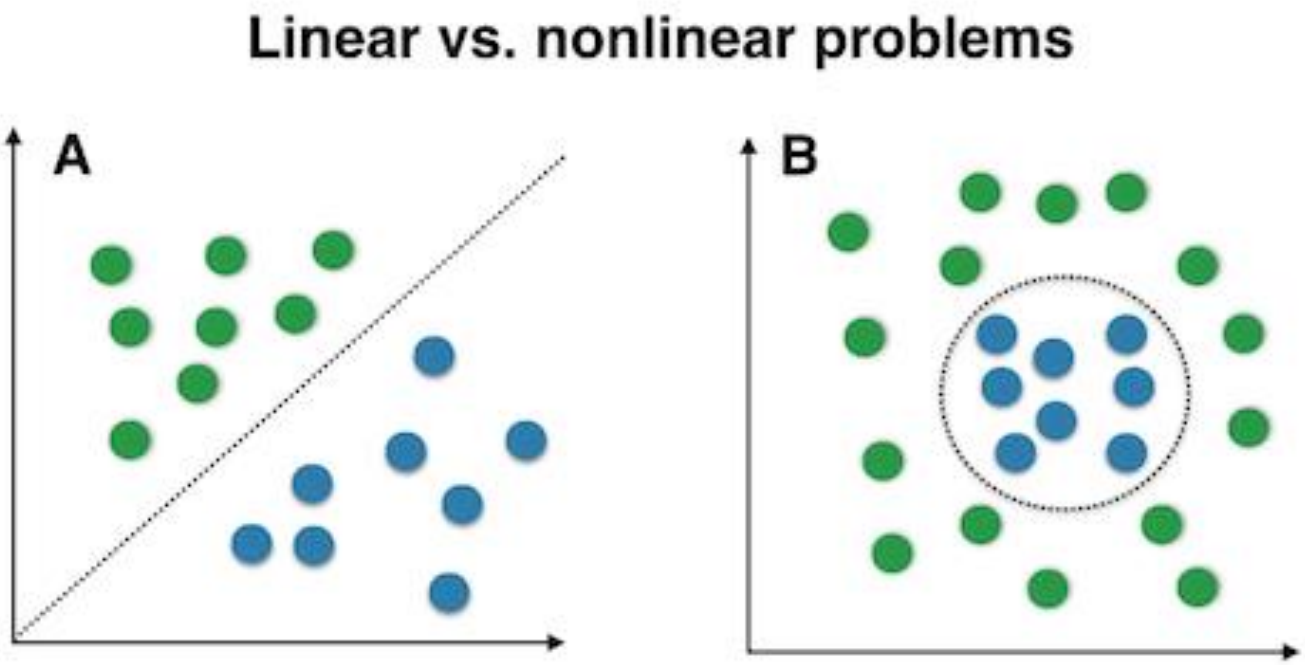
Ο κύριος σκοπός της συνάρτησης κόστους είναι να μετρήσει το μέγεθος του σφάλματος που προκύπτει από τους υπολογισμούς του μοντέλου του νευρωνικού δικτύου, ώστε να καθοδηγήσει την διαδικασία της εκπαίδευσης πρός την κατεύθυνση ελαχιστοποίησης του σφάλματος αυτού (κόστος). Αφού υπολογισθεί το σφάλμα, η διαδικασία εκμάθησης συνεχίζεται με την υλοποίηση του αλγόριθμου αντίστροφης διάδοσης σφάλματος. Λαμβάνοντας υπόψιν την τιμή του σφάλματος, την οποία τροφοδοτεί πρός τα πίσω στο μοντέλο (backpropagation), ο αλγόριθμος επιδιώκει, μέσω συνεχών εκτελέσεων (ο αλγόριθμος εκτελείται για κάθε παράδειγμα που λαμβάνεται στο επίπεδο εισαγωγής δεδομένων) των διαδικασιών εμπρόσθιας τροφοδότησης και αντίστροφης διάδοσης, να ρυθμίσει τους συντελεστές των βαρών που εφαρμόζονται στα δεδομένα εισαγωγής. Η ρύθμιση των συντελεστών γίνεται με τέτοιο τρόπο, ώστε στο τέλος της διαδικασίας να έχει βρεθεί ο κατάλληλος συνδυασμός βαρών που θα μας δώσει, για τα εκάστοτε δεδομένα εισαγωγής, τα αντίστοιχα αποτελέσματα τα οποία θέλουμε να είναι, αν όχι ίδια, πάρα πολύ κοντά στις αναμενόμενες τιμές.

Για την επίτευξη του στόχου αυτού, δηλαδή για την εύρεση των κατάλληλων συντελεστών των βαρών, η διαδικασία της αντίστροφης διάδοσης σφαλμάτων χρησιμοποιεί τον αλγόριθμο βαθμιδωτής καθόδου (gradient descent). Ουσιαστικά πρόκεται για εναν αλγόριθμο βελτιστοποίησης, του οποίου η βασική ιδέα είναι σε κάθε επανάληψη να υπολογίζει την παράγωγο της συνάρτησης κόστους, για κάθε παράμετρο της συνάρτησης, και στην συνέχεια να ενημερώνει τις τιμές των συντελεστών πρός την αντίθετη κατεύθυνση από αυτή που δείχνει η παράγωγος.

Παραδείγματα Συναρτήσεων Ενεργοποίησης

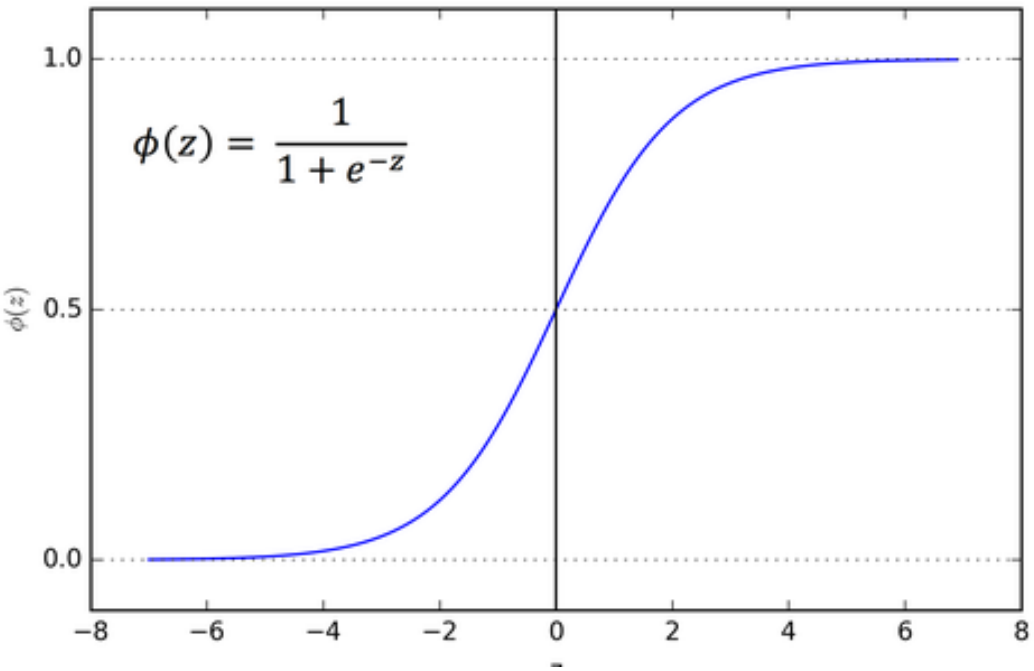
Η επιλογή της συνάρτησης ενεργοποίησης παίζει καθορίστικό ρόλο κατά την διάρκεια της εκπαίδευσης ενός νευρωνικού δικτύου, καθώς είναι εκείνη που διαχωρίζει τα γραμμικά μοντέλα από τα μη γραμμικά μοντέλα. Όσο πιο περίπλοκες είναι οι εξαρτήσεις μεταξύ των στοιχείων ενός συνόλου δεδομένων τα οποία δίνονται ως δεδομένα εισαγωγής στο δίκτυο, τα γραμμικά μοντέλα δεν έχουν τη δυνατότητα να μάθουν την συνάρτηση η οποία εκφράζει αυτές τις

εξαρτήσεις. Παρακάτω παρουσιάζονται οι πιο διαδεδομένες συναρτήσεις ενεργοποίησης που χρησιμοποιούνται για την εκπαίδευση των νευρωνικών δικτύων.



Εικόνα 5 - Διαφορά μεταξύ γραμμικής συνάρτησης και μη γραμμικής συνάρτησης ενεργοποίησης

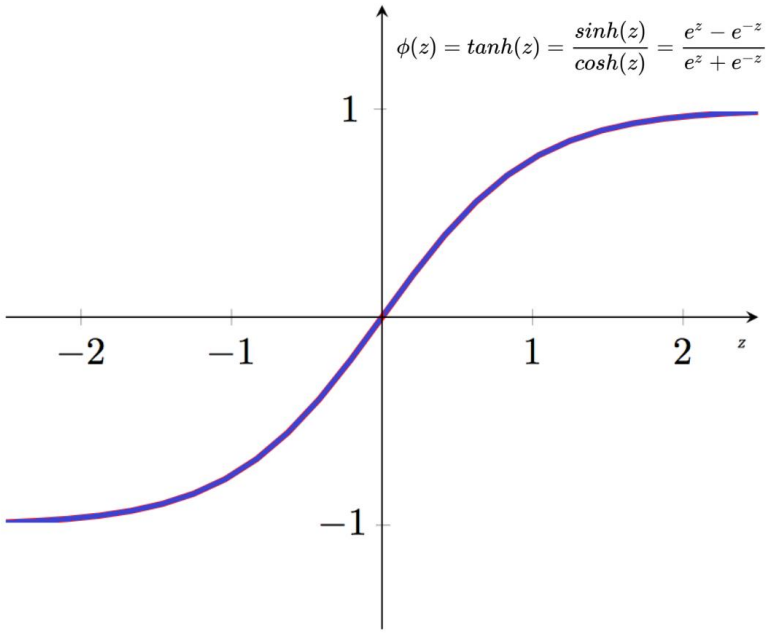
Στην εικόνα 5 φαίνεται η διαφορά μεταξύ της επιλογής μιας γραμμικής συνάρτησης ενεργοποίησης και μιας μη γραμμικής συνάρτησης. Η επιλογή μιας γραμμικής συνάρτησης θα ήταν αδύνατο να μοντελοποίησει τα δεδομένα του γραφήματος Β, καθώς φαίνεται πώς δεν έχουν κάποια γραμμική συσχέτιση.



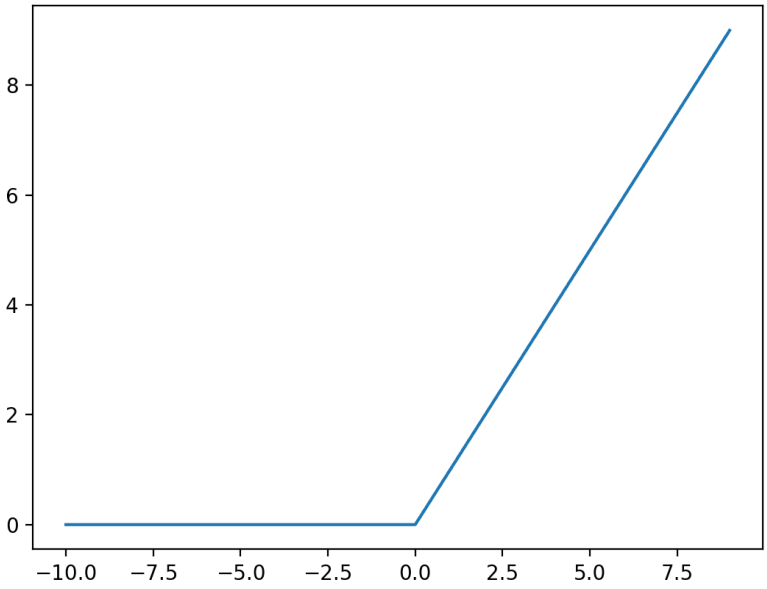
Εικόνα 6 - Σιγμοειδής συνάρτηση

Η σιγμοειδής συνάρτηση, λόγω της ιδιότητας της να περιφράζει τα δεδομένα ανάμεσα στις τιμές 0 και 1, χρησιμποιείται συνήθως σε περιπτώσεις που θέλουμε να παράγουμε τιμές πιθανοτήτων.

Η υπερβολική εφαπτομένη στην εικόνα 7 δίνει τη δυνατότητα να ενεργοποιηθέι ενας νευρώνας σε ενα μεγαλύτερο εύρος τιμών σε σχέση με την σιγμοειδή συνάρτηση, καθώς η ελάχιστη τιμή της είναι το -1, σε σχέση με την σιγμοδειδή συνάρτηση που η ελάχιστη τιμή της είναι το 0.

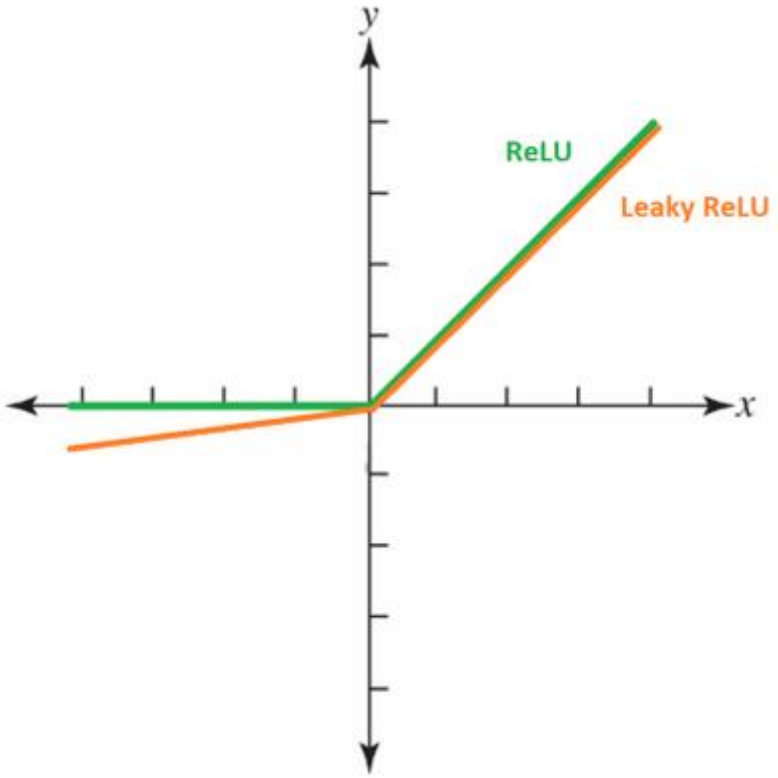


Εικόνα 7 - Συνάρτηση υπερβολικής εφαπτομένη



Εικόνα 8 - Συνάρτηση διορθωμένης γραμμικής μονάδας (ReLU)

Οι πρώτες δυο συναρτήσεις ενεργοποίησης έχουν το μειονέκτημα οτι δε μπορούν να αποφύγουν ενα βασικό πρόβλημα που προκύπτει στην εκπαίδευση των νευρωνικών δικτύων: το πρόβλημα των εξαφανιζόμενων κλίσεων (το οποίο θα εξετάσουμε σε επόμενη ενότητα). Η συνάρτηση διορθωμένης γραμμικής μονάδας ή ReLU της εικόνας 8 διορθώνει σε αρκετό βαθμό το πρόβλημα αυτό και για αυτό προτιμάται έναντι των άλλων δυο. Ωστόσο έχει ενα μειονέκτημα. Όταν οι τιμές ενεργοποίησης είναι μικρότερες του μηδενός ($x < 0$), τότε η κλίση της συνάρτησης είναι μηδέν. Αυτό έχει ως αποτέλεσμα κάποιοι κόμβοι να μην ενεργοποιούνται κατά την εκπαίδευση του δικτύου και τα βάρη να μην ενημερώνονται για αυτές τις μονάδες. Μια παραλλαγή της συνάρτησης ReLU είναι η Leaky ReLU, η οποία προσθέτει μια μικρή κλίση στο εύρως τον αρνητικών τιμών. Αυτό έχει ως αποτέλεσμα όταν τα δεδομένα εισαγωγής έχουν μηδενικές τιμές, να παράγεται μια μικρή αρνητική τιμή αντί του μηδενός.



Εικόνα 9 - Συνάρτηση Leaky ReLU

Τέλος, η συνάρτηση softmax παράγει σαν αποτέλεσμα μια κατανομή πιθανοτήτων και δίνει τις πιθανότητες για όλες τις υπάρχουσες κλάσεις στις οποίες μπορεί να ανήκουν τα δεδομένα.

Περιορισμοί των Νευρωνικών Δικτύων MLP

Τα MLP απαντώνται σε μια πληθώρα εφαρμογών στο πεδίο της Μηχανικής Μάθησης και είναι μια αρχιτεκτονική η οποία έχει χρησιμοποιηθεί κατα κόρον για την επίλυση διαφόρων προβλημάτων. Βρίσκονται στο επίκεντρο της έρευνας γύρω από την Τεχνητή Νοημοσύνη και είναι μια θεμελιώδης αρχιτεκτονική νευρωνικών δικτύων. Παρόλο που έχουν καταφέρει σημαντικές επιτυχίες σε προβλήματα όπως είναι η αναγνώριση αντικειμένων, δεν έχουν αντίστοιχα αποτελέσματα σε προβλήματα που θέτουν ερωτήματα γύρω από τη μοντελοποίηση ακολουθιών. Στην συνέχεια θα εξετάσουμε μοντέλα νευρωνικών δικτύων, τα οποία έχουν σαν κορμό την αρχιτεκτονική των MLP, ωστόσο επεκτείνουν τις δυνατότητες των κλασσικών μοντέλων, κάνοντας χρήση συνδέσεων ανατροφοδότησης.

Νευρωνικά Δίκτυα και Μοντελοποίηση Ακολουθιών

Στο προηγούμενο κεφάλαιο έγινε μια εισαγωγή στα νευρωνικά δίκτυα και στην βασική ορολογία που συναντάται στην βιβλιογραφία γύρω από αυτά. Έγινε μια επισκόπηση της αρχιτεκτονικής που αποτελεί την βάση των μοντέλων της βαθιάς μάθησης: του πολυεπίπεδου Perceptron (MPL). Είδαμε συνοπτικά την διαδικασία εκπαίδευσης των πολυεπίπεδων νευρωνικών δικτύων και καταλήξαμε στην παρατήρηση οτι τα μοντέλα αυτά δεν παρέχουν τις βέλτιστες λύσεις για προβλήματα που έχουν ως στόχο τη μοντελοποίηση ακολουθιών. Στη συνέχεια του παρόντος κεφαλαίου, θα εξετάσουμε τα μοντέλα που απασχολούν την σύγχρονη επιστημονική έρευνα γύρω από το πεδίο της μοντελοποίησης ακολουθιων: τα **Αναδρομικά Νευρωνικά Δίκτυα (RNN)**.

