Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων
Τομεας

Εκτίμηση αστοχιών κρυφής μνήμης

Πρόβλεψη αστοχιών κρυφής μνήμης σε διάφορες αρχιτεκτονικές κρυφών μνημών

Διπλωματικη Εργασια

του

ΡΑΜΟΥ Γ. ΜΑΡΚΟΥ

Επιβλέπων: Νεκτάριος Κοζύρης
Καθηγητής

Αθήνα, Απρίλιος 2024

Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων
Τομεας

Εκτίμηση αστοχιών κρυφής μνήμης

Πρόβλεψη αστοχιών κρυφής μνήμης σε διάφορες αρχιτεκτονικές κρυφών μνημών

Διπλωματικη Εργασια

του

ΡΑΜΟΥ Γ. ΜΑΡΚΟΥ

Επιβλέπων: Νεκτάριος Κοζύρης
Καθηγητής

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 5 Απριλίου 2024.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Νεκτάριος Κοζύρης	Διονύσιος Πνευματικάτος	Γεώργιος Γκούμας
Καθηγητής	Καθηγητής	Αναπληρωτής Καθηγητής

Αθήνα, Απρίλιος 2024

Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων
Τομεας

ΔΗΛΩΣΗ ΜΗ ΛΟΓΟΚΛΟΠΗΣ ΚΑΙ ΑΝΑΛΗΨΗΣ ΠΡΟΣΩΠΙΚΗΣ ΕΥΘΥΝΗΣ

Με πλήρη επίγνωση των συνεπειών του νόμου περί πνευματικών δικαιωμάτων, δηλώνω ενυπογράφως ότι είμαι αποκλειστικός συγγραφέας της παρούσας Πτυχιακής Εργασίας, για την ολοκλήρωση της οποίας κάθε βοήθεια είναι πλήρως αναγνωρισμένη και αναφέρεται λεπτομερώς στην εργασία αυτή. Έχω αναφέρει πλήρως και με σαφείς αναφορές, όλες τις πηγές χρήσης δεδομένων, απόψεων, θέσεων και προτάσεων, ιδεών και λεκτικών αναφορών, είτε κατά κυριολεξία είτε βάσει επιστημονικής παράφρασης. Αναλαμβάνω την προσωπική και ατομική ευθύνη ότι σε περίπτωση αποτυχίας στην υλοποίηση των ανωτέρω δηλωθέντων στοιχείων, είμαι υπόλογος έναντι λογοκλοπής, γεγονός που σημαίνει αποτυχία στην Πτυχιακή μου Εργασία και κατά συνέπεια αποτυχία απόκτησης του Τίτλου Σπουδών, πέραν των λοιπών συνεπειών του νόμου περί πνευματικών δικαιωμάτων. Δηλώνω, συνεπώς, ότι αυτή η Πτυχιακή Εργασία προετοιμάστηκε και ολοκληρώθηκε από εμένα προσωπικά και αποκλειστικά και ότι, αναλαμβάνω πλήρως όλες τις συνέπειες του νόμου στην περίπτωση κατά την οποία αποδειχθεί, διαχρονικά, ότι η εργασία αυτή ή τμήμα της δεν μου ανήκει διότι είναι προϊόν λογοκλοπής άλλης πνευματικής ιδιοκτησίας.

(Υπογραφή)

