1ο σετ ασκήσεων

Δεκέμβριος 2024

Άσκηση

Ενώ οι παραδοσιακές μέθοδοι νευρωνικών δικτύων, όπως τα CNNs, λειτουργούν σε δεδομένα με δομημένη μορφή (π.χ. εικόνες ή κείμενο), τα Graph Convolutional Networks (GCNs) είναι σχεδιασμένα για δεδομένα που περιγράφονται με τη χρήση γράφων. Η βασική αρχή πίσω από τα GCNs είναι η μάθηση σύνθετων αναπαραστάσεων για κάθε κόμβο του γράφου, ενσωματώνοντας τόσο τα χαρακτηριστικά του κόμβου όσο και τη δομή του δικτύου. Αυτό επιτυγχάνεται μέσω της αλληλεπίδρασης ενός κόμβου με τους γείτονές του, όπου οι πληροφορίες διαδίδονται και συνδυάζονται σε πολλαπλά επίπεδα (layers).

Τα χαρακτηριστικά όλων των κόμβων οργανώνονται σε έναν πίνακα H για κάθε επίπεδο του δικτύου, στον οποίο κάθε γραμμή αντιστοιχεί σε έναν κόμβο του γράφου και περιέχει τα χαρακτηριστικά του. Η συγκέντρωση των χαρακτηριστικών των γειτόνων κάθε κόμβου επιτυγχάνεται μέσω του πολλαπλασιασμού του πίνακα γειτνίασης A με τον πίνακα των χαρακτηριστικών εισόδου H_{in}. Σε κάθε επίπεδο ενός GCN, θα πρέπει αυτή η συγκεντρωμένη πληροφορία να πολλαπλασιαστεί με έναν πίνακα εκπαιδεύσιμων βαρών W, ώστε να προκύψουν τα χαρακτηριστικά εξόδου H_{out} κάθε κόμβου. Ο πίνακας H_{out} θα τροφοδοτηθεί στο επόμενο επίπεδο, ως H_{in}, προκειμένου μετά από κάποια επίπεδα να παραχθεί η τελική ταξινόμηση των κόμβων.

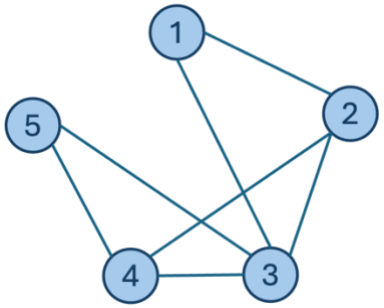
Η διαδικασία υπολογισμού της εξόδου ενός επιπέδου ενός GCN, μπορεί να συνοψιστεί στον πολλαπλασιασμό τριών πινάκων, όπως φαίνεται στην παρακάτω εξίσωση:

$$H_{out} = \sigma\left(\tilde{A} H_{in} W\right)$$

Όπου

- $\sigma()$ μία συνάρτηση ενεργοποίησης, συνήθως χρησιμοποιείται η ReLU για τα ενδιάμεσα επίπεδα και η SoftMax για το τελευταίο
- H είναι ο πίνακας χαρακτηριστικών, μεγέθους $N \times K$
- W είναι ο πίνακας των εκπαιδεύσιμων βαρών του επιπέδου, μεγέθους $K \times M$
- $\tilde{A}$ είναι ο κανονικοποιημένος πίνακας γειτνίασης, μεγέθους $N \times N$, ο οποίος μπορεί να υπολογιστεί ως $\tilde{A} = D^{-\frac{1}{2}}(A + I)D^{-\frac{1}{2}}$, όπου I ο μοναδιαίος πίνακας και D ο διαγώνιος πίνακας βαθμών

Για παράδειγμα, έστω ότι έχουμε τον παρακάτω γράφο με πίνακα γειτνίασης A



$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

Προκειμένου ο κάθε κόμβος να κρατάει την πληροφορία των χαρακτηριστικών του και τον υπολογισμό του επιπέδου του GCN, προσθέτουμε self-loops σε κάθε κόμβο, προσθέτοντας τον μοναδιαίο στον πίνακα γειτνίασης, οπότε έχουμε:

$$A' = A + I = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

Ο πίνακας των βαθμών D, τότε θα είναι

$$D = \begin{bmatrix} 3 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 & 3 \end{bmatrix}$$

Άρα $D^{-1/2}$ είναι ο διαγώνιος πίνακας με αντίστροφες τετραγωνικές ρίζες των βαθμών:

$$D^{-1/2} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{4}} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{5}} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\sqrt{4}} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{\sqrt{3}} \end{bmatrix}$$