Αδυναμίες των Πολυεπίπεδων Perceptron στη Μοντελοποίηση Ακολουθιών

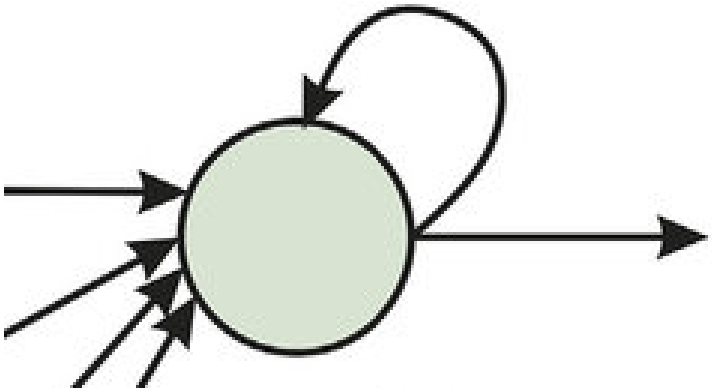
Δεδομένα Ακολουθιών

Ως δεδομένα ακολουθιών ορίζονται τα δεδομένα τα οποία τα συνδέει μια σχέση αλληλεξάρτησης. Δηλαδή στοιχεία ενός συνόλου δεδομένων εξαρτώνται άμεσα από στοιχεία του ίδιου συνόλου δεδομένων. Παραδείγματα τέτοιων συνόλων δεδομένων είναι τα δεδομένα από μετεωρολογικές παρατηρήσεις όπως η θερμοκρασία, τα δεδομένα ενός σήματος ήχου, δεδομένα χρονικών σειρών, δεδομένα ενός βίντεο, δεδομένα της αλληλουχίας του DNA κτλ. Οι τιμές των παρατηρήσεων (των στοιχείων του συνόλου) από σύνολα δεδομένων όπως τα προαναφερθέντα, έχουν άμεση εξάρτηση από τις τιμές των παρατηρήσεων που έχουν προηγηθεί του εκάστοτε στοιχείου. Παραδείγματος χάριν, η πρόβλεψη της τιμής της θερμοκρασίας της επόμενης ώρας σχετίζεται άμεσα με την τιμή της θερμοκρασίας που υπάρχει την δεδομένη στιγμή που μετράμε την θερμοκρασία. Αυτό σημαίνει οτι αν μετρήσουμε την θερμοκρασία σε μια δεδομένη χρονική στιγμή και βρούμε οτι έχει τιμή ίση με 15°, τότε η θερμοκρασία στην επόμενη ώρα θα κυμαίνεται γύρω από αυτή την θερμοκρασία και δεν θα πέσει ξαφνικά στους 0°. Βλέπουμε, επομένως, οτι τα δεδομένα ακολουθιών χαρακτηρίζονται και από την χρονική σειρά με την οποία υπάρχουν μέσα στο σύνολο δεδομένων. Για τη μοντελοποίηση τέτοιων δεδομένων θα χρειαστούμε εξειδικευμένα μοντέλα που θα συμπεριλαμβάνουν στους υπολογισμούς τους τέτοιου είδους συσχετίσεις. Στην επόμενη ενότητα θα εξετάσουμε το λόγο για τον οποίο τα κοινά πολυεπίπεδα νευρωνικά δίκτυα εμπρόσθιας τροφοδότησης υστερούν στη μοντελοποίηση των δεδομένων ακολουθιών.

MLP Αρχιτεκτονικές και Δεδομένα Ακολουθιών

Η κλασσική αρχιτεκτονική των πολυεπίπεδων νευρωνικών δικτύων εμπρόσθιας τροφοδότησης (deep feed forward neural networks) που είδαμε στο προηγούμενο κεφάλαιο, προωθεί τα αποτελέσματα κάθε κόμβου ενός επιπέδου, πρός τους κόμβους των επόμενων επιπέδων, χωρίς να υπάρχουν συνδέσεις από κόμβους που βρίσκονται σε επιίπεδα πιο βαθιά στο δίκτυο, πρός κόμβους που βρίσκονται σε προγενέστερα επίπεδα. Αυτό συνεπάγεται πως η γνώση που αποκτάται κατα την διαδικασία της εκπαίδευσης, δηλαδή οι τιμές που εξάγουν οι κόμβοι κάθε επιπέδου, δεν έχουν τρόπο να μεταδοθούν πίσω στο δίκτυο και να συμβάλλουν στην συνολική εκπαίδευση του δικτύου. Αυτό φαίνεται και στην Εικόνα 2 του προηγούμενου

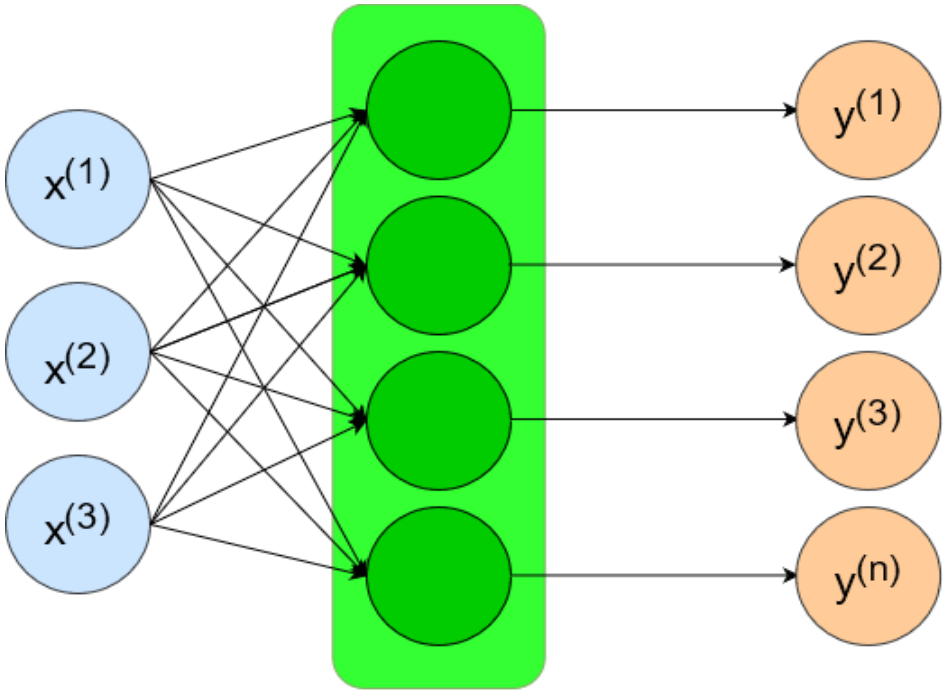
κεφαλαίου, όπου τα βέλη μεταξύ των επιπέδων δείχνουν την κατεύθυνση της πληροφορίας που παράγεται. Ένας κόμβος του δικτύου, με κατεύθυνση του αποτελέσματος πρός τον ίδιο τον κόμβο, έχει στο μοντέλο του ενα βέλος που δείχνει πως τα δεδομένα εξαγωγής της συνάρτησης του νευρώνα, ανατροφοδοτούνται πίσω στον ίδιο τον νευρώνα. Η παρακάτω εικόνα αναπαριστά εναν τέτοιο κόμβο.



Εικόνα 10 - Νευρώνας που κατευθύνει τη γνώση και στον εαυτό του

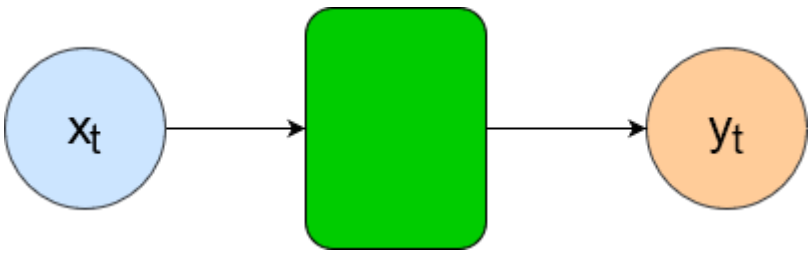
Η ανθρώπινη σκέψη έχει την δυνατότητα να συνδυάζει εμπειρίες και γνώσεις που έχουν αποκτηθεί στο παρελθόν, με πληροφορίες που αποκτούνται σε μεταγενέστερες στιγμές. Όπως όταν διαβάζουμε ενα βιβλίο και η κατανόηση μας προκύπτει από κάθε προηγούμενη λέξη, έτσι και η σκέψη μας χρησιμοποιεί τις γνώσεις που έχουμε αποκτήσει κατα την διάρκεια της ζωής μας. Έχουμε, δηλαδή, σαν άτομα την ιδιότητα διατηρούμε τις εμπειρίες και τις γνώσεις μας στη μνήμη μας. Τα κλασσικά νευρωνικά δίκτυα εμπρόσθιας τροφοδότησης έχουν το μειονέκτημα οτι δεν έχουν την δυνατότητα, λόγω της αρχιτεκτονικής τους, να διατηρήσουν την γνώση που αποκτούν σε μια αντίστοιχη λειτουργία όπως αυτή της ανθρώπινης μνήμης.

Για να δούμε πιο αναλυτικά τον παραπάνω περιορισμό των πολυεπίπεδων νευρωνικών δικτύων εμπορόσθιας τροφοδότησης, θα χρειαστεί να εξετάσουμε ξανά την αρχιτεκτονική του. Ας θεωρήσουμε ενα τέτοιο δίκτυο όπως στην παρακάτω εικόνα:



Εικόνα 11 – Ενα MLP με τρείς κόμβους εισαγωγής δεδομένων και τέσσερις εξαγωγής δεδομένων

Το απεικονιζόμενο δίκτυο αποτελείται από το επίπεδο εισαγωγής του στο οποίο υπάρχουν τρείς κόμβοι εισαγωγής δεδομένων, το επίπεδο εξαγωγής δεδομένων στο οποίο υπάρχουν τέσσερις κόμβοι και ενα κρυφό επίπεδο νευρώνων Perceptron. Για λόγους διευκόλυνσης στην ανάλυσή μας θα απλοποιήσουμε την παραπάνω εικόνα ως εξής:



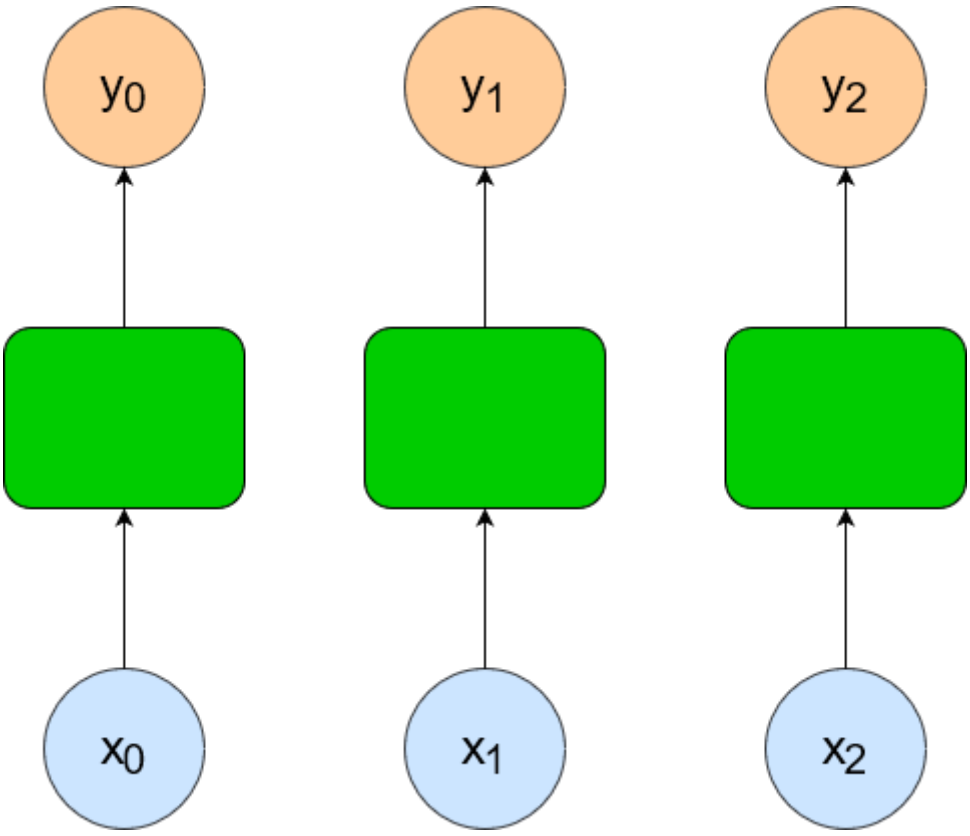
Εικόνα 12- Απλοποιημένο MLP

Στην εικόνα 12 έχουμε συμπτίξει τα επίπεδα εισαγωγής και εξαγωγής σε ξεχωριστούς κόμβους οι οποίοι αναπαρίστανται από τα διανύσματα

$$\boldsymbol{x}_t \in \mathbb{R}^{\boldsymbol{m}}$$
$$\boldsymbol{y}_t \in \mathbb{R}^{\boldsymbol{n}}$$

Όπου $\boldsymbol{x}_t$ και $\boldsymbol{y}_t$ είναι διανύσματα μήκους $\boldsymbol{m}$ και $\boldsymbol{n}$ αντίστοιχα, τα οποία εισάγονται στο δίκτυο την χρονική στιγμή t.

Αν θέλουμε να εισάγουμε δεδομένα ακολουθίας σε ενα MLP, θα μπορούσαμε να χρησιμοποιήσουμε το ίδιο ακριβώς δίκτυο, δημιουργώντας νεα στιγμιότυπά του, τα οποία θα λαμβάνουν διανύσματα δεδομένων από την ακολουθία για κάθε διαφορετικό χρονικό βήμα (**timestep**).



Εικόνα 13 - Μοντελοποίηση Δεδομένων Ακολουθίας με MLP

Στην παραπάνω εικόνα βλέπουμε την αναπαράσταση της μοντελοποίησης μιας ακολουθίας με MLP. Όπως αναφέραμε σε προηγούμενη ενότητα, τα δεδομένα μιας ακολουθίας έχουν το χαρακτηριστικό οτι υπάρχει μια χρονική αλληλεξάρτηση (**temporal interdependence**) μεταξύ τους. Η κάθε τιμή που υπολογίζει το δίκτυο έχει άμεση εξάρτηση από προηγούμενους υπολογισμούς . Αυτό σημαίνει οτι, στο μοντέλο της εικόνας 13, ο κόμβος y_2 εξαρτάται από τους προηγούμενους κόμβους y_0 και y_1.

Ενα τέτοιο μοντέλο, παρόλο που μπορεί να παράγει αποτελέσματα για κάθε χρονικό βήμα της ακολουθίας, δεν είναι ικανό να μοντελοποιήσει σωστά μια ακολουθία. Η αδυναμία αυτή του μοντέλου έγκειται στο γεγονός οτι δεν υπάρχει κάποιου είδους σύνδεση μεταξύ των διαφορετικών στιγμιοτύπων του δικτύου. Χωρίς αυτή τη σύνδεση, το μοντέλο δεν έχει την δυνατότητα να λάβει υπόψιν του τις προβλέψεις που έγιναν σε προηγούμενα επίπεδα, καθώς εξετάζει το κάθε σύνολο δεδομένων ενός χρονικού βήματος μεμονωμένα. Αν και μπορούμε να επαναλαμβάνουμε τους υπολογισμούς, για κάθε χρονικό βήμα της ακολουθίας, με το ίδιο ακριβώς μοντέλο, εισάγοντας στο δίκτυο τα δεδομένα του εκάστοστε χρονικού βήματος, το μοντέλο δεν θα μπορέσει να μάθει την οποιαδήποτε σχέση εξάρτησης υπάρχει μεταξύ αυτών.

Με βάση τα παραπάνω, γίνεται αντιληπτό οτι, για τη μοντελοποίηση ακολουθιών, υπάρχει η ανάγκη για την χρήση ενός διαφορετικού μοντέλου από αυτό των πολυεπίπεδων νευρωνικών δικτύων εμπρόσθιας τροφοδότησης, καθώς εξ'ορισμού δεν συμπεριλαμβάνουν στην αρχιτεκτονική τους κάποιου είδους μνήμη, στην οποία υπάρχει όλη η γνώση που έχει αποκτήσει το δίκτυο μέχρι κάποιο χρονικό βήμα t. Στην παρακάτω ενότητα θα εξετάσουμε τα Αναδρομικά Νευρωνικά Δίκτυα, τα οποία δημιουργήθηκαν για την επίλυση τέτοιων προβλημάτων.

Αναδρομικά Νευρωνικά Δίκτυα (Recurrent Neural Networks - RNN)

Όπως είδαμε στην προηγούμενη ενότητα, τα κλασικά νευρωνικά δίκτυα υπολείπονται της δυνατότητας να μοντελοποιήσουν δεδομένα ακολουθιών λόγω του οτι η αρχιτεκτονική τους δεν παρέχει εναν τρόπο στο δίκτυο να χρησιμοποιεί τους υπολογισμούς κάθε χρονικού βήματος της ακολουθίας, παρά μόνο εξετάζει τα δεδομένα της ακολουθίας μεμονωμένα. Το κενό αυτό έρχεται να καλύψει μια ειδική κατηγορία νευρωνικών δικτύων, τα αναδρομικά νευρωνικά δικτύα, τα οποία προτάθηκαν το 1986 από τον Rumelhart και τα οποία αποτελούν ενα σημαντικό μοντέλο στη μοντελοποίηση ακολουθιών.

Αρχιτεκτονική Αναδρομικών Νευρωνικών Δικτύων

Ένα κλασικό πρόβλημα, για το οποίο επιστρατεύονται τα νευρωνικά δίκτυα, βρίσκεται στο πεδίο της Επεξεργασίας της Φυσικής Γλώσσας (**Natural Language Processing - NLP**). Η φυσική γλώσσα είναι ενα πολύ χαρακτηριστικό παράδειγμα δεδομένων ακολουθίας, καθώς οι λέξεις σε μια πρόταση αποτελούν στοιχεία μιας ακολουθίας. Το πρόβλημα πρός επίλυση είναι η πρόβλεψη της επόμενης λέξης μια πρότασης. Ας πάρουμε σαν παράδειγμα τις παρακάτω προτάσεις για τις οποίες θέλουμε το μοντέλο μας να προβλέψει την τελευταία λέξη:

1. *«Η δουλειά μου προκαλέι κούραση»*
2. *«Όταν είμαι κουρασμένος θέλω να ξεκουραστώ»*
3. *«Ήμουν πολύ κουρασμένος αλλά έπρεπε να τελειώσω το καθάρισμα πριν να ξεκουραστώ»*

Παρατηρώντας αυτές τις προτάσεις θα δούμε οτι η καθεμία από αυτές έχει διαφορετικό μήκος, διαφορετικό αριθμό λέξεων. Σε ενα νευρωνικό δίκτυο μοντελοποίησης δεδομένων ακολουθιών, θέλουμε να μπορούμε να διαχειριζόμαστε ακολουθίες διαφορετικών μηκών (**Variable Sequence Length**). Τα κλασσικά νευρωνικά δίκτυα εμπρόσθιας τροφοδότησης αδυνατούν να διαχειριστούν τέτοιου είδους καταστάσεις, καθώς στο επίπεδο εισαγωγής τους δέχονται διανύσματα σταθερών διαστάσεων.