.....
Μάρκος Γεώργιος Ραμός

5 Απριλίου 2024

Περίληψη

Δεδομένης της αυξανόμενης διαφοράς στην ταχύτητα επεξεργασίας μεταξύ της κύριας μνήμης και της ΚΜΕ, ο ρόλος των κρυφών μνημών της ΚΜΕ στην επίτευξη της μέγιστης δυνατής ταχύτητας επεξεργασίας είναι τόσο σημαντικός όσο ποτέ άλλοτε. Κάθε πρόσβαση στην κρυφή μνήμη που δεν βρίσκει τα δεδομένα πρέπει να καλεί την κύρια μνήμη, χρησιμοποιώντας δεκάδες, αν όχι εκατοντάδες, κύκλους της μηχανής. Η πρόβλεψη της συμπεριφοράς μιας διεργασίας πριν από την εκτέλεσή της σε διαφορετικές αρχιτεκτονικές κρυφής μνήμης είναι ένα σημαντικό έργο. Θα βοηθήσει να καθοριστεί ποιος επεξεργαστής θα λειτουργήσει καλύτερα για ένα πρόγραμμα ή πώς να κατανεμηθεί βέλτιστα η υπολογιστική ισχύς. Ο άμεσος στόχος αυτής της διατριβής είναι να προτείνει ένα μοντέλο μηχανικής μάθησης που θα προβλέπει με όσο το δυνατόν μεγαλύτερη ακρίβεια τις αστοχίες στην κρυφή μνήμη ενός προγράμματος, ανάλογα αποκλειστικά με την αρχιτεκτονική της κρυφής μνήμης, τον κώδικα του προγράμματος και τα ιστογράμματα των αποστάσεων επαναχρησιμοποίησης. Για την επίτευξη αυτού του στόχου, προτείνουμε έναν αριθμό μοντέλων μετασχηματιστών που συνδυάζουν τον μετασχηματισμένο κώδικα εισόδου με πληροφορίες μέσα στα ιστογράμματα αποστάσεων επαναχρησιμοποίησης και εκτιμούν το ποσοστό αστοχιών κρυφής μνήμης. Εκπαιδεύσαμε και δοκιμάσαμε αυτά τα μοντέλα με προσομοιωμένα δεδομένα και παρήγαμε προβλέψεις υψηλής ακρίβειας για μεγάλο αριθμό πολιτικών αντικατάστασης και μεγεθών κρυφής μνήμης LLC. Αυτά τα δίκτυα μπορούν να λειτουργήσουν ως ένα χρήσιμο εργαλείο στην πρόβλεψη της κρυφής μνήμης και μπορούν να χρησιμοποιηθούν σε πραγματικές μηχανές σε μελλοντική έρευνα.

Abstract

Given the increasing difference in processing speed between the main memory and the CPU, the role of CPU caches in achieving the maximum possible processing speed is as big as ever. Every cache access that doesn't find the data has to invoke the main memory, using tens, if not hundreds, of machine cycles. Predicting the behavior of a process prior to execution on different cache architectures is an important task. It will help determine what processor will work best for a program or how to distribute computing power optimally. The immediate goal of this thesis is to propose a machine learning model that will predict as accurately as possible the cache misses of a program depending solely on the cache architecture, the program code, and the reuse-distance histograms. To achieve this goal, we propose a number of transformer models that combine the transformed input code with information within the reuse-distance histograms and lead to an output cache miss ratio. We trained and tested these models with simulated data and produced high-accuracy predictions for a large number of replacement policies and LLC cache sizes. These networks can act as a useful tool in cache prediction and may be used on real machines in future research.

Περιεχόμενα

Κατάλογος Σχημάτων

Κατάλογος Εικόνων

Κατάλογος Πινάκων

Κεφάλαιο 1

Εισαγωγή

Ας εμβαθύνουμε στη δυναμική σφαίρα όπου η αρχιτεκτονική ενός υπολογιστή δεν είναι απλώς ένα σχέδιο αλλά ένας καταλύτης, που ασκεί τεράστια επιρροή στα αποτελέσματα των επιδόσεων. Από τον περίπλοκο χορό των επεξεργαστών έως τα δαιδαλώδη μονοπάτια της μνήμης, κάθε αρχιτεκτονική πτυχή ενορχηστρώνει μια συμφωνία που καθορίζει την ταχύτητα, την αποδοτικότητα και τη μετασχηματιστική δύναμη των υπολογιστών. Η κατανόηση αυτής της περίπλοκης αλληλεπίδρασης ανοίγει τις πόρτες για την απελευθέρωση απαράμιλλων δυνατοτήτων απόδοσης, αναδιαμορφώνοντας το ψηφιακό τοπίο με μία αρχιτεκτονική καινοτομία τη φορά.