Οπότε ο κανονικοποιημένος πίνακας γειτνίασης θα είναι:

$$\tilde{A} = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{4}} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{5}} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\sqrt{4}} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{\sqrt{3}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{4}} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{5}} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\sqrt{4}} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{\sqrt{3}} \end{bmatrix}$$

$$= \begin{bmatrix} 0.33 & 0.29 & 0.26 & 0.00 & 0.00 \\ 0.29 & 0.25 & 0.22 & 0.20 & 0.00 \\ 0.26 & 0.22 & 0.20 & 0.18 & 0.15 \\ 0.00 & 0.20 & 0.18 & 0.22 & 0.20 \\ 0.00 & 0.00 & 0.15 & 0.20 & 0.33 \end{bmatrix}$$

Στην άσκηση αυτή καλείστε να υλοποιήσετε τον υπολογισμό ενός επιπέδου ενός GCN, στο οποίο θα χρησιμοποιείτε τη ReLU ($ReLU(x) = \max(0, x)$) ως συνάρτηση ενεργοποίησης. Δηλαδή θα πρέπει να υλοποιήσετε τον υπολογισμό της εξίσωσης:

$$H_{out} = ReLU\left(\tilde{A} H_{in} W\right)$$

Πιο συγκεκριμένα, σε κάθε στοιχείο του γινομένου των τριών πινάκων θα πρέπει να εφαρμόσετε τη συνάρτηση ReLU. Για ευκολία θεωρήστε ότι στην είσοδο του κυκλώματός σας θα λαμβάνετε έτοιμο τον κανονικοποιημένο πίνακα γειτνίασης $\tilde{A}$. Καθώς και ότι ο πίνακας βαρών σας δίνεται έτοιμος στο αρχείο weights.txt μαζί με τη συνάρτηση ανάγνωσης από txt σε δισδιάστατο πίνακα στο αρχείο read_txt.h, που συμπεριλαμβάνονται στο hw-1-2024.zip μαζί με την παρούσα εκφώνηση. Εσείς θα πρέπει στο testbench σας να φτιάξετε μόνο τον πίνακα $\tilde{A}$ για έναν τυχαίο μη-κατευθυνόμενο γράφο, καθώς και τον πίνακα H_{in} των χαρακτηριστικών εισόδου για κάθε κόμβο του γράφου που φτιάξατε. Θεωρείστε ότι κάθε χαρακτηριστικό μπορεί να πάρει οποιαδήποτε ακέραια τιμή στο εύρος [0, 255] και ότι ο πίνακας H_{in} είναι μεγέθους $N \times 500$, ενώ ο W είναι μεγέθους 500×16.

Το κύκλωμα που θα σχεδιάσετε θα πρέπει να έχει τα εξής χαρακτηριστικά:

- Η διεπαφή του κυκλώματος θα περιλαμβάνει **μια διεπαφή μνήμης** των 16bit (signed integer), όπου ο κάθε πίνακας είναι αποθηκευμένος μονοδιάστατα με τα στοιχεία του να είναι διαδοχικά τοποθετημένα σε μία μνήμη (αυτό είναι η τυπική επιλογή). Οι διεπαφές αυτές θα πρέπει να είναι τύπου 1R1W.
- Η συχνότητα λειτουργίας να είναι 500MHz στην τεχνολογία των 45nm

Θεωρώντας ότι οι πόρτες των μνημών έχουν τέτοιο πλάτος ώστε να μπορούν είτε να γράφουν είτε να διαβάζουν 8 λέξεις σε κάθε κύκλο, στόχος σας είναι το κύκλωμα που θα σχεδιάσετε να πετυχαίνει το μικρότερο δυνατό χρόνο εκτέλεσης. Για να το πετύχετε αυτό θα πρέπει, αφού ορίσετε το κατάλληλο interleave ή block size στις μνήμες, να κάνετε κατάλληλα unroll τις επαναλήψεις.

Παράλληλα με το κύκλωμα σας φροντίσετε να δημιουργήσετε ένα ευέλικτο πρόγραμμα ελέγχου σε C++, ώστε να ελέγξετε την ορθή λειτουργία του κυκλώματός σας καθώς και να σιγουρευτείτε πως το κύκλωμά σας περνάει από RTL προσομοίωση (cosimulation).

Μαζί με τους κώδικες που υλοποιήσατε, ετοιμάστε και ένα report (και 2 slides αρκούν) όπου θα φαίνεται η βελτιστοποίηση του χρόνου εκτέλεσης του κυκλώματος καθώς και στιγμιότυπα από την επιτυχημένη προσομοίωση.