Ένα αλλο σημείο που θέλουμε να διαχειριστούμε κατά τη μοντελοποίηση ακολουθιών, είναι η σείρά (**Sequence Order**) με την οποία υπάρχουν τα δεδομένα στην ακολουθία. Όπως είπαμε σε προηγούμενη ενότητα, τα στοιχεία μιας ακολουθίας χαρακτηρίζονται από την ιδιότητα της σειράς με την οποία υπάρχουνε μέσα στην ακολουθία. Συνεχίζοντας με ενα παράδειγμα πάλι από το πεδίο της Επεξεργασίας της Φυσική Γλώσσας ας δούμε τις παρακάτω προτάσεις:

1. *«Η συναυλία ήταν αρκετά καλή, καθόλου ενοχλητική»*
2. *«Η συναυλία ήταν αρκετά ενοχλητική, καθόλου καλή»*

Οι παραπάνω προτάσεις αποτελούνται από το ίδιο πλήθος λέξεων και από τις ίδιες ακριβώς λέξεις. Είναι εμφανές οτι η σειρά που παρουσιάζονται τα στοιχεία της ακολουθίας, η λέξεις δηλαδή της κάθε πρότασης, διαφοροποιούν πλήρως το νόημα των προτάσεων, παρόλο που και στις δυο προτάσεις χρησιμοποιούνται ακριβώς οι ίδιες λέξεις. Η πρώτη πρόταση δίνει την αίσθησης αποκόμησης ευχαρίστησης από την συναυλία, ενώ η δεύτερη πρόταση δίνει ενα αρνητικό συναίσθημα. Είναι σημαντικό, επομένως , τα μοντέλα που καλούνται να μοντελοποιήσουν ακολουθίες, να μπορούν να εξάγουν αποτελέσματα στα οποία να υπάρχει νοήμα στην σειρά των δεδομένων.

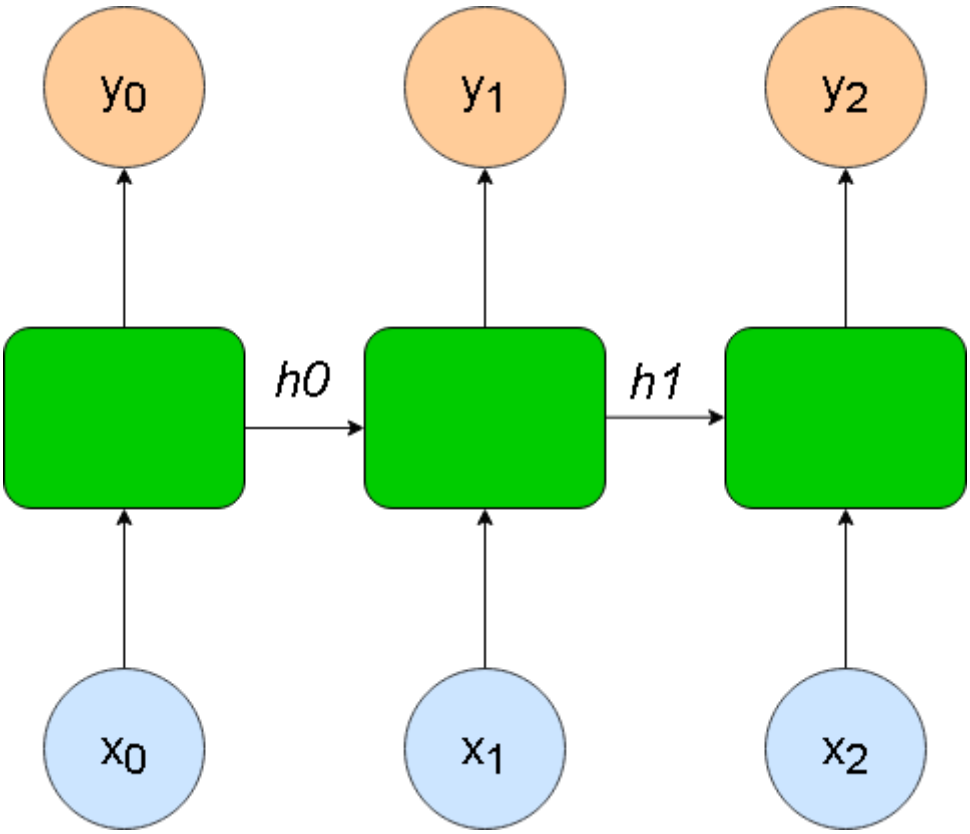
Μια ακόμα δυνατότητα που θα πρέπει να έχουν αυτά τα μοντέλα, είναι να μπορούν να μοντελοποιούν μακροχρόνιες εξαρτήσεις (**Long Term Dependencies**) ανάμεσα στα στοιχεία της ακολουθίας. Ας δούμε πάλι ενα παράδειγμα της φυσικής γλώσσας για να κατανοήσουμε καλύτερα τις μακροχρόνιες εξαρτήσεις σε μια ακολουθία:

«Μεγάλωσα στην **Ελλάδα**, αλλά τώρα μένω στην Ολλανδία. Μιλάω άπταιστα ____.»

Είναι φανερό πώς η τελευταία λέξη στην πρόταση είναι η «Ελληνικά». Για έναν άνθρωπο, το συμπέρασμα αυτό είναι απολύτως λογικό και αυτονόητο. Ενα νευρωνικό δίκτυο όμως θα πρέπει να είναι σε θέση να *θυμάται* τέτοιες μακροχρόνιες εξαρτήσεις ανάμεσα σε δεδομένα της ίδιας πρότασης, ώστε να μπορεί να προβλέπει με μεγαλύτερη ακρίβεια το τελικό αποτέλεσμα.

Επεκτείνοντας τώρα το μοντέλο του νευρωνικού δικτύου εμπρόσθιας τροφοδότησης της προηγούμενης ενότητας, ας δούμε πώς η αρχιτεκτονική των αναδρομικών νευρωνικών δικτύων, τα οποία έχουν ως βάση τα MLP, βελτιώνει την δυνατότητα των δικτύων να μοντελοποιούν ακολουθίες. Το κύριο χαρακτηριστικό που θέλουμε να συμπεριλάβουμε σε αυτή τη μοντελοποίηση, είναι συσχέτιση που έχουν μεταξύ τους τα δεδομένα της ακολουθίας για κάθε χρονικό βήμα που εισάγουμε στον μοντέλο μας. Είδαμε πως το σχεδιάγραμμα του μοντέλου της εικόνας 13 αδυνατεί να συμπεριλάβει τέτοιες συσχετίσεις. Η ειδοποιός διαφορά των αναδρομικών νευρωνικών δικτύων από τις κλασσικές αρχιτεκτονικές, είναι η εισαγωγή της έννοιας της «μνήμης» στο μοντέλο. Για να μπορέσουν τα αναδρομικά νευρωνικά δίκτυα να συσχετίσουν τους υπολογισμούς παρελθοντικών χρονικών βημάτων της ακολουθίας με τους υπολογισμούς που λαμβάνουν χώρα σε επομενα χρονικά βήματα, χρησιμοποιούν μια εσωτερική κατάσταση (**Internal State**) η οποία ουσιαστικά παίζει τον ρόλο της μνήμης του νευρωνικού δικτύου και διατηρεί την πληροφορία από τους υπολογισμούς παρελθοντικών

χρονικών βημάτων. Η εσωτερική κατάσταση ουσιαστικά είναι όλη η πληροφορία που εξάγει ο κόμβος ενός δικτύου στο εκάστοτε χρονικό βήμα. Έπειτα, η πληροφορία αυτή προωθείται μέσω της εσωτερικής κατάστασης/μνήμης σε μεταγενέστερους κόμβος οι οποίοι την λαμβάνουν πλέον σαν δεδομένο εισαγωγής. Η παρουσία των συνδέσεων μεταξύ των χρονικών βημάτων του νευρωνικο μέσω της εσωτερικής κατάστασης μετατρέπει το δίκτυο σε αναδρομικό νευρωνικό δίκτυο, το οποίο πλέον αποτελείται από αναδρομικά επίπεδα όπως θα δούμε στην συνέχεια. Στην παρακάτω εικόνα μπορούμε να δούμε την έννοια της μνήμης.



Εικόνα 14 - Εσωτερική μνήμη νευρωνικού δικτυου

Εισάγοντας την έννοια της εσωτερικής μνήμης στο δίκτυο της παραπάνω εικόνας, βλέπουμε πλέον την άμεση συσχέτιση των δεδομένων εξαγωγής των εκάστοτε χρονικών βημάτων με προηγούμενους υπολογισμούς του δικτύου μέσω της εσωτερικής μνήμης. Στο κλασσικό MLP μοντέλο η συνάρτηση που παράγει τα δεδομένα εξαγωγής είναι η

$$\widehat{y_t} = f(x_t) \tag{2.1}$$

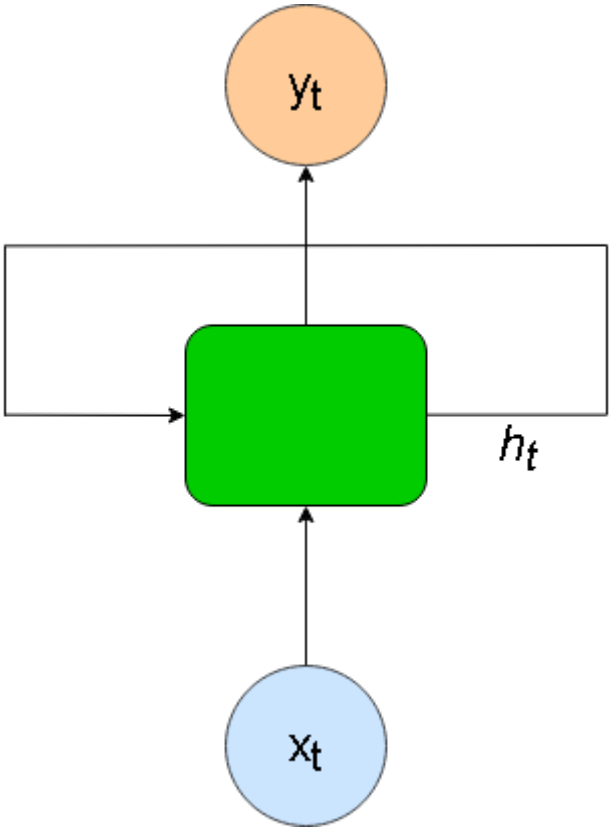
Αν στην συνάρτηση (2.1) συμπεριλάβουμε και τον όρο της εσωτερικής μνήμης, τότε η συνάρτηση διαμορφώνεται ως εξής:

$$\widehat{y_t} = f(x_t, h_{t-1}) \tag{2.2}$$

Η συνάρτηση (2.2) δείχνει πώς οι υπολογισμοί κάθε χρονικού βήματος της ακολουθίας, εξάρτώνται όχι μόνο από τα δεδομένα εισαγωγής στο συγκεκριμένο χρονικό βήμα, αλλά ταυτόχρονα και από την εσωτερική κατάσταση/μνήμη που διατηρεί το δίκτυο μέχρι το προηγούμενο χρονικό βήμα, η οποία έχει αποθηκεύσει πληροφορία από προηγούμενα δεδομένα εισαγωγής αλλά και από προηγούμενα δεδομένα εξαγωγής. Ο όρος h_{t-1} κατέχει πληροφορία σχετικά με όλη την γνώση που έχει λάβει το νευρωνικό δίκτυο μέχρι το χρονικό βήμα t. Διατηρεί δηλαδή όλη την ιστορικότητα του δικτυου, ιδιότητα η οποία διαφοροποιεί τα αναδρομικά νευρωνικά δίκτυα από τα κλασικά MLP και τα χαρακτηρίζει ως κατάλληλα για την επεξεργασία δεδομένων ακολουθίας.

Το νευρωνικό δίκτυο της εικόνας 14 αποτελεί το «ξετυλιγμένο» υπολογιστικό γράφημα (**Unrolled Computational Graph**) ενός αναδρομικού νευρωνικού δικτύου το οποίο το έχουμε επεκτείνει στην διάσταση του χρόνου. Το μέγεθος ενός τέτοιου γραφήματος εξαρτάται άμεσα από το συνολικό μήκος της ακολουθίας πρός επεξεργασία. Για παράδειγμα, αν είχαμε μια ακολουθία τριών λέξεων, θα μπορούσαμε να χρησιμοποιήσουμε το μοντέλο της εικόνας 14 για να μοντελοποιήσουμε την ακολουθία, όπου ο το κάθε επίπεδο του δικτύου θα αντιστοιχούσε σε κάθε μια από τις λέξεις της ακολουθίας. Ενα επίπεδο για κάθε λέξη. Το δομικό στοιχείο του γραφήματος είναι μια επαναλαμβανόμενη δομή νευρώνων που αντιστοιχούν σε μια αλυσίδα από γεγονότα ή υπολογισμούς.

Ένας άλλος τρόπος να παρουσιάσουμε το γράφημα της εικόνας 14, είναι να χρησιμοποιήσουμε τον όρο h_{t-1} στην σχεδίαση του δικτύου. Ο όρος h_{t-1} μας βοηθάει να εκφράσουμε το γράφημα αυτό με όρους αναδρομής, διατηρώντας τις υπαρχουσες σχέσεις, εισάγοντας μια κυκλική σχέση στον νευρώνα, όπως φαίνεται στην παρακάτω εικόνα:

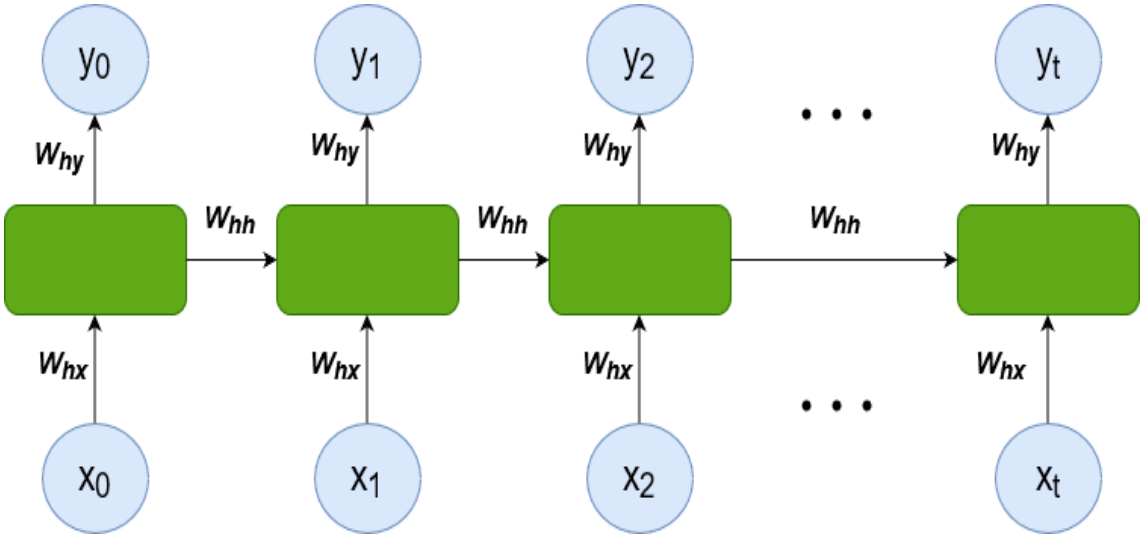


Εικόνα 15 - Νευρώνας με αναδρομή

Το γράφημα της παραπάνω εικόνας απλοποιεί σημαντικά την αναπαράσταση ενός αναδρομικού νευρωνικού δικτύου, καθώς είναι ενα πιο περιεκτικό γράφημα και παρουσιάζει την γενική ιδέα γύρω από τα RNN. Το «ξετυλιγμένο» υπολογιστικό γράφημα μπορεί να μας δώσει περισσότερες πληροφορίες σχετικά με την εσωτερική λειτουργία του δικτύου, καθώς μπορούμε να προσθέσουμε παραπάνω έννοιες σχετικές με την διαδικασία με την οποία γίνονται οι υπολογισμοί σε κάθε νευρώνα, άμεσα πάνω στο γράφημα, όπως φαίνεται στην εικόνα 16.

Με ενα τέτοιο «ξετυλιγμένο» γράφημα, μπορούμε να παρουσιάσουμε και την ιδέα της κοινής χρήσης παραμέτρων (**Parameter Sharing**) στο σύνολο της δομής του νευρωνικού δικτύου. Όπως φαίνεται και στην εικόνα 16, το δίκτυο χρησιμοποιεί σε κάθε χρονικό βήμα τις ίδιες παραμέτρους. Κάθε στιγμιότυπο χρησιμοποεί τις ίδιες μεταβλητές βαρών, $\mathbf{W_{hx}}$, $\mathbf{W_{hy}}$, $\mathbf{W_{hh}}$ κατά την διάρκεια των υπολογισμών. Αν για κάθε χρονικό βήμα είχαμε διαφορετικές παραμέτρους, δε θα μπορούσαμε να γενικεύσουμε την χρήση του μοντέλου σε διαφορετικά μήκη ακολουθιών, τα οποία δεν έχει ξανασυναντήσει (δηλαδή κατά την διάρκεια της εκπαίδευσης). Κάνοντας χρήση κοινών παραμέτρων σε όλη την έκταση του μοντέλου, μπορούμε να μοντελοποιήσουμε ακολουθίες διαφορετικών μηκών. Ένα κλασικό μοντέλο MLP θα είχε διαφορετικές παραμέτρους για κάθε στοιχείο της ακολουθίας και αυτό θα είχε ως αποτέλεσμα να μοντελοποιεί ξεχωρίστα αυτά τα στοιχεία. Οι RNN αρχιτεκτονικές μπορούν να χρησιμοποιούν τις ίδιες παραμέτρους καθώς τα δεδομένα εξαγωγής σε κάθε στιγμιότυπο είναι συνάρτηση των προηγούμενων

δεδομένων εξαγωγής και οι κανόνες ενημέρωσης των παραμέτρων είναι κοινές σε κάθε βήμα. Μέσα από αυτή την ιδιότητα των αναδρομικών νευρωνικών δικτύων ουσιαστικά μας δίνεται η δυνατότητα να εκπαιδεύσουμε ενα κοινό μοντέλο, το οποίο μπορεί να χρησιμοποιηθεί σε νέες ακολουθίες και, επίσης, μας δίνεται η δυνατότητα να εκπαιδεύσουμε το δίκτυο με αρκετά λιγότερα παραδείγματα εκπαίδευσης, σε σχέση με ενα μοντέλο που χρειάζεται διαφορετικές παραμέτρους για κάθε καινούριο στοιχείο που του εισάγουμε.



Εικόνα 16- Ξετυλιγμένο γράφημα με πληροφορίες σχετικές με τους εσωτερικούς υπολογισμούς

Όπως αναφέραμε και προηγουμένως, σε κάθε χρονικό βήμα της ακολουθίας, ορίζεται η εσωτερική κατάσταση $\mathbf{h}^{(t)}$ η οποία ερμηνεύται ως η ιστορικότητα του δικτύου μέχρι το τρέχων χρονικό βήμα. Η εξίσωση (2.3) είναι η γενική εξίσωση που ορίζει το πώς υπολογίζεται η εσωτερική κατάσταση σε κάθε χρονικό βήμα.

$$\boldsymbol{h}^{t} = f\left(\boldsymbol{h}^{(t-1)}, \boldsymbol{x}^{(t)}; \boldsymbol{\theta}\right) \qquad (2.3)$$

Η εξίσωση (2.3) μας δείχνει πως η εσωτερική κατάσταση κάθε χρονικού βήματος ορίζεται ως μια συνάρτηση της εσωτερικής κατάστασης του προηγούμενο χρονικού βήματος $\boldsymbol{h}^{(t-1)}$, των δεδομένων εισαγωγής του τρέχοντος χρονικού βήματος $\boldsymbol{x}^{(t)}$, καθώς και των παραμέτρων του μοντέλου $\boldsymbol{\theta}$ που, όπως αναφέραμε, είναι ίδιες για όλα τα χρονικά βήματα της ακολουθίας.