Κατά τη διαπίστωση του κομβικού ρόλου της αρχιτεκτονικής υπολογιστών στη διαμόρφωση του τοπίου των επιδόσεων, είναι επιτακτική ανάγκη να αναγνωρίσουμε τις περίπλοκες δυναμικές πίσω από αυτή την τεχνολογική συμφωνία. Η αρχιτεκτονική υπολογιστών υπερβαίνει το ρόλο ενός απλού σχεδίου- χρησιμεύει ως ο ακρογωνιαίος λίθος που υπαγορεύει την αποδοτικότητα και τις δυνατότητες της σύγχρονης πληροφορικής. Για να κατανοήσουμε τον βαθύ αντίκτυπό της, είναι απαραίτητο να εξετάσουμε την περίπλοκη ενορχήστρωση των επεξεργαστών και των μονοπατιών μνήμης, διερευνώντας πώς κάθε αρχιτεκτονική πτυχή εναρμονίζεται για να καθορίσει την ταχύτητα, την αποτελεσματικότητα και τις μετασχηματιστικές δυνατότητες στο ψηφιακό πεδίο.

Η αδιάκοπη επιδίωξη ταχύτερων και αποδοτικότερων επεξεργαστών είναι μια ιστορία συνεχούς καινοτομίας και τεχνολογικής προόδου. Από την απαρχή της πληροφορικής, η αναζήτηση της ταχύτητας οδήγησε τους μηχανικούς και τους επιστήμονες να διευρύνουν τα όρια του εφικτού. Αρχικά, οι ΚΜΕ κατασκευάζονταν με τη χρήση βασικών ηλεκτρονικών εξαρτημάτων. Ωστόσο, καθώς η τεχνολογία εξελισσόταν, στα τέλη του 20ού αιώνα σημειώθηκε μια μνημειώδης αλλαγή με την έλευση των ολοκληρωμένων κυκλωμάτων και των μικροεπεξεργαστών. Ο νόμος του Μοορε, που διατυπώθηκε από τον συνιδρυτή της Ιντελ Γορδον Μοορε το 1965, προέβλεψε τον διπλασιασμό του αριθμού των τρανζίστορ σε ένα τσιπ περίπου κάθε δύο χρόνια. Η αρχή αυτή έγινε κατευθυντήρια δύναμη, οδηγώντας τις φιλοδοξίες της βιομηχανίας για ταχεία πρόοδο. Με την πάροδο του χρόνου, η διαδικασία κατασκευής υπέστη σημαντικές βελτιώσεις, μεταβαίνοντας από μεγαλύτερα τρανζίστορ σε μικρότερα, πιο πυκνά τοποθετημένα, οδηγώντας σε αυξημένες ταχύτητες επεξεργασίας και βελτιωμένη αποδοτικότητα. Οι καινοτομίες στην τεχνολογία των ημιαγωγών, όπως η ανάπτυξη τσιπ με βάση το πυρίτιο, η εισαγωγή πολυπύρηνων επεξεργαστών και η βελτίωση των αρχιτεκτονικών μέσω της επένδυσης σωλήνων και της παράλληλης επεξεργασίας, υπήρξαν καθοριστικής σημασίας

για την αδιάκοπη πορεία προς ταχύτερους επεξεργαστές. Επιπλέον, οι πρόοδοι στην επιστήμη των υλικών, τη νανοτεχνολογία και τις μεθοδολογίες σχεδιασμού των τσιπ συνέβαλαν συλλογικά στη συνεχή εξέλιξη των ΚΜΕ, τροφοδοτώντας μια εποχή υπολογιστικής ισχύος που συνεχίζει να επαναπροσδιορίζει τα όρια των τεχνολογικών δυνατοτήτων.