Σε προηγούμενη παράγραφο, αναφέραμε πώς το ξετυλιγμένο γράφημα της εικόνας παρέχει επιπλέον πληροφορίες σχετικά με την λειτουργία του νευρωνικού δικτύου. Παρατηρώντας την εικόνα 16, θα δούμε πως, σε κάθε χρονικό βήμα, ο κάθε νευρώνας έχει τις ίδιες παραμέτρους βαρών με όλους τους υπόλοιπου νευρώνες, οι οποίες χρησιμοποιούνται στο πέρασμα του

χρόνου (στα διαδοχικά χρονικά βήματα της ακολουθίας) για την εκπαίδευση του νευρωνικού δικτύου. Αυτές οι παράμετροι είναι οι εξής:

- Στην σύνδεση μεταξύ του επιπέδου εισαγωγής δεδομένων και του κρυφού επιπέδου υπάρχει η μήτρα βαρών W_{hx}.
- Στην σύνδεση μεταξύ των κρυφών επιπέδων μεταξύ των γειτονικών νευρώνων υπάρχει η μήτρα βαρών W_{hh}.
- Στην σύνδεση μεταξύ του κρυφού επιπέδου και του επιπέδου εξαγωγής δεδομένων υπάρχει η μήτρα βαρών W_{hy}.

Κατά την διάρκεια εκπαίδευσης των αναδρομικών νευρωνικών δικτύων, όπως και στις κλασικές αρχιτεκτονικές MLP, η εύρεση των κατάλληλων συντελεστών βαρών που θα παράγουν τα πιο ακριβή αποτελέσματα είναι ο κύριος στόχος της εκπαίδευσης.

Πρίν προχωρήσουμε στην εκπαίδευση των αναδρομικών νευρωνικών δικτύων, θα παρουσιάσουμε μερικές από τις πιο διαδεδομένες αρχιτεκτονικές και θα αναφέρουμε κάποια παραδείγματα χρήσης τους που θα μας βοηθήσουν να αποκτήσουμε μια καλύτερη αίσθηση σχετικά με την λειτουργία τους.

Σημαντικές Αρχιτεκτονικές Αναδρομικών Νευρωνικών Δικτύων

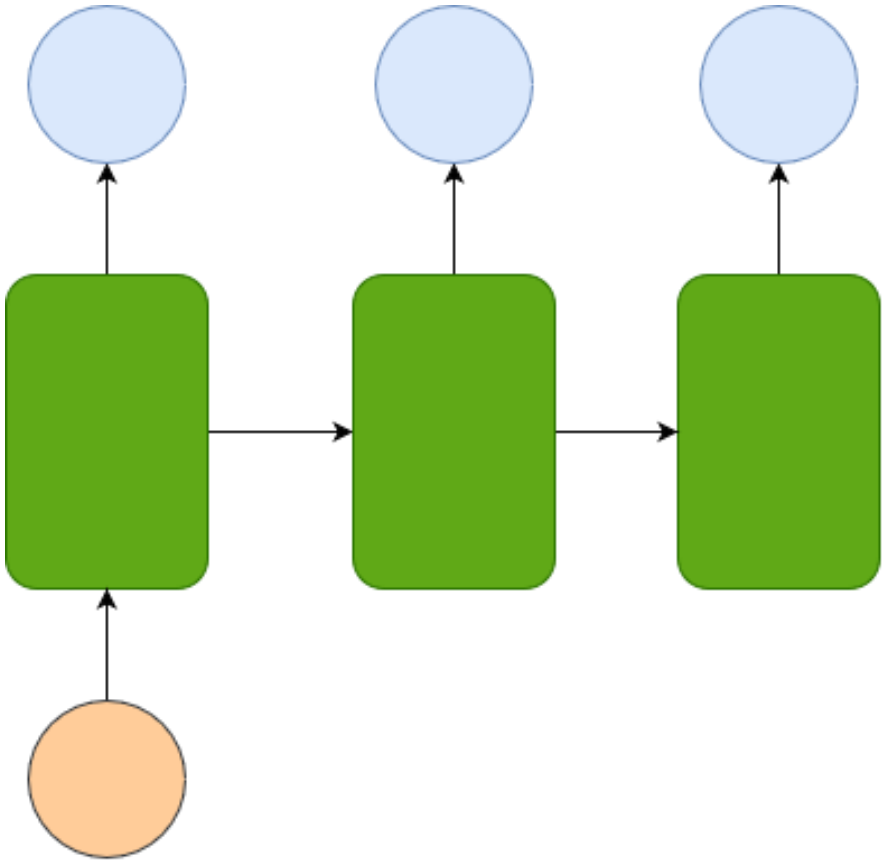
Έχοντας πλέον μια πρώτη εικόνα της βασικής αρχιτεκτονικής των αναδρομικών νευρωνικών δικτύων, μπορούμε να δούμε μερικά επιπλέον σχέδια δικτύων τα οποία βρίσκουν εφαρμογή σε διαφορετικά ερευνητικά πεδία. Συγκεκριμένα, θα αναφερθούμε στις εξής δομές:

- Αρχιτεκτονική ενα πρός πολλά
- Αρχιτεκτονική πολλά πρός ενα
- Αρχιτεκτονική πολλά πρός πολλά

Τα παραπάνω μοντέλα είναι παραδείγματα που φανερώνουν την διαφορά των MLP δικτύων, στα οποία η αρχιτεκτονική είναι ενα πρός ενα, δηλαδή δέχονται ως δεδομένα ενα σταθερού μεγέθους διάνυσμα και παράγουν ως αποτέλεσμα ενα άλλο διάνυσμα σταθερού μεγέθους. Επιπλέον, τα υπολογιστικά βήματα που μπορούν να εκτέλεσουν τα MLP είναι όσα και τα επίπεδα του μοντέλου.

Εικόνα 17 - Αρχιτεκτονική ενα πρός ενα MLP

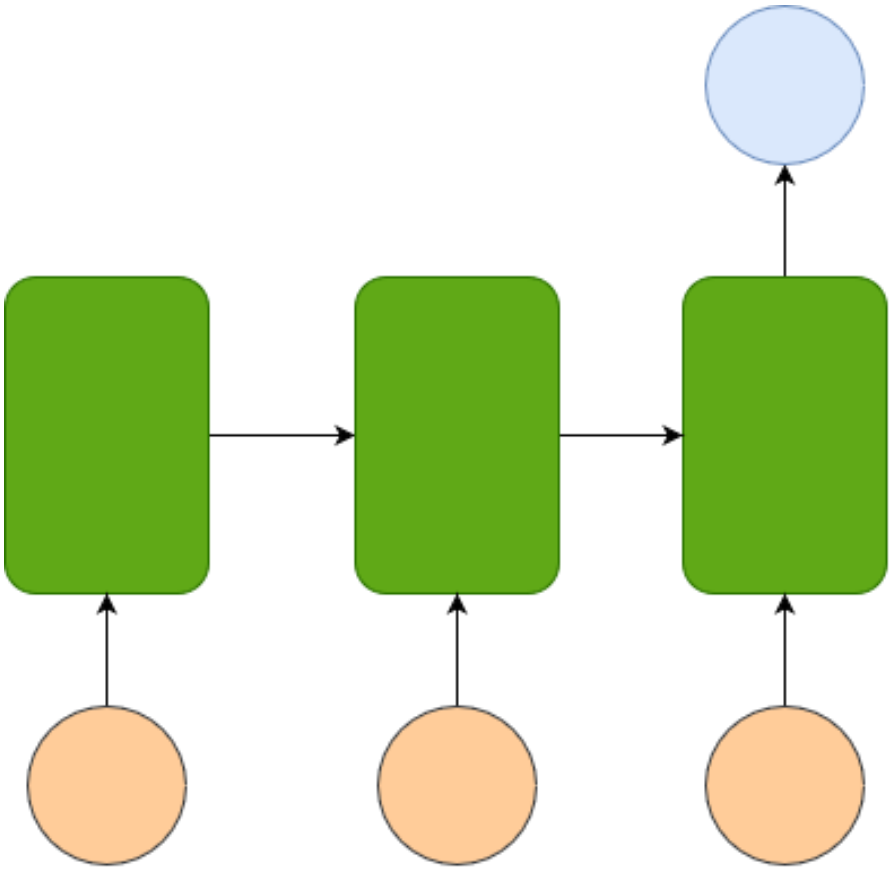
Αρχιτεκτονική ενα πρός πολλά



Εικόνα 18 - Αρχιτεκτονική ενα πρός πολλά

Το συγκεκριμένο μοντέλο δέχεται σαν είσοδο ενα διάνυσμα σταθερού μήκους και παράγει σαν έξοδο δεδομένα ακολουθίας. Παραδείγματα στα οποία μπορούν να χρησιμοποιηθούν τέτοιου είδους μοντέλα είναι περιπτώσεις όπως το να εισάγουμε στο δίκτυο μια εικόνα (η εικόνα έχει σταθερό μήκος σαν είσοδος) και να περιμένουμε ως αποτέλεσμα το δίκτυο να παράγει κάποια ακολουθία λέξεων η οποία θα περιγράφει την εικόνα που δέχτηκε στο επίπεδο εισαγωγής.

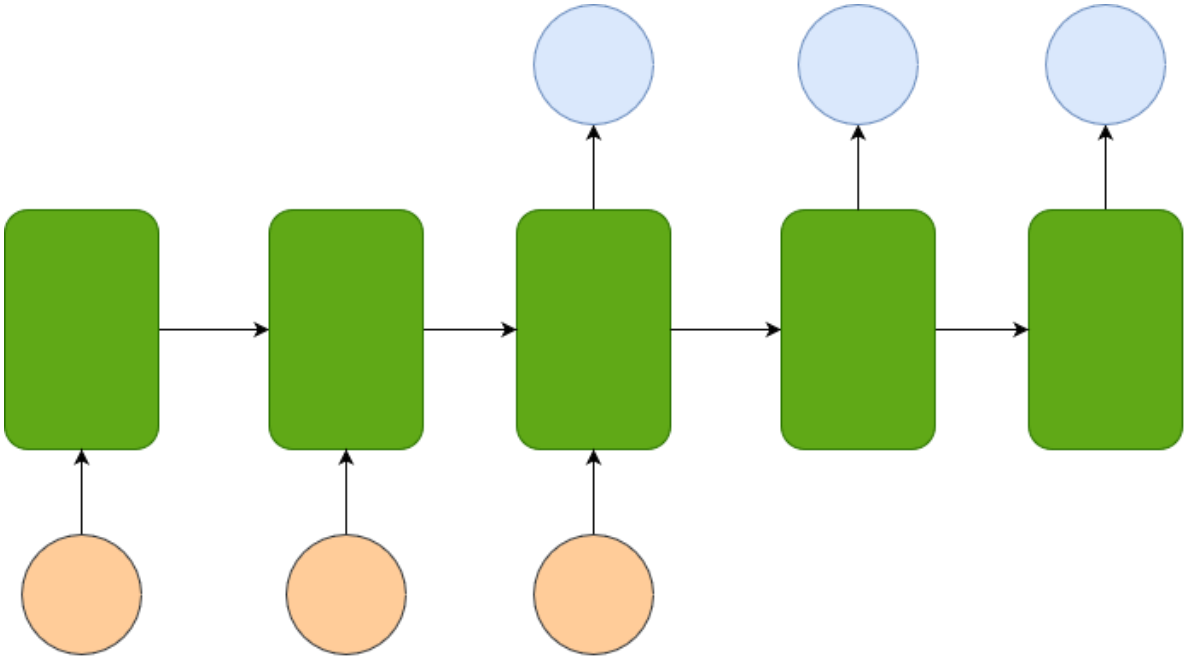
Αρχιτεκτονική πολλά πρός ενα



Εικόνα 19 - Αρχιτεκτονική πολλά πρός ενα

Το μοντέλο της παραπάνω εικόνας δέχεται σαν είσοδο μια ακολουθία δεδομένων και παράγει σαν αποτέλεσμα ενα διάνυσμα. Τέτοιου είδους νευρωνικά δίκτυα μπορούν να χρησιμοποιηθούν σε περιπτώσεις όπου εισάγουμε στο δίκτυο μια πρόταση, π.χ. «Η παράσταση ήταν απογοητευτική», και περιμένουμε σαν αποτέλεσμα την κατηγοριοποίηση της πρότασης ώς πρός το συναίσθημα που προκαλέι, δηλαδή αν είναι θετικό ή αρνητικό.

Αρχιτεκτονική πολλά πρός πολλά



Εικόνα 20 - Αρχιτεκτονική πολλά πρός πολλά

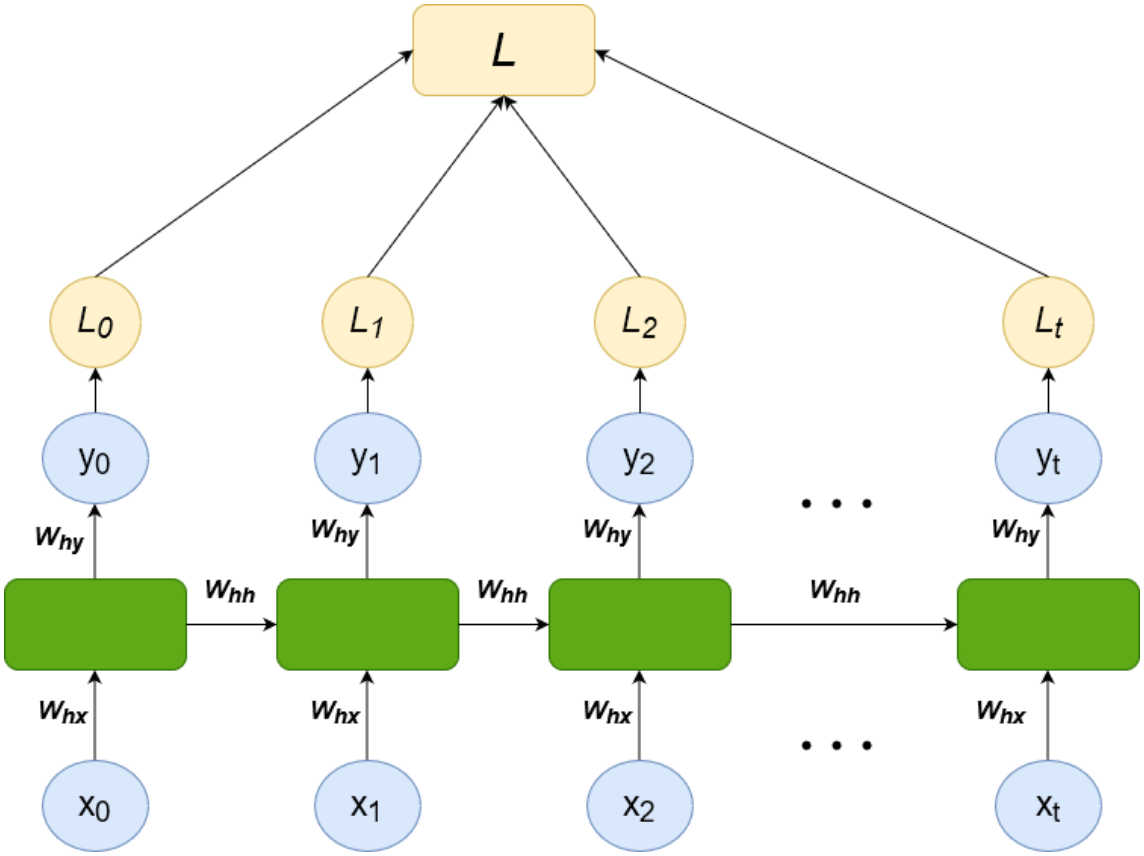
Σε μια δομή όπως της εικόνας 20, το μοντέλο δέχεται στα δεδομένα εισαγωγής μια ακολουθία και παράγει σαν αποτέλεσμα μια άλλη ακολουθία. Ενα παράδειγμα εφαρμογής που χρησιμοποιεί τέτοιες αρχιτεκτονικές είναι η μετάφραση μηχανής (**Machine Translation**) όπου εισάγουμε στο δίκτυο μια πρόταση στα ελληνικά και θέλουμε να πάρουμε σαν αποτέλεσμα την ίδια πρόταση μεταφρασμένη σε μια άλλη γλώσσα, όπως τα γερμανικά. Ενα τέτοιο μοντέλο θα μπορούσε επίσης να χρησιμοποιηθεί στην σύνθεση νεας μουσικής, οπου το δίκτυο δέχεται μια ακολουθία από νότες και παράγει μια νεα σειρά από νότες.

Στην επόμενη υποενότητα θα ασχοληθούμε με την εκπαίδευση των αναδρομικών νευρωνικών δικτύων και στο πώς διαφέρει από την εκπαίδευση των MLP δικτύων.

Εκπαίδευση Αναδρομικών Νευρωνικών Δικτύων

Ο τρόπος με τον οποίο εκπαιδεύεται ενα αναδρομικό νευρωνικό δίκτυο είναι παρόμοιος με τον τρόπο που παρουσιάσαμε σε προηγούμενο κεφάλαιο, όπου είδαμε την διαδικασία εκπαίδευσης ενός νευρωνικού δικτύου εμπρόσθιας τροφοδότησης. Είδαμε πως στόχος της εκπαίδευσης είναι η ελαχιστοποίηση της συνάρτης κόστους $\mathcal{L}$, η οποία επιτυγχάνεται με τον αλγόριθμο της αντίστροφης διάδοσης σφαλμάτων. Ο αλγόριθμος αυτός ενημερώνει τις παραμέτρους τους συστήματος με βάση την παράγωγο της συνάρτησης κόστους προωθώντας προς τα πίσω στο δίκτυο τη μέτρηση του λάθους. Σκοπός επομένως της εκπαίδευσης ενός αναδρομικού δικτύου είναι και πάλι η ελαχιστοποίηση μιας συνάρτησης κόστους με βάση τη μέτρηση σφάλματος και την παράγωγο της συνάρτησης αυτής.

Το πρώτο βήμα στην διαδικασία εκπαίδευσης είναι αυτό της εμπρόσθιας τροφοδοτησης. Στην εικόνα 21 βλέπουμε ενα παρόμοιο σχήμα με το νευρωνικό δίκτυο της εικόνας 16, ωστόσο τώρα έχουμε προσθέσει και την μέτρηση του σφάλματος, που είναι ουσιαστικά το άθροισμα όλων των επιμέρους αποτελεσμάτων των συναρτήσεων κόστους σε κάθε χρονικό βήμα της ακολουθίας.



Εικόνα 21 – Βήμα εμπρόσθιας τροφοδότησης σε αναδρομικό νευρωνικό δίκτυο

Σε κάθε επιμέρους χρονικό βήμα, οι υπολογισμοί που χρειάζεται να κάνει το δίκτυο είναι ο υπολογισμός του y και η ενημέρωση της εσωτερικής κατάστασης του δικτύου h_t. Για να υπολογίσουμε την εσωτερική κατάσταση του δικτύου χρησιμοποιούμε την εξίσωση (2.4).

$$h_t = \tanh\left(\boldsymbol{W_{hh}^T h_{t-1}} + \boldsymbol{W_{xh}^T x_t}\right) \qquad (2.4)$$

Στην εξίσωση (2.4) βλέπουμε πως η τιμή τρέχουσα κατάσταση του δικτύου είναι το άθροισμα των πολλαπλασιασμών

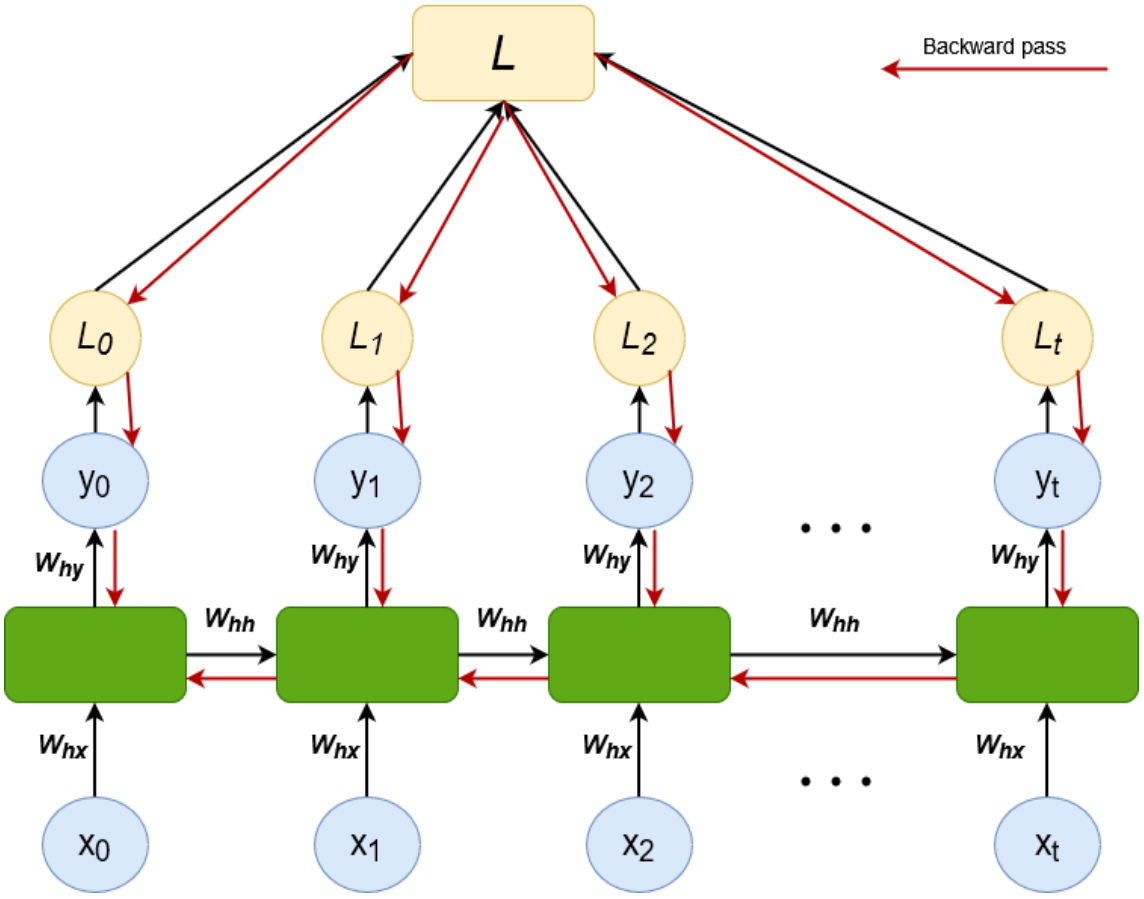
- της προηγούμενης εσωτερικής κατάστασης του δικτύου με τον πίνακα βαρών που αντιστοιχεί στην εσωτερική κατάσταση, ο οποίος είναι ο $\boldsymbol{W_{hh}^T}$
- του διανύσματος εισαγωγής με τον πίνακα βαρών που αντιστοιχεί στα δεδομένα εισαγωγής, ο οποίος είναι ο $\boldsymbol{W_{xh}^T}$.

Στο αποτέλεσμα του πολλαπλασιασμού εφαρμόζουμε μια μη γραμμική συνάρτηση για να καταλήξουμε στην τιμή της τρέχουσας κατάστασεις του δικτύου.

Το τελευταίο βήμα για να καταλήξουμε στον υπολογισμό της μεταβλητής y είναι να τροποποιήσουμε την τιμή της εσωτερικής κατάστασης πολλαπλασιάζοντάς την με άλλο πίνακα βαρών $\boldsymbol{W_{hy}^T}$ και να πάρουμε το αποτέλεσμα.

$$\hat{y}_t = \boldsymbol{W_{hy}^T} h_t \qquad (2.5)$$

Οι εξισώσεις (2.4) και (2.5) χρησιμοποιούνται στο πρώτο βήμα της εκπαίδευσης του δικτύου που είναι το βήμα της εμπρόσθιας τροφοδότησης. Το επόμενο βήμα είναι η ενημέρωση των πινάκων βαρών του δικτύου. Για την ενημέρωση αυτών των βαρών, τα αναδρομικά νευρωνικά δίκτυα χρησιμοποιούν και αυτά τον αλγόριθμο που είδαμε στο προηγούμενο κεφάλαιο: τον αλγόριθμο αντίστροφης διάδοσης σφαλμάτων. Όμως στην περίπτωση των RNN, λόγω της εξέλιξης του δικτύου μέσα στον χρόνο, ο αλγόριθμος ονομάζεται αλγόριθμος αντίστροφης διάδοσης σφαλμάτων στον χρόνο (**Backpropagation Through Time**), καθώς οι μετρήσεις σφάλματος πλέον πρέπει να διαδοθούν όχι μόνο σε εναν χρονικό βήμα, δηλαδή σε ενα μεμονωμένο MLP, αλλά σε όλα τα χρονικά βήματα του δικτύου, για να μπορέσουν να ενημερωθούν κατάλληλα οι παράμετροι σε όλα τα βήματα, όπως στο σχήμα της εικόνας 22.



Εικόνα 22 - Πέρασμα Αντίστροφης Διάδοσης Σφαλμάτων σε RNN

Όπως βλέπουμε στην εικόνα 22, ο αλγόριθμος Backpropagation through time (BPTT) για να ενημερώσει σωστά τα βάρη σε ενα αναδρομικό νευρωνικό δίκτυο μεταδίδει το σφάλμα, όχι μόνο από τον τελευταίο κόμβο που αθρόιζονται τα σφάλματα τον επιμερους συναρτήσεων κόστους, πρός τους μεμονωμένους νευρώνες κάθε χρονικού βήματος, αλλά και από το τελευταίο χρονικό βήμα της ακολουθίας πρός όλα τα προηγούμενα βήματα.

Αξίζει να σημειώσουμε οτι η εκπαίδευση των RNN είναι γενικά αργή καθώς δε μπορεί να υπάρξει παράλληλη επεξεργασία. Ο λόγος είναι οτι λόγω της εξάρτησης που έχει ο κάθε νευρώνας από τον προηγούμενο, πρέπει υποχρεωτικά να περιμένει τα αποτελέσματα του προηγούμενου κόμβου πρίν ξεκινήσει τους δικούς του υπολογισμούς.

Προβλήματα κατά την εκπαίδευση του νευρωνικού δικτύου με BPTT

Όπως έχει αναφερθεί, ο αλγόριθμος Backpropagation υπολογίζει την παράγωγο μιας συνάρτησης κόστους υπολογίζοντας τις επιμέρους μερικές παραγώγους των παραμέτρων του δικτύου και στην συνέχεια ενημερώνει τις παραμέτρους με βάση την τιμή της παραγώγου.

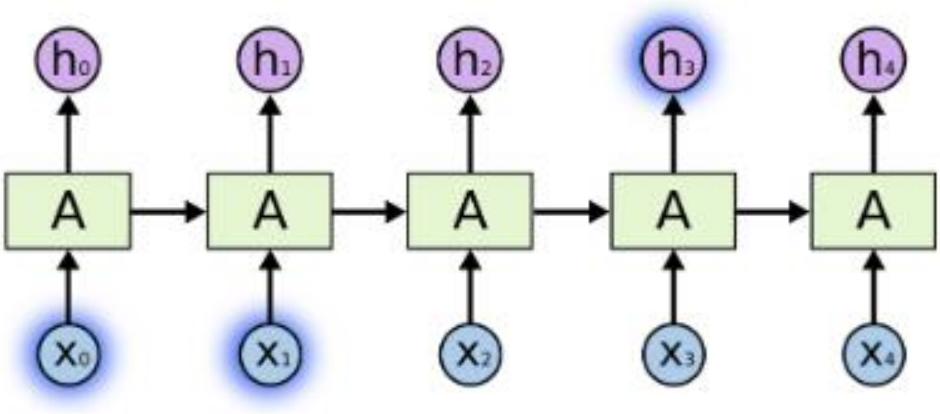
Ενα πρόβλημα που προκύπτει κατά την εκπαίδευση των αναδρομικών νευρωνικών δικτύων κατά το αντίστροφο πέρασμα, είναι το γεγονός οτι όσο ο αλγόριθμος υπολογίζει μερικές παραγώγους για χρονικά βήματα τα οποία βρίσκονται όλο και πιο πίσω στο παρελθόν, τόσο περισσότεροι πολλαπλασιασμοί μεταξύ των μητρών των βαρών χρειάζεται να λάβουν χώρα στην διαδικασία. Για νευρωνικά δίκτυα με πολλά επίπεδα, ο όγκος των πολλαπλασιασμών προκαλεί τα φαινόμενα των εκτινασσόμενων (**Exploding Gradients**) ή εξαφανιζόμενων (**Vanishing Gradients**) κλίσεων.

Το πρώτο φαινόμενο των εκτινασσόμενων κλίσεων πηγάζει από το γεγονός οτι όταν η πλειοψηφία των τιμών των κλίσεων που υπολογίζει ο αλγόριθμος είναι σημαντικά μεγαλύτερες της μονάδος (> 1), όσο πιο πίσω πάμε στο παρελθόν (σε παρελθοντικά χρονικά βήματα), τόσο οι πολλαπλασιασμοί που προκύπτουν, έχουν ως αποτέλεσμα η τελική τιμή να έχει μια πάρα πολύ μεγάλη τιμή.

Το δεύτερο φαινόμενο προκαλείται από το γεγονός οπου η πλειοψηφία των τιμών των κλίσεων είναι σημαντικά μικρότερες της μονάδος (< 1). Έτσι, όταν ο αλγόριθμος υπολογίζει τιμές των κλίσεων σε όλο και πιο παρελθοντικά χρονικά βήματα, τόσο η τελική τιμή γίνεται όλο και μικρότερη.

Ειδικότερα, το πρόβλημα των εξαφανιζόμενων κλίσεων έχει ως αποτέλεσμα το δίκτυο να μη μπορεί να συνδέσει πληροφορίες που εμφανίζονται στα αρχικά στάδια μιας ακολουθίας, με πληροφορίες που εμφανίζονται σε μεταγενέστερα χρονικά βήματα. Δε μπορεί δηλαδή να μάθει την σχέση των εξαρτήσεων ανάμεσα σε αυτά τα χρονικά βήματα (**Long-Term Dependencies problem**). Έτσι, για μεγαλύτερες ακολουθίες και, κατά συνέπεια, μεγαλύτερου μήκους «ξετυλιγμένα» γραφήματα, φαίνεται πως τα αναδρομικά νευρωνικά δίκτυα, στην γενική τους μορφή υστερούν στη μοντελοποίηση των ακολουθιών αυτών. Για να κατανοήσουμε καλύτερα το πρόβλημα, ας δούμε το παρακάτω παράδειγμα.

Για μικρού μήκους ακολουθίες, όπως είναι η πρόταση «Χθές έφαγα *φαγητό*», ενα αναδρομικό νευρωνικό δίκτυο που προσπαθεί να προβλέψει την τελευταία λέξη στην πρόταση (την λέξη *φαγητό*), δεν αντιμετωπίζει το πρόβλημα των εξαφανιζόμενων κλίσεων καθώς η ακολουθία είναι μίκρου μήκους και το «ξετυλιγμένο» γράφημα του δικτύου αποτελείται από λίγα επίπεδα. Συνεπώς οι εξαρτήσεις ανάμεσα στα στοιχεία της ακολουθίας δεν απέχουν πολύ σε όρους χρονικών βημάτων της ακολουθίας. Το νευρωνικό δίκτυο μπορεί να προβλέψει την λέξη *φαγητό* από τα συμφραζόμενα. Η εικόνα που ακολουθεί, αντικατοπτρίζει ενα τέτοιο δίκτυο.



Εικόνα 23 - Μοντέλο RNN στο οποίο δεν προκύπτει το προβλημα των εξαφανιζόμενων κλίσεων

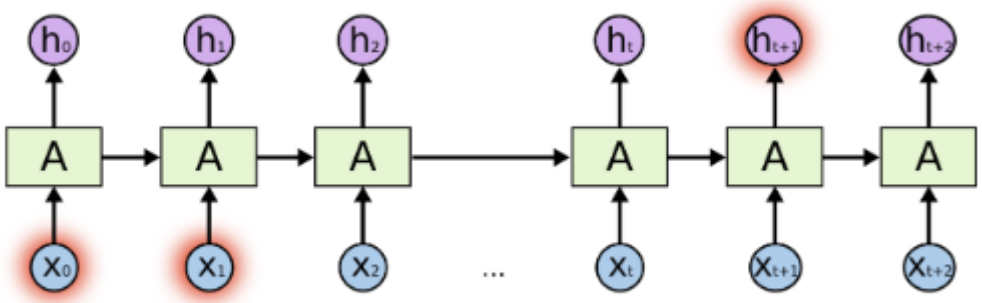
Το πρόβλημα σε ενα τέτοιου είδους μοντέλο εμφανίζεται όταν το μήκος της ακολουθίας μεγαλώνει και συνεπώς η συσχέτιση μεταξύ των διαφόρων στοιχείων (λέξεων) της ακολουθίας είναι πιο δύσκολο να υπολογιστεί από το δίκτυο. Ας πάρουμε για παράδειγμα μια πρόταση όπως η παρακάτω

«Μεγάλωσα στην Ελλάδα... Μιλάω άπταιστα *Ελληνικά*»

Σε μια τέτοια περίπτωση όπου το δίκτυο χρειάζεται να αναγνωρίσει όχι μόνο οτι, σύμφωνα με την πρόσφατη πληροφορία, η λέξη που πρέπει να προβλέψει είναι μια γλώσσα, αλλά να καταφέρει να προβλέψει και την συγκεκριμένη γλώσσα, τα *Ελληνικά*, εμφανίζεται το πρόβλημα των μακροχρόνιων εξαρτήσεων που αναφέρθηκε σε προηγούμενη παράγραφο. Όσο πιο μεγάλη είναι η απόσταση μεταξύ της λέξης που πρέπει να προβλεφθεί και της λέξης η οποία παρέχει όλη την σχετική πληροφορία για την πρόβλεψη, τόσο πιο δύσκολη γίνεται η πρόβλεψη λόγω των εξαφανιζόμενων κλίσεων.

Αυτό συμβαίνει γιατί όσο πιο πίσω πάμε στα χρονικά βήματα του δικτύου, τόσο μικρότερες τιμές έχουν οι κλίσεις που υπολογίζει το δίκτυο. Και λόγω των αμελητέων αυτών τιμών, το δίκτυο αποκτά μια προκατάληψη στο να μαθαίνει τις πιο βραχυπρόθεσμες σχέσεις μεταξύ των στοιχείων της ακολουθίας και να μην ενσωματώνει τις πιο μακρυπρόθεσμες εξαρτήσεις με την ίδια βαρύτητα στους υπολογισμούς.

Στην εικόνα 24 παρουσιάζεται το πρόβλημα των μακροχρόνιων εξαρτήσεων κατά τη μοντελοποίηση μιας ακολουθίας μεγάλου μήκους. Στο μοντέλο αυτό, η ακολουθία «ξεδιπλώνει» ενα γράφημα με μεγάλο αριθμό χρονικών βημάτων, με αποτέλεσμα να εμφανίζεται το πρόβλημα των εξαφανιζόμενων κλίσεων.



Εικόνα 24 - Πρόβλημα μακροχρόνιων εξαρτήσεων σε RNN

Σε μοντέλα όπως αυτό της εικόνας 24, οι νευρώνες που βρίσκονται στα μεταγενέστερα χρονικά βήματα της ακολουθίας γίνονται όλο και λιγότερο ευαίσθητα στα δεδομένα εισαγωγής των αρχικών χρονικών βημάτων, με αποτέλεσμα το δίκτυο να «ξεχνάει» τα αρχικά δεδομένα (τις λέξεις στο παράδειγμά μας)(Graves 2012).

Για την επίλυση του προβλήματος των αναδρομικών δικτύων να μοντελοποιήσουν σωστά ακολουθίες δεδομένων, χωρίς να εμφανίζουν το πρόβλημα των εξαφανιζόμενων ή εκτινασσόμενων κλίσεων, προτάθηκε μια παραλλαγή της γενικής αρχιτεκτονικής των αναδρομικών δικτύων, τα δικτυα μακράς βραχείας μνήμης (**LSTM - Long Short-Term Memory**) (Hochreiter and Schmidhuber ,1997), τα οποία θα εξετάσουμε στην επόμενη ενότητα.

Δίκτυα Μακράς Βραχείας Μνήμης (LSTM)

Όπως αναφέραμε και στην προηγούμενη ενότητα, οι καθιερωμένες αρχιτεκτονικές των αναδρομικών νευρωνικών δικτύων αδυνατούν να διατηρήσουν τις μακροχρόνιες εξαρτήσεις ανάμεσα στα δεδομένα μιας μεγάλης ακολουθίας. Το πρόβλημα αυτό προκύπτει από το φαινόμενο των εξαφανιζόμενων κλίσεων (Hochreiter, 1991, Bengio et al., 1994). Στην παρούσα ενότητα θα παρουσιάσουμε, παραλείποντας τη μαθηματική ανάλυση, τα νευρωνικά δίκτυα μακράς βραχείας μνήμης τα οποία είναι μια ειδική κατηγορία των αναδρομικών νευρωνικών δικτύων και τα οποία επιλύουν το παραπάνω πρόβλημα καθιστώντας δυνατή τη μοντελοποίηση ακολουθιών μεγάλου μήκους.

Αρχιτεκτονική LSTM

Η ονομασία των δικτύων μακράς βραχείας μνήμης πηγάζει από το μέσο με το οποίο αποθηκεύουν την πληροφορία, το οποίο διαφέρει από το αντίστοιχο μέσο των αναδρομικών νευρωνικών δικτύων. Ενα RNN έχει δύο τύπων μνήμες:

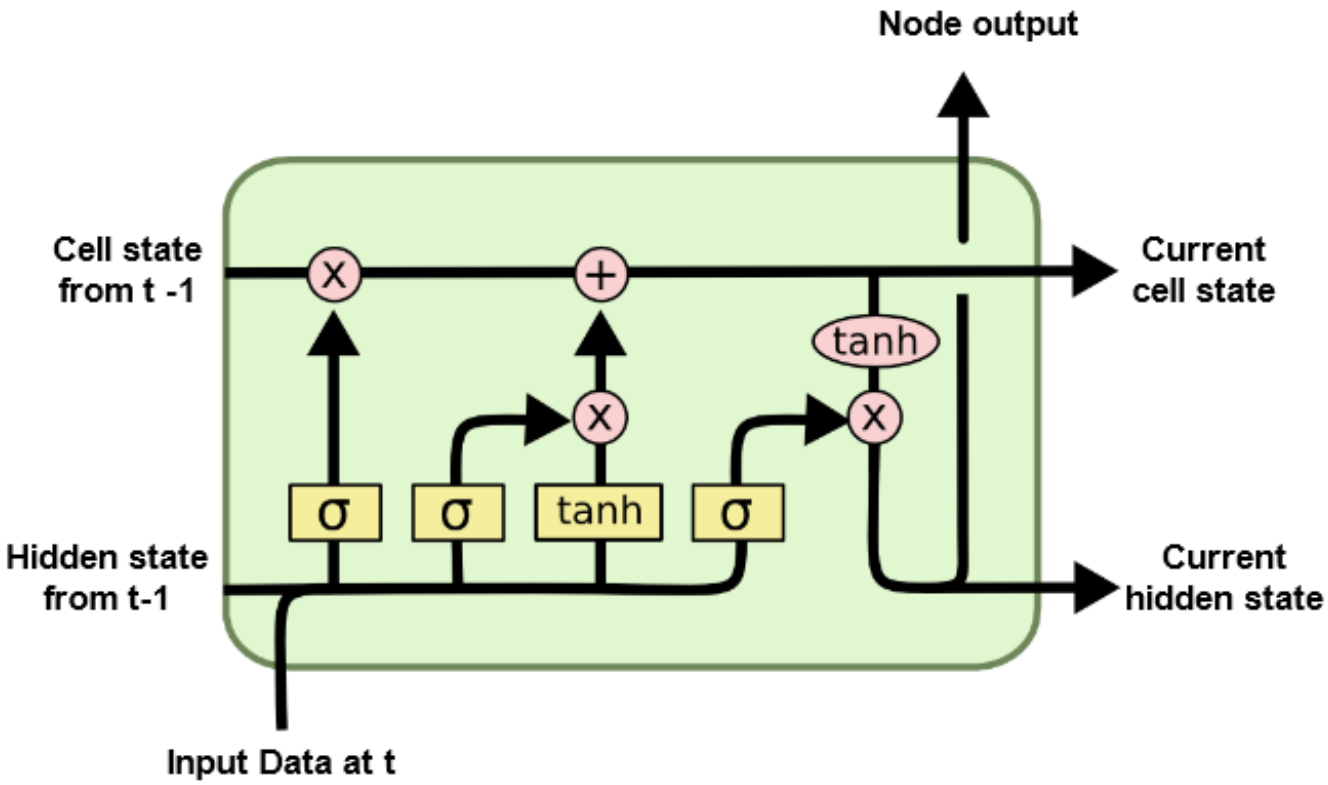
1. Τη μακροπρόθεσμη μνήμη, η οποία υπάρχει στα βάρη του δικτύου που υπόκεινται σε μια διαδικασία ενημέρωσης κατά την διαρκεια της εκπαίδευσης και τα οποία ενσωματώνουν όλη την γνώση που αποκτά το δίκτυο στο πέρασμα του χρόνου.
2. Την βαρχυπρόθεσμη μνήμη, η οποία συναντάται στη μορφή των συναρτήσεων ενεργοποιήσεων οι οποίες μεταφέρουν πληροφορία από τον ένα κόμβο του δικτύου στον επόμενο στην σειρά.

Τα δίκτυα LSTM έχουν εναν ενδιάμεσου τύπου μνήμης, η οποία συναντάται στο βασικό δομικό στοιχείο τους, το κύταρρο μνήμης, το οποίο είναι ενα σύνθετο στοιχείο που αποτελείται από πιο απλά στοιχεία, συνδεδεμένα με ενα συγκεκριμένο τρόπο.

Όπως και τα αναδρομικα νευρωνικά δίκτυα, έτσι και τα LSTM έχουν σαν βασική αρχιτεκτονική δομή τους επαναληπτικούς κόμβους που προκύπτουν σε κάθε βήμα της ακολουθίας. Όμως, στην περίπτωση των LSTM, η εσωτερική δομή των επαναληπτικών κόμβων διαφέρει αρκετά σε σχέση με αυτή των απλών RNN. Ο κάθε κόμβος αποτελείται από «κύτταρα LSTM» τα οποία, πέρα από την εξωτερική επανάληψη που υπάρχει στο σύνολο του δικτύου, έχουν και μια δικία τους αναδρομική σχέση. Τα κύτταρα αυτά έχουν περισσότερες παραμέτρους από ενα απλό RNN και επίσης είναι εξοπλισμένα με ενα σύστημα μονάδων πυλών ώστε να μπορούν να διαχειρίζονται την πληροφορία που μεταφέρεται μέσα στο δίκτυο.

Οι κόμβοι των απλών αναδρομικών δικτύων, όπως είδαμε σε προηγούμενες ενότητες, αποτελούνται από ενα επίπεδο νευρωνικού δικτύου MLP (εικόνα 12) και αυτή η δομή επαναλαμβάνεται στο «ξετύλιγμα» της ακολουθίας. Οι κόμβοι των δικτύων μακράς βραχείας μνήμης έχουν την δομή των «κυττάρων LSTM», τα οποία παρουσιάσαμε στην προηγούμενη παράγραφο, τα οποία με την σειρά τους αναλύονται σε τέσσερα επίπεδα υποδικτύων MLP, τα οποία αλληλεπιδρούν μεταξύ τους για να δημιουργήσουν τη μνήμη του δικτύου.

Παρακάτω απεικονίζεται η εσωτερική δομή των «κυττάρων LSTM», τα οποία είναι το βασικό χαρακτηριστικό που απαλείφει το πρόβλημα των εξαφανιζόμενων κλίσεων και καθιστά τα LSTM καταλληλότερα για τη μοντελοποίηση δεδομένων ακολουθιών.

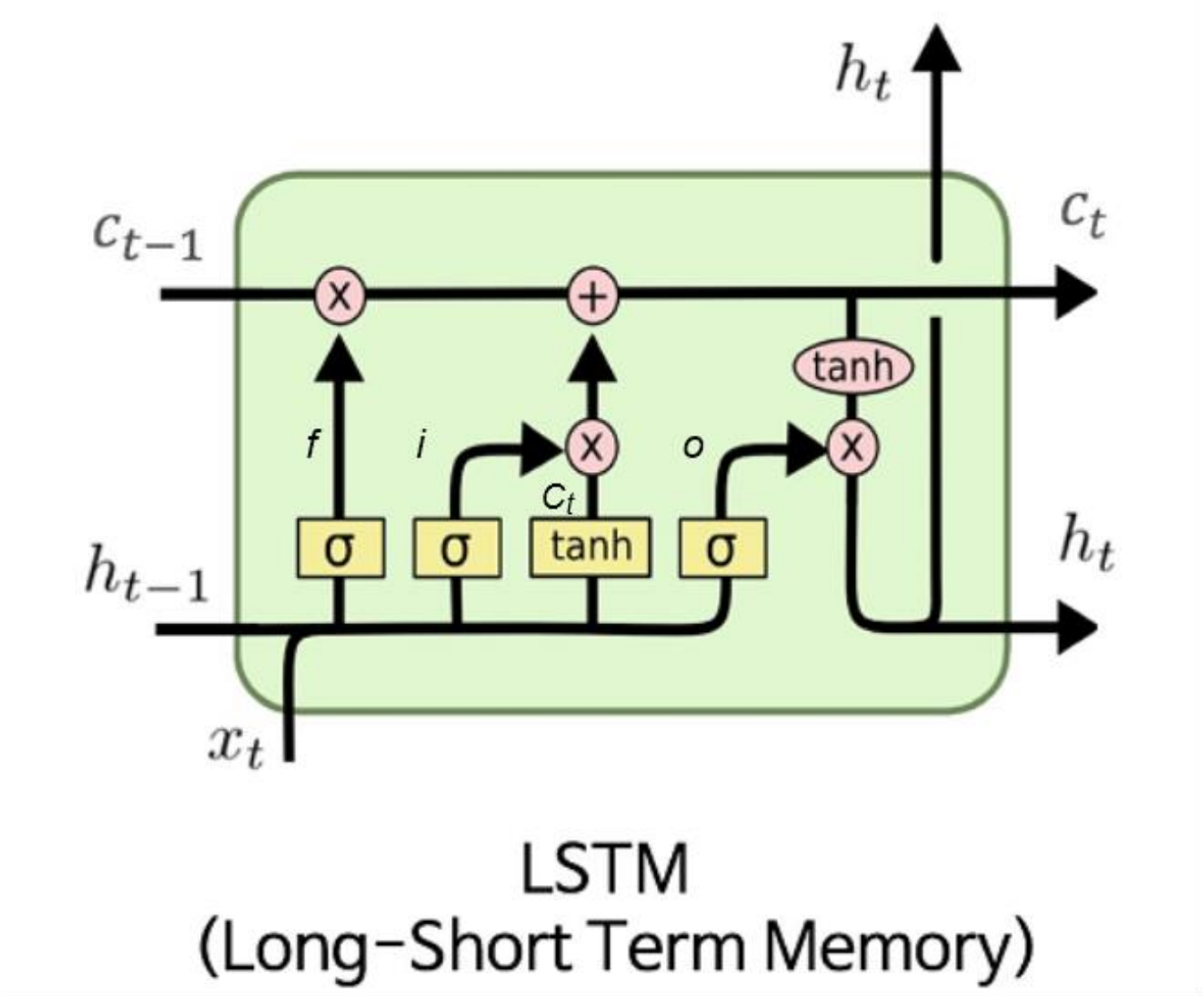


Εικόνα 25 – Ένα «κύτταρο LSTM»

Η εικόνα 25 παρουσιάζει την εσωτερική αρχιτεκτονική των επαναλαμβανόμενων δομών ενός δικτύου LSTM. Στην επόμενη παράγραφο θα αναλύσουμε το διάγραμμα του «κυττάρου LSTM». Πρότου όμως προχωρήσουμε θα ορίσουμε τους εξής συμβιβασμούς:

- Ως i θα συμβολίζουμε τις πύλες εισόδου
- Ως f θα συμβολίζουμε τις πύλες λησμόνησης
- Ως o θα συμβολίζουμε τις πύλες εξόδου.
- Ως C_t θα συμβολίζουμε την εσωτερική κατάσταση του κυτάρρου (cell state στο διάγραμμα)
- Ως h_t θα συμβολίζουμε την κρυφή κατάσταση (hidden state στο διάγραμμα)

Η εικόνα 26 αναπαριστά το ίδιο διάγραμμα, αυτή τη φορά όμως έχουμε αντικαταστήσει τα κείμενα με τα σύμβολα που ορίσαμε πιο πάνω.

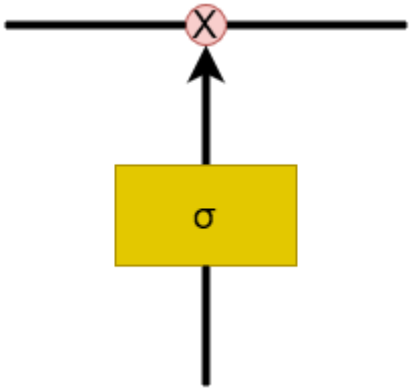


Εικόνα 26 - Το μοντέλο LSTM

Παρακάτω θα αναλύσουμε περαιτέρω την λειτουργία και τον ρόλο του κάθε επιπέδου μέσα στο δίκτυο.

Πύλες: σε ενα LSTM δίκτυο αναφέρονται στα στοιχεία εκείνα τα οποία ρυθμίζουν την εισαγωγή νέων δεδομένων στην εσωτερική κατάσταση του δικτύου ή την διαγραφή υπαρχόντων δεδομένων από αυτήν. Είναι μονάδες απλών νευρωνικών δικτύων οι οποίες χρησιμοποιούν σαν συνάρτηση ενεργοποίησης την σιγμοειδή συνάρτηση. Ονομάζονται πύλες γιατί ο ρόλος

τους είναι να διακόπτουν ή να επιτρέπουν την ροή της πληροφορίας στο δίκτυο. Αν η τιμή του αποτελέσματος της πύλης είναι ίση με 1, τότε η πληροφορία επιτρέπεται να διαδοθεί ανέπαφη πρός το δίκτυο. Αν η τιμή όμως είναι 0, τότε διακόπεται η μετάδοση της οποιασδήποτε πληροφορίας. Τα δεδομένα των πυλών, τα οποία χρησιμοποιούν στους υπολογισμούς τους, είναι τα δεδομένα εισαγωγής στο τρέχων χρονικό βήμα, x_t, καθώς και η εσωτερική κρυφή κατάσταση, h_{t-1}, του προηγούμενο χρονικού βήματος. Το διάγραμμα μιας πύλης φαίνεται στην εικόνα 27.



Εικόνα 27 - Πύλη ενός LSTM

- *Πύλη λησμόνησης* (f) *:* Σε αυτό το σημείο γίνεται η αξιολόγηση της προηγούμενης εσωτερικής κατάστασης του δικτύου, h_{t-1}, σε συνδυασμό με τα δεδομένα εισαγωγής του τρέχοντος χρονικού βήματος, x_t. Είναι ουσιαστικά ενας μηχανισμός με τον οποίο το δίκτυο αποφασίζει ποιές πληροφορίες είναι περισσότερο σημαντικές και ποιές όχι. Μέσω της σιγμοειδούς συνάρτησης, η πύλη επιστρέφει εναν αριθμό ανάμεσα στο 0 και στο 1 για κάθε αριθμό που υπάρχει στην κατάσταση του κυττάρου C_t, για το χρονικό βήμα $t = t - 1$. Η εξίσωση με την οποία υπολογίζεται η έξοδος της πύλης δίδεται από τον παρακάτω τύπο:

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \qquad (2.6)$$

- *Πύλη εισαγωγής* (ι)*:* Αυτή η πύλη, όπως και η πύλη λησμόνησης, έχει σαν είσοδο τα δεδομένα εισαγωγής του τρέχοντος χρονικου βήματος, x_t, και της προηγούμενης εσωτερικής κατάστασης του δικτύου, h_{t-1}. Η πύλη εισαγωγής, χρησιμοποιεί και αυτή μια σιγμοειδή συνάρτηση, με σκοπό να αποφασίσει αυτή την φορά ποιές τιμές θα ενημερωθούν και κατά πόσο θα μεταβληθεί η τιμή τους. Η εξίσωση με την οποία υπολογίζεται η έξοδος της πύλης είναι παρόμοια με αυτή της πύλης λησμόνησης και

δίδεται από τον παρακάτω τύπο:

$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] + b_i\right) \quad (2.7)$$

- *Εσωτερική κατάσταση του κυττάρου:* Για την ενημέρωση της κατάστασης του κυττάρου του προηγούμενου χρονικού βήματος, C_{t-1}, χρειάζεται να υπολογιστεί ενα νεο διάνυσμα το οποίο θα έχει ως στοιχεία του πιθανές τιμές που θα θέλαμε να εισάγουμε στην νέα εσωτερική κατάσταση του χρονικού βήματος C_t. Σε αυτό το βήμα, τα δεδομένα εισαγωγής του τρέχοντος χρονικου βήματος, x_t, και της προηγόυμενης εσωτερικής κατάστασης του δικτύου, h_{t-1}, φιλτράρονται μέσα από μια συνάρτηση υπερβολικής εφαπτομένης (tanh) και πλέον έχουμε τις υποψήφιες πρός εισαγωγή στην εσωτερική κατάσταση τιμές. Η εξίσωση με την οποία υπολογίζονται οι υποψήφιες τιμές δίνεται από την σχέση:

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (2.8)$$

Αφού υπολογιστούν και οι υποψήφιες τιμές πρός εισαγωγή, η διαδικασία ενημέρωσης της εσωτερικής κατάστασης του κυττάρου συνεχίζεται χρησιμοποιώντας τα αποτελέσματα των προηγούμενων διαδικασιών.

1. Πολλαπλασιάζουμε την παλιά εσωτερική κατάσταση C_{t-1} με το διάνυσμα f_t που περιέχει την πληροφορία που θέλουμε να διαγραφεί από τη μνήμη του δικτύου.
2. Πολλαπλασιάζουμε το διάνυσμα των νέων τιμών που θέλουμε να ενημερωθούν με το διάνυσμα των πιθανών νέων τιμών που θα εισαχθούν στη νεα εσωτερική κατάσταση.
3. Τέλος προσθέτουμε αυτές τις δυο μεταβλητές για να υπολογίσουμε τη νεα εσωτερική κατάσταση του κυττάρου.

Ο υπολογισμός της νέας εσωτερική κατάστασης, C_t, δίνεται από την εξίσωση (2.9).

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (2.9)$$

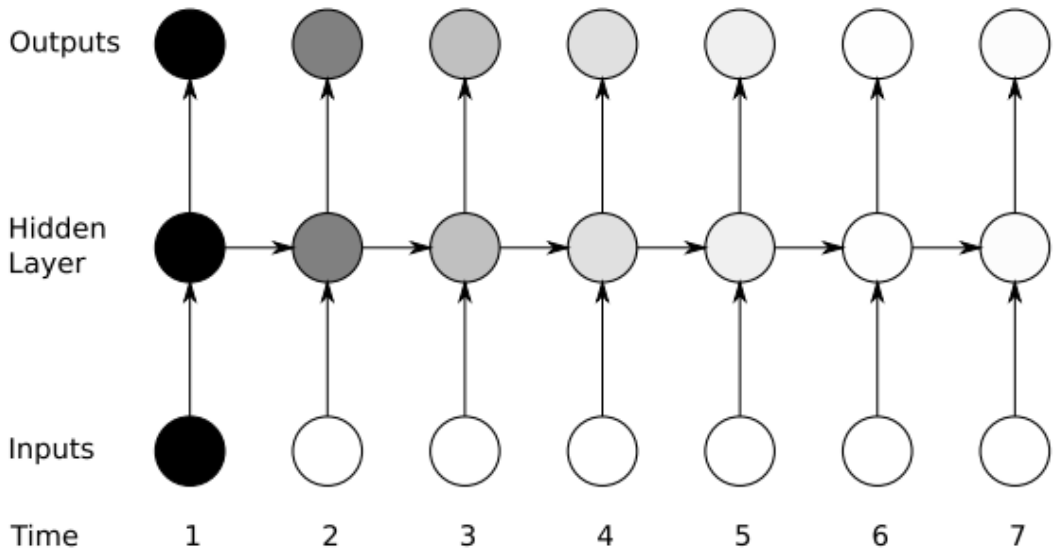
- *Πύλη εξαγωγής (o_t):* Η πύλη αυτή είναι υπεύθυνη για το τελικό αποτέλεσμα που εξάγει το «κύτταρο LSTM». Για τον υπολογισμό της τελικής τιμής του κυττάρου χρειάζεται να πολλαπλασιάσουμε την ενημερωμένη εσωτερική κατάσταση, C_t, με το αποτέλεσμα της πύλης εξόδου, o_t. Η συνήθης διαδικασία είναι να εφαρμόσουμε τη μη γραμμική συνάρτηση $tanh$ πάνω στην εσωτερική κατάσταση του κυττάρου. Αυτή η ενέργεια θα αναγκάσει τις τιμές του διανύσματος να περιοριστούν ανάμεσα στο -1 και το 1. Στην

συνέχεια εφαρμόζουμε την σιγμοειδή συνάρτηση στο αποτέλεσμα της πύλης εξόδου ώστε να αποφασίσουμε ποια πληροφορία από την καινούρια εσωτερική κατάσταση θέλουμε τελικά να εξάγουμε. Τέλος, πολλαπλασιάζουμε τα αποτελέσματα των δυο αυτών ενεργειών και έχουμε ως έξοδο την πληροφορία που αποφασίασμε να κρατήσουμε από την συνολική διαδικασία. Οι εξισώσεις που περιγράφουν τα παραπάνω είναι οι εξής:

$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right) \tag{2.10}$$

$$h_t = o_t * \tanh(C_t) \tag{2.11}$$

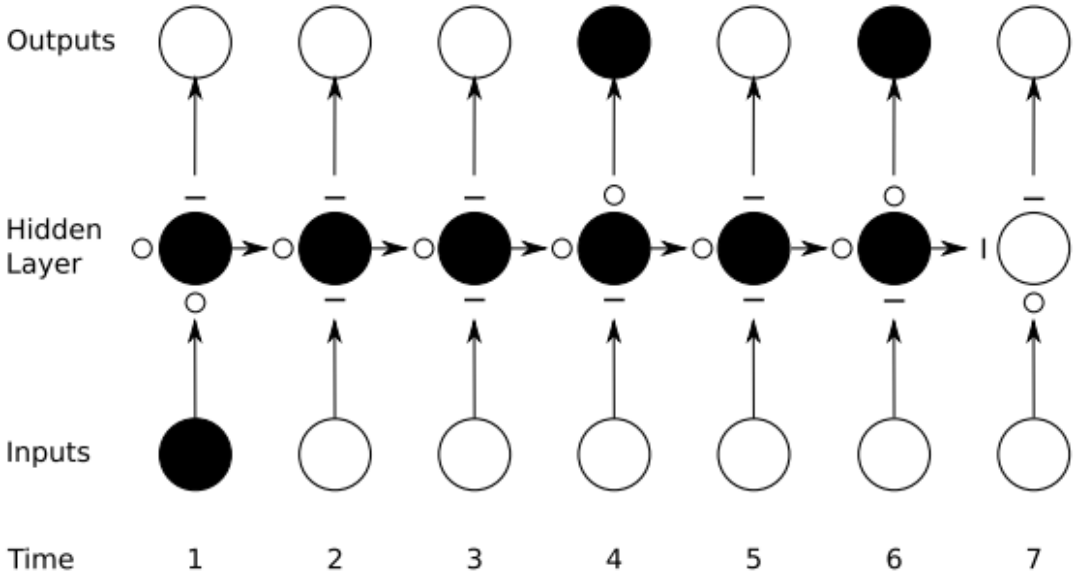
Στις εικόνες 28 και 29 βλέπουμε το πρόβλημα των εξαφανιζόμενων κλίσεων που παρουσιάζεται στην απλή αρχιτεκτονική αναδρομικών νευρωνικών δικτύων (εικόνα 28) καθώς και την επίλυση του προβλήματος αυτού από την αρχιτεκτονική LSTM.



Εικόνα 28 - Το πρόβλημα των εξαφανιζόμενων κλίσεων σε RNN

Το γράφημα της εικόνας 28 δείχνει ενα «ξετυλιγμένο» γράφημα ενός αναδρομικού δικτύου στον χρόνο. Παρατηρούμε πως όσο πιο πολύ εκτυλίσσεται το δίκτυο στην πάροδο του χρόνου, τόσο η ένταση της σκίασης εξασθενεί σε κάθε κόμβο. Η σκίαση σε αυτό το γράφημα ερμηνεύεται ως η ευαισθησία των κόμβων κάθε χρονικού βήματος στις τιμές εισόδου του χρονικού βήματος 1, ώπου, όσο πιο έντονη (σκοτεινή) είναι η σκίαση, τόσο μεγαλύτερη ευαισθησία έχει ο κόμβος σε αυτές τις τιμές εισόδου. Παρατηρούμε, λοιπόν, πως οι κόμβοι των μεταγενέστερων χρονικών

βημάτων είναι λιγότερο ευαίσθητοι στα δεδομένα εισαγωγής του χρονικού βήματος 1 (η σκίαση εξασθενεί όλο και περισσότερο μέχρι το σημείο να μην υπάρχει καθόλου σκίαση) σε σχέση με τους αρχικούς κόμβους του δικτύου, γεγονός που σημαίνει πως το δίκτυο δε μπορεί να συνδέσει την πληροφορία που έχει αποκτήσει στα πρωταρχικά στάδια, με τα πιο πρόσφατα γεγονότα.



Εικόνα 29 - Διατήρηση της πληροφορίας των μακροπρόθεσμων εξαρτήσεων από τα LSTM

Το γράφημα της εικόνας 29 αναπαριστά την ικανότητα των δικτύων μακράς βραχείας μνήμης να διατηρούν τις μακροχρόνιες εξαρτήσεις, σε σχέση με τα απλά αναδρομικά νευρωνικά δίκτυα. Το γράφημα αυτό είναι παρόμοιο με της εικόνας 28, με την διαφορά οτι τώρα εμφανίζονται οι πύλες εισόδου, λησμόνησης και εξαγωγής δεδομένων (κάτω, αριστερά και πάνω από τον κόμβο αντίστοιχα). Και σε αυτή την περίπτωση, η σκίαση των κόμβων αντικατοπτρίζει την ευαισθησία των κόμβων στα δεδομένα εισαγωγής του χρονικού βήματος 1. Το γράφημα παρουσιάζει όλες τις πυλές ως εντελώς ανοιχτές («Ο»), ή ως εντελώς κλειστές («−»), για λόγους απλότητας. Όσο η πύλη λησμόνησης παραμένει ανοιχτή και η πύλη εισόδου κλειστή, η εσωτερική κατάσταση του κυττάρου μπορεί να διατηρεί την πληροφορία από το πρώτο χρονικό βήμα της ακολουθίας. Η πύλη εξόδου αποφασίζει ποιά πληροφορία θα εξαχθεί σαν αποτέλεσμα χωρίς να επηρεάσει την εσωτερική κατάσταση του κυττάρου. Με αυτό τον τρόπο, η πληροφορία μπορεί να «ρέει» ανάμεσα στα χρονικά βήματα του «ξετυλιγμένου» γραφήματος ενός LSTM παραμένοντας αναλλοίωτη.

Παραδείγματα εφαρμογών των RNN/LSTM

Αδυναμίες των RNN/LSTM και άλλα νευρωνικά δίκτυα για τη μοντελοποίηση ακολουθιών

Όπως και σε όλες τις τεχνολογίες, έτσι και στα αναδρομικά νευρωνικά δίκτυα υπάρχουν περιορισμοί στις δυνατότητες των μοντέλων:

- *Απώλεια πληροφορίας:* Αν και τα αναδρομικά νευρωνικά δίκτυα μπορούν να χρησιμοποιηθούν σε φαινόμενα που φαίνονται αρκετά περίπλοκα, όπως η παραγωγή μιας ακολουθίας από μουσικές νότες, δεν λύνουν απόλυτα το υποβόσκων πρόβλημα: η ροή και η μετάδοση της πληροφορίας μέσα από μια αρχιτεκτονική η οποία αποτελείται από πολλές αναδρομικές συνδέσεις οδηγεί τελικά σε απώλεια πληροφορίας και ενδεχομένως σε προβλήματα στην εκπαίδευση του δικτύου.
- *Αδυναμία παραλληλοποίησης των υπολογισμών:* Τα αναδρομικά νευρωνικά δίκτυα επεξεργάζονται τα δεδομένα εισαγωγής τους σειριακά, καθώς οι υπολογισμοί σε κάθε χρονικό βήμα λαμβάνουν χώρα με τη σειρά. Η εξ'ορισμού σειριακή φύση τους τα καθιστά μη αποδοτικά σε σχέση με τις δυνατότητες παράλληλης επεξεργασίας των μοντέρνων υπολογιστών που χρησιμοποιούνται για την εκπαίδευση των νευρωνικών δικτύων.
- *Αδυναμία μοντελοποίησης ακολουθιών ιδιαίτερα μεγάλου μήκους:* Παρόλο που τα LSTM δίκτυα έχουν καταφέρει να μοντελοποιούν ακολουθίες που έχουν μήκος χρονικών βημάτων της τάξεως του 10 ή του 100, δεν προσαρμόζονται το ίδιο αποτελεσματικά όταν οι ακολουθίες έχουν μήκος χρονικών βημάτων της τάξεως του 1.000 ή των 10.000.

Οι παραπάνω περιορισμοί των αναδρομικών νευρωνικών δικτύων οδήγησαν στην ιδέα μιας διαφορετικού τύπου αρχιτεκτονικής για τη μοντελοποίηση ακολουθιών, η οποία μοιάζει με την αρχιτεκτονική των πλήρως συνδεδεμένων δικτύων MLP: Τους μετασχηματιστές (**Transformers**).

Στην επόμενη ενότητα θα δούμε συνοπτικά την δομή των μετασχηματιστών καθώς και τους λόγους που τους καθιστούν αποδοτικότερους σε σχέση με τα αναδρομικά νευρωνικά δίκτυα για τη μοντελοποίηση ακολουθιών.

Μηχανισμοί Προσοχής – Μετασχηματιστές (Attention Mechanisms – Transformers)

Όπως αναφέρθηκε και στην προηγούμενη ενότητα, οι αδυναμίες των RNN μοντέλων είναι οτι κατά την διάρκεια των υπολογισμών τους πάσχουν από απώλεια πληροφορίας, λόγω της σειριακής δομής τους δε μπορούμε να παραλληλίσουμε τους υπολογισμούς και, τέλος, αδυνατούν να μοντελοποιήσουν ακολουθίες ιδιαίτερα μεγάλου μήκους.

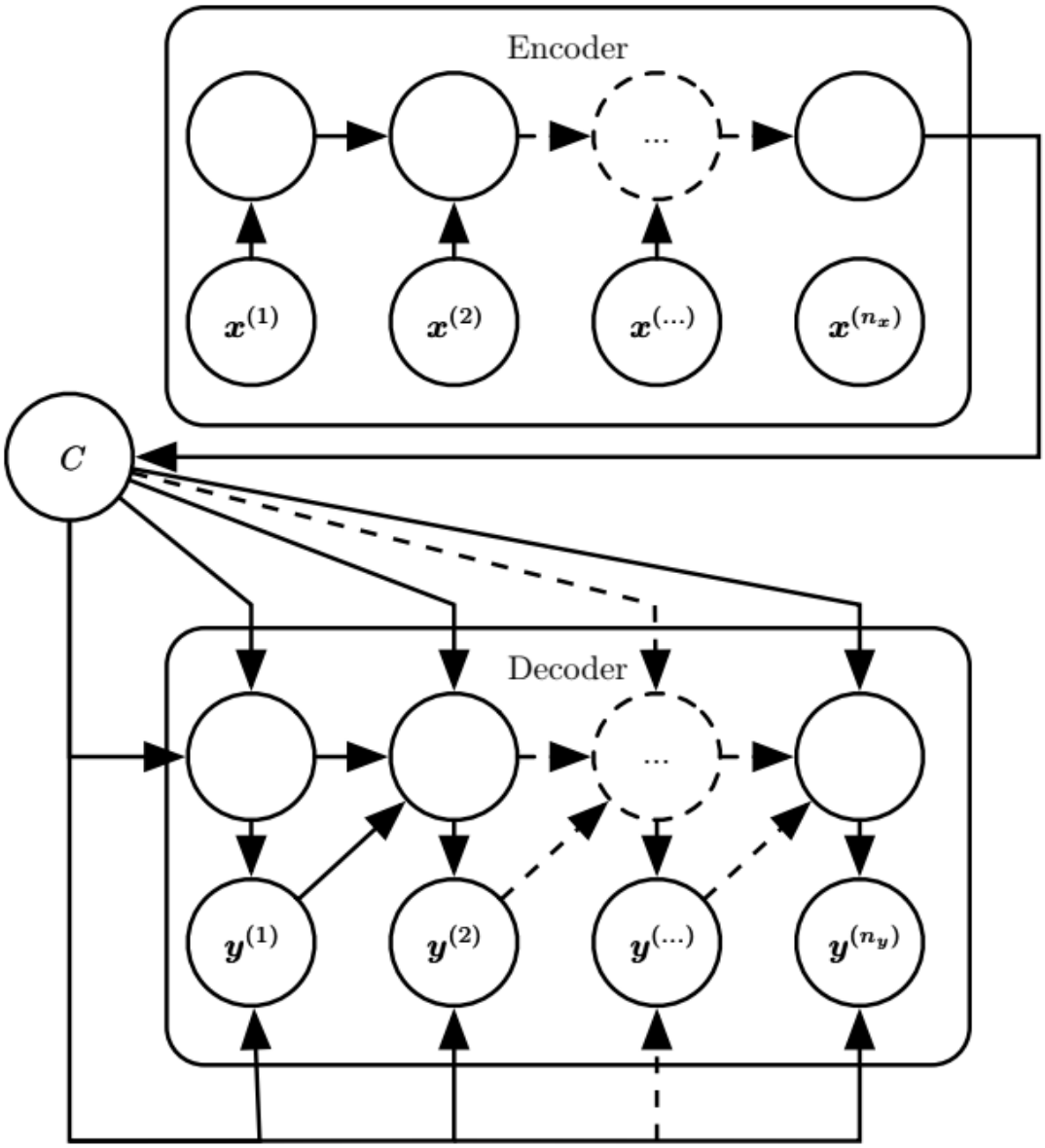
Μηχανισμός Προσοχής

Η κύρια ιδέα πίσω από την ανάπτυξη των μετασχηματιστών είναι οτι χρησιμοποιούν ενα μηχανισμό προσοχής (**Attention mechanism**) ο οποίος δίνει έμφαση σε εκείνα τα στοιχεία της ακολουθίας που πρέπει να εισαχθούν στο μοντέλο.

Ο μηχανισμός προσοχής είναι μια επέκταση της οικογένειας των δικτύων κωδικοποιητή – αποκωδικοποιητή (**Encoder - Decoder**)**.** Ο κωδικοποιητής λαμβάνει μια ακολουθία σαν είσοδο (την είσοδο αυτή την αποκαλούμε και «συμφραζόμενα») και την αποτυπώνει σε ενα σταθερού μήκους διάνυσμα. Στη συνέχεια, ο αποκωδικοποιητής εξάγει ενα άλλο διάνυσμα βασισμένος στο κωδικοποιημένο διάνυσμα που λαμβάνει σαν είσοδο. Η αρχιτεκτονική αυτή, που ονομάζεται και ακολουθία-σε-ακολουθία αρχιτεκτονική, έχει χρησιμοποιηθεί κατά κόρον σε αναδρομικά νευρωνικά μοντέλα τα οποία εκπαιδεύονται στην αντιστοίχιση μιας ακολουθίας σε μια άλλη ακολουθία η οποία ενδέχεται να έχει διαφορετικό μήκος.

Στην εικόνα 30 φαίνεται ενα παράδειγμα ενός μοντέλου κωδικοποιητή – αποκωδικοποιητή. Το μοντέλο εκπαιδεύεται στο να μάθει να παράγει μια ακολουθία $y^{(1)}, y^{(2)}, \dots, y^{(n)}$ δεδομένης μιας άλλης ακολουθίας $x^{(1)}, x^{(2)}, \dots, x^{(n)}$. Το μοντέλο αποτελείται από τον κωδικοποιητή ο οποίος διαβάζει την ακολουθία και τον αποκωδικοποιητή ο οποίος παράγει την ακολουθία εξόδου. Τέτοια μοντέλα χρησιμοποιούνται αρκετά σε προβλήματα μετάφρασης μηχανής, όπου σαν είσοδος δίνεται μια ακολουθία η οποία μπορεί να είναι μια πρόταση στα Ελληνικά και σαν έξοδος αναμένεται μια ακολουθία η οποία θα αποτελεί τη μετάφραση της ελληνικής πρότασης σε μια άλλη γλώσσα, π.χ. τα Γαλλικά. Το τελευταίο κρυφό επίπεδο αυτού του μοντέλου παράγει μια σταθερού μεγέθους μεταβλητή *C,* η οποία ερμηνεύεται ως η σημασιολογική περίληψη της ακολουθίας εισόδου. Η έξοδος αυτή του τελευταίου κρυφού επιπέδου είναι η είσοδος που λαμβάνει ο αποκωδικοποιητής.

Το μειονέκτημα αυτών των μοντέλων είναι το γεγονός οτι η μεταβλητή *C,* λόγω του σταθερού μεγέθους της, ενδέχεται να μη μπορεί να αποτυπώσει σωστά το νόημα μιας μεγάλης πρότασης. Ως λύση στο πρόβλημα αυτό προτάθηκε ο μηχανισμός προσοχής (εικόνα 31), ο οποίος, σε αντίθεση με το σύστημα κωδικοποιητή – αποκωδικοποιητή που χρησιμοποιεί τη μεταβλητή *C* για να αποτυπώσει όλο το νόημα της ακολουθίας, διαβάζει όλη την ακολουθία (π.χ. μια πρόταση ή μια παράγραφο) και παράγει μεμονωμένες μεταφράσεις των λέξεων που την αποτελούν, συγκεντρώνοντας, κάθε φορά, την προσοχή του σε διαφορετικά κομμάτια της ακολουθίας με σκοπό να καταλάβει ποιές είναι οι σημασιολογικές συνδέσεις μεταξύ των διαφόρων λέξεων, ώστε να προβλέψει την επόμενη λέξη.

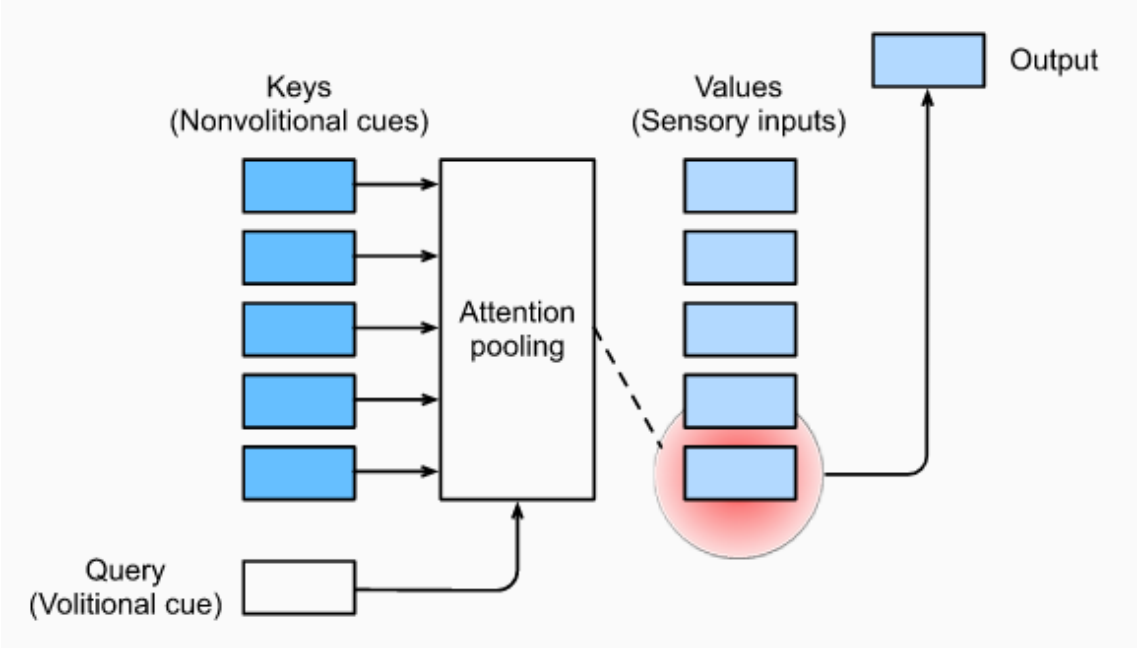


Εικόνα 30 - Μοντέλο κωδικοποιητή – αποκωδικοποιητή

Ιστορικά, ο μηχανισμός προσοχής προσομοιάζει την ικανότητα που έχουμε οι άνθρωποι να λαμβάνουμε πληροφορία από το περιβάλλον μας, π.χ. παρατηρώντας ενα τοπίο, και να συγκεντρώνουμε την προσοχή μας σε ενα μικρό κομμάτι της συνολικής πληροφορίας και να παραμερίζουμε την πληροφορία που δεν μας ενδιαφέρει.

Για παράδειγμα, ας υποθέσουμε οτι έχουμε μπροστά μας μια σειρά από αντεικείμενα όπως ενα βιβλίο, μια εφημερίδα, ενα φλιτζάνι καφέ, μια δέσμη απο χαρτιά και ενα τετράδιο και ας υποθέσουμε οτι το φλιτζάνι του καφέ έχει κόκκινο χρώμα ενώ όλα τα υπόλοιπα αντικείμενα είναι ασπρόμαυρα. Σε αυτή την περίπτωση, όταν κοιτάξουμε τα αντικείμενα, παρά την θέληση μας η προσοχή μας θα πέσει πάνω στο έντονο κόκκινο χρώμα του φλιτζανιού το οποίο διαφέρει από τα υπόλοιπα αντικείμενα (**non-volitional cue**). Αφού περάσει η ακούσια ενέργεια μας να συγκεντρώσουμε την προσοχή μας στο φλιτζάνι, επιλέγουμε, με την βούληση μας πλέον, να στρέψουμε το βλέμμα μας στο βιβλίο, το οποίο θέλουμε να διαβάσουμε (**volitional cue**). Σε σχέση με την πρώτη περίπτωση του φλιτζανιού το οποίο μας προδιαθέτει να το κοιτάξουμε παρά την θέλησή μας, στην δεύτερη περίπτωση η προσοχή που δίνουμε στο βιβλίο είναι σκόπιμη και συγκεντρώνει περισσότερο την προσοχή μας σε αυτό που θέλουμε να κάνουμε.

Ας μεταφράσουμε τώρα το παραπάνω παράδειγμα σε όρους του μηχανισμού προσοχής για ενα νευρωνικό δίκτυο. Στο πλαίσιο των νευρωνικών δικτύων, η περίπτωση της επιλογής του βιβλίου με την δικία μας βούληση αναφέρετε ως ερώτημα (**query**). Δεδομένου ενός ερωτήματος, ο μηχανισμός προσοχής προδιαθέτει την επιλογή μας μέσω μιας διαδικασίας συγκέντρωσης της προσοχής (**attention pooling**) σε ενα σύνολο από δεδομένα εισόδου (**values**). Κάθε δεδομένο εισόδου συνδέεται με την τιμή ενός κλειδιού (**key**), το οποίο είναι η ακούσια επιλογή μας.



Εικόνα 31 - Μηχανισμός Προσοχής

Η εικόνα 31 παρουσιάζει το μηχανισμό προσοχής, στον οποίο το ερώτημα αλληλεπιδρά με τα κλειδιά με σκοπό να προδιαθέσουν την επιλογή μιας τιμής.

Μετασχηματιστές

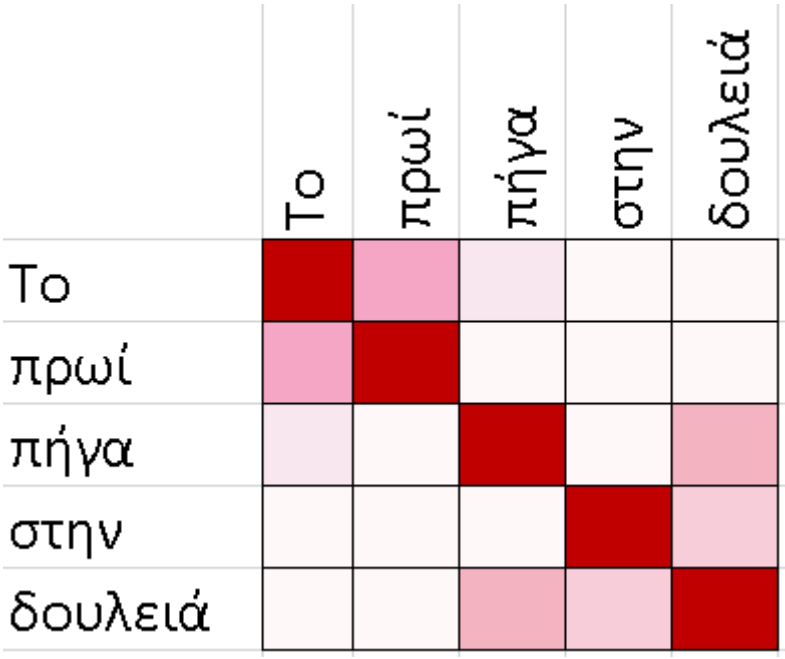
Όπως έχουμε ήδη αναφέρει, οι μετασχηματιστές δεν επεξεργάζονται τα δεδομένα της ακολουθίας σειριακά, όπως τα κλασσικά RNN, άλλα επεξεργάζονται κάθε στοιχείο των δεδομένων εισόδου ξεχωριστά. Αυτή η ιδιότητά τους έχει ως αποτέλεσμα το μοντέλο να έχει πρόσβαση σε όλα τα στοιχεία εισόδου μέχρι και το τρέχων στοιχείο και επιπλέον λόγω αυτής της ανεξαρτησίας στην επεξεργασία των δεδομένων εισόδου μπορούμε να παραλληλοποιήσουμε τους υπολογισμού κερδίζοντας σε αποδοτικότητα.

Το δομικό στοιχείο που έρχονται να προσθέσουν οι μετασχηματιστές είναι το επίπεδο αυτοπροσοχής (**self-attention layer**). Τέτοια επίπεδα δίνουν τη δυνατότητα στο δίκτυο να εξάγει και να χρησιμοποιεί την πληροφορία από ακολουθίες μεγάλου μήκους χωρίς να χρειάζεται να τροφοδοτεί την πληροφορία σε αναδρομικά επίπεδα όπως στην περίπτωση των RNN. Μέσω των επιπέδων αυτοπροσοχής, τα δίκτυα μπορούν να εξάγουν την σχέση που έχουν τα διάφορα στοιχεία της ακολουθίας μεταξύ τους και μέσω αυτής της διαδικασίας να παράγουν το αποτέλεσμα για τα δεδομένα εισαγωγής.

Ας θεωρήσουμε τους εξής ρόλους για τα στοιχεία του μηχανισμού προσοχής που είδαμε στην προηγούμενη υποενότητα: Το ερώτημα (query) έχει τον ρόλο του στοιχείου στο οποίο είναι συγκεντρωμένη η πρόσοχή του δικτύου, το κλειδί (key) έχει τον ρόλο του προηγούμενου στοιχείου εισαγωγής και η τιμή (value) είναι το στοιχείο που συμμετέχει στο αποτέλεσμα που εξάγεται. Αυτοί οι ρόλοι προκύπτουν όταν το διάνυσμα εισαγωγής συνδυάζεται (πολλαπλασιάζεται) με τρεις διαφορετικές μήτρες βαρών: $\mathbf{W^Q}$, $\mathbf{W^K}$, $\mathbf{W^V}$. Μέσω του πολλαπλασιασμού αυτού, το διάνυσμα εισαγωγής συμμετέχει σε τρεις διαφορετικούς ρόλους ως το ερώτημα, το κλειδί και η τιμή.

$$\boldsymbol{q}_i = \boldsymbol{W^Q x}_i \, , \boldsymbol{k}_i = \boldsymbol{W^K x}_i \, , \, \boldsymbol{v}_i = \boldsymbol{W^V x}_i \qquad (2.12)$$

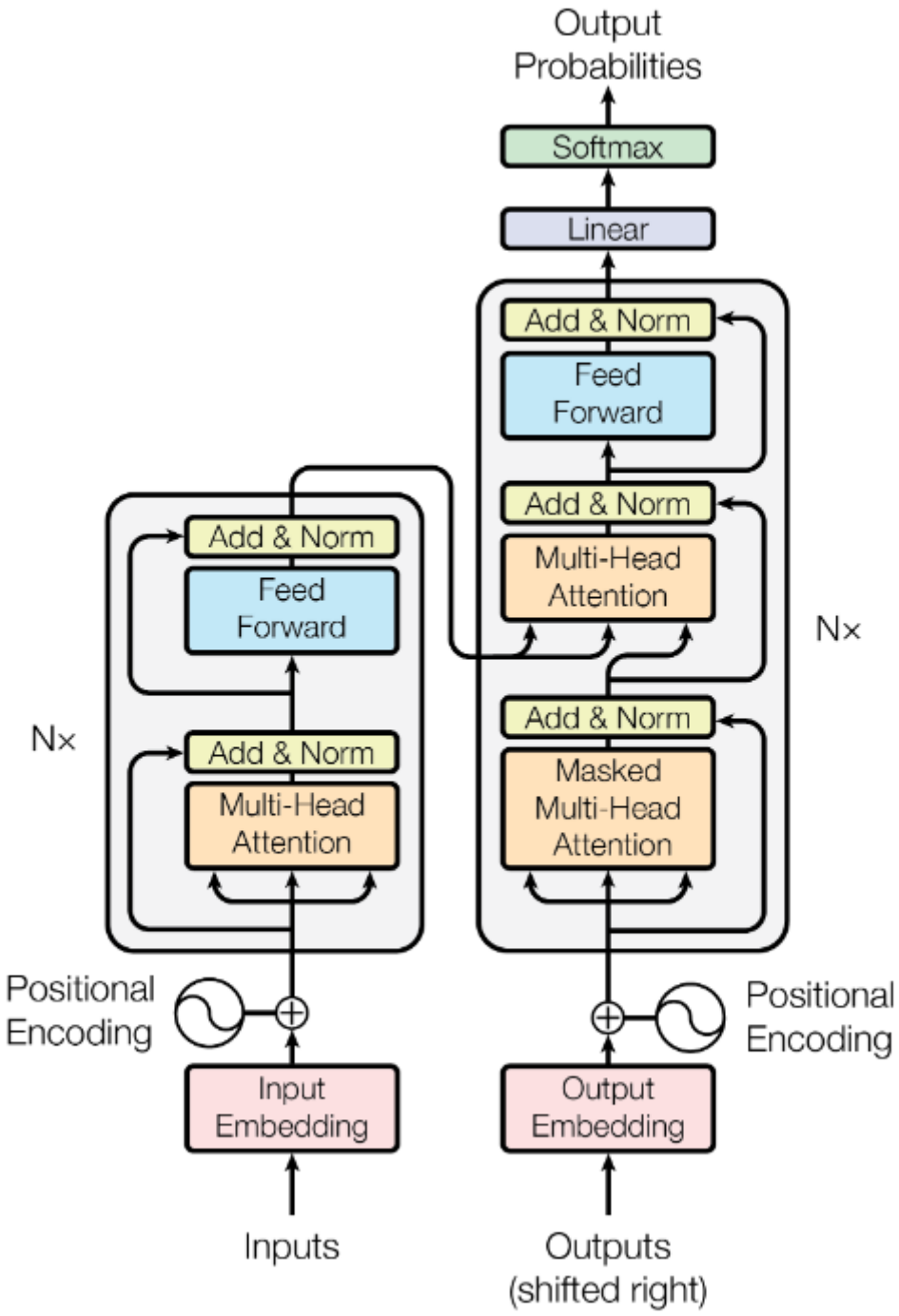
Μέσω αυτών των ρόλων μπορούμε να εξάγουμε πληροφορία που να μας δίνει ενα μέγεθος συσχέτισης μεταξύ του στοιχείου στο οποίο είναι συγκεντρωμένη η προσοχή, $\boldsymbol{x}_i$ και ενός παρελθοντικού στοιχείου, $\boldsymbol{x}_j$. Στην εικόνα 32 μπορούμε να δούμε μια απεικόνιση των βαρών του μηχανισμού προσοχής για την περίπτωση όπου η ακολουθία είναι μια πρόταση. Συγκεκριμένα η πρόταση είναι η εξής: «Το πρωί πήγα στην δουλειά». Όσο πιο έντονα είναι τα χρώματα μέσα στα κελιά, τόσο μεγαλύτερη είναι η συσχέτιση μεταξύ των λέξεων.



Εικόνα 32 - Απεικόνιση των βαρών του μηχανισμού προσοχής

Η εικόνα 32 απεικονίζει το αποτέλεσμα του πολλαπλασιασμού του μεταξύ των ερωτημάτων και των κλειδιών. Στη συνέχεια, το αποτέλεσμα αυτό πολλαπλασιάζεται με τις τιμές για να παραχθεί το τελικό αποτέλεσμα.

Στην εικόνα 33 φαίνεται η αρχιτεκτονική που χρησιμοποιούν οι μετασχηματιστές, όπως παρουσιάστηκε στο πρωτότυπο επιστημονικό άρθρο (αναφορά στην βιβλιογραφία).



Εικόνα 33 - Η αρχιτεκτονική των μετασχηματιστών

Επίλογος

Στην παρούσα εργασία παρουσιάσαμε τις αδυναμίες των κλασικών αρχιτεκτονικών νευρωνικών δικτύων, όπως είναι τα πολυεπίπεδα δίκτυα Perceptron, να μοντελοποιήσουν δεδομένα ακολουθιών και τους λόγους για τους οποίους υπήρξε η ανάγκη να σχεδιαστούν νέα μοντέλα που θα φέρνανε εις πέρας αυτό το εγχείρημα.

Επίσης, εστιάσαμε σε κάποιες βασικές αρχιτεκτονικές, οι οποίες είναι το αντικείμενο της σύγχρονης έρευνας, για την μοντελοποίηση δεδομένων ακολουθιών. Τέτοιες αρχιτεκτονικές είναι τα αναδρομικά νευρωνικά δίκτυα (RNN), τα μοντέλα μακράς βραχείας μνήμης. Είδαμε τους λόγους για τους οποίους τα RNN στην γενική τους μορφή χρησιμοποιήθηκαν σε μεγάλο βαθμό για τη μοντελοποίηση ακολουθιών και συζητήσαμε τα μειονεκτήματα των μοντέλων αυτών σε περιπτώσεις που οι ακολουθίες έχουν μεγάλο μήκος.

Στην συνέχεια, βάλαμε στην συζήτηση μια υποομάδα των αναδρομικών νευρωνικών δικτύων, τα δίκτυα μακράς βραχείας μνήμης ή LSTM, τα οποία έχουν ως δομικό τους στοιχείο διάφορες μονάδες πυλών, μέσω των οποίων τα LSTM ξεπερνάνε, εν μέρει, τις αδυναμίες των κλασικών RNN και επιτυγχάνουν καλύτερα αποτελέσματα. Επίσης, είδαμε πως ακόμα και τα LSTM, παρόλο που έχουν πολλές περισσότερες δυνατότητες από τα απλά RNN μοντέλα, αδυνατούν και αυτά να λύσουν ολοκληρωτικά το πρόβλημα των συσχετίσεων μεταξύ των στοιχείων ακολουθιών με πολύ μεγάλο μήκος.

Τέλος, παρουσιάσαμε συνοπτικά μια σχετικά νεα αρχιτεκτονική νευρωνικών δικτύων, τους μετασχηματιστές. Τα μοντέλα αυτά έχουν κερδίσει σημαντικό έδαφος στο αντικείμενο της μοντελοποίησης ακολουθιών, αφού, αρχικά, μέσω της αρχιτεκτονικής τους αποσύρουν τις αναδρομικές σχέσεις μεταξύ των επιπέδων του δικτύου, με αποτέλεσμα να μπορούμε να έχουμε μια πολύ πιο γρήγορη εκπαίδευση του δικτύου μέσω της παραλληλοποίησης, και, επιπλέον, με τη χρήση των μηχανισμών προσοχής καταφέρνουν να αποθηκεύουν τις συσχετίσεις μεταξύ των στοιχείων ακόμα και πολύ μεγάλου μήκους ακολουθιών.

Βιβλιογραφία

Francois Chollet. (2021). *Deep Learning with Python. Shellter Island, NY. Manning.*

Goodfellow I., Bengio Y., Courville A. (2016). *Deep Learning*. London, England. The MIT Press.

Alex Graves. *Supervised sequence labelling with recurrent neural networks*, volume 385. Springer, 2012.

Lipton C. Z., Berkowitz J. *A critical review of recurrent neural networks for sequence learning*. *CoRR*, abs/1506.00019, 2015.

Zhang A., Lipton C. Z., Li M. Smola A. J. (2021). *Dive into Deεp Learning. https://d2l.ai/.* Release 0.16.6.

Karpathy Andrej, Johnson Justin, Fei-Fei Li. (2015). *Visualizing and Understanding Recurrent Networks.* https://arxiv.org/pdf/1506.02078.pdf

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. *On the difficulty of training recurrent neural networks*. *arXiv preprint arXiv:1211.5063*, 2012.

Alexander Amini and Ava Soleimany. *MIT 6.S191: Introduction to Deep Learning.* IntroToDeepLearning.com

Daniel Jurafsky and James H. Martin (2022). *Speech and Language Processing*. https://web.stanford.edu/~jurafsky/slp3/9.

Yoshua Bengio, Patrice Simard, and Paolo Frasconi. *Learning long-term dependencies with gradient descent is difficult*. Neural Networks, IEEE Transactions on, 5(2):157{166, 1994.

Christopher Olah. (2015). *Understanding lstm networks*. http://colah.github.io/posts/2015-08-Understanding-LSTMs/.

Karpathy Andrej. (2015). *The Unreasonable Effectiveness of Recurrent Neural Networks*. http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N., Kaiser L., Polosukhin I. (2017). *Attention is All You Need.* 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

Jason Brownlee. (2021). *The Transformer Model*. https://machinelearningmastery.com/the-transformer-model/.