Οι κρυφές μνήμες επεξεργαστή χρησιμεύουν ως ένας κρίσιμος μηχανισμός για την άμβλυνση των σημείων συμφόρησης στις επιδόσεις, ελαχιστοποιώντας τον χρόνο που χρειάζεται ο επεξεργαστής για την πρόσβαση σε συχνά χρησιμοποιούμενα δεδομένα. Όταν μια CPU εκτελεί λειτουργίες, ανακτά συνεχώς δεδομένα και εντολές από τη μνήμη. Ωστόσο, η πρόσβαση σε δεδομένα απευθείας από την κύρια μνήμη μπορεί να είναι σχετικά αργή λόγω της διαφοράς ταχύτητας μεταξύ της ΚΜΕ και της μνήμης. Οι κρυφές μνήμες ενεργούν ως ενδιάμεσος υψηλής ταχύτητας μεταξύ της ΚΜΕ και της κύριας μνήμης, αποθηκεύοντας δεδομένα και εντολές στις οποίες γίνεται συχνή πρόσβαση. Με τον τρόπο αυτό, μειώνουν την καθυστέρηση στην ανάκτηση πληροφοριών που απαιτούνται από την ΚΜΕ. Όταν ο επεξεργαστής χρειάζεται δεδομένα, ελέγχει πρώτα την κρυφή μνήμη. Εάν τα απαιτούμενα δεδομένα βρεθούν στην κρυφή μνήμη (cache hit), μπορεί να γίνει πρόσβαση σε αυτά πολύ πιο γρήγορα από την ανάκτησή τους από την πιο αργή κύρια μνήμη. Με τον τρόπο αυτό αποφεύγεται η ανάγκη αναμονής για δεδομένα από την κύρια μνήμη, με αποτέλεσμα να αμβλύνεται η δυσχέρεια επιδόσεων που προκαλείται από την καθυστέρηση πρόσβασης στη μνήμη. Οι αποτελεσματικά χρησιμοποιούμενες κρυφές μνήμες βελτιστοποιούν την απόδοση της CPU μειώνοντας τον χρόνο αδράνειας, επιτρέποντας ταχύτερη πρόσβαση σε δεδομένα και βελτιώνοντας τη συνολική απόδοση του συστήματος σε διάφορες υπολογιστικές εργασίες.

Οι αστοχίες στην κρυφή μνήμη προκαλούν σημαντικές χρονικές ποινές στις υπολογιστικές ροές εργασίας. Όταν η CPU αναζητά δεδομένα που δεν είναι αποθηκευμένα στην κρυφή μνήμη, προκαλείται αστοχία στην κρυφή μνήμη, γεγονός που ωθεί τον επεξεργαστή να διακόψει την εκτέλεσή του, ενώ ανακτά τις απαιτούμενες πληροφορίες από την πιο αργή κύρια μνήμη. Αυτή η μετάβαση μεταξύ της κρυφής μνήμης και της κύριας μνήμης προκαλεί αξιοσημείωτη καθυστέρηση λόγω της σημαντικής διαφοράς ταχύτητας μεταξύ αυτών των επιπέδων μνήμης. Αυτή η καθυστέρηση, που ονομάζεται ποινή αστοχίας στη μνήμη cache, έχει ως αποτέλεσμα δεκάδες, αν όχι εκατοντάδες, αδρανείς κύκλους επεξεργαστή, παρεμποδίζοντας την ομαλή εκτέλεση των εντολών και δυσχεραίνοντας τη συνολική απόδοση του συστήματος. Ως εκ τούτου, ο εντοπισμός και η διαλεύκανση των σημείων συμφόρησης των επιδόσεων που προκύπτουν από τις αστοχίες της κρυφής μνήμης είναι ζωτικής σημασίας.

Η ποικιλομορφία στα σχέδια, τα μεγέθη, τις στρατηγικές αντιστοίχισης, τις πολιτικές αντικατάστασης και τις τεχνολογικές εξελίξεις μεταξύ των διαφόρων επεξεργαστών έχει ως αποτέλεσμα διαφορετικές συμπεριφορές αστοχίας στην κρυφή μνήμη, γεγονός που καθιστά απαραίτητη την εξέταση των ιδιαίτερων χαρακτηριστικών κάθε αρχιτεκτονικής επεξεργαστή κατά την ανάλυση της απόδοσης της κρυφής μνήμης.

Η συμπεριφορά των αστοχιών της κρυφής μνήμης μπορεί να κυμαίνεται σημαντικά μεταξύ διαφορετικών ΚΜΕ λόγω των διαφοροποιήσεων στην αρχιτεκτονική, το μέγεθος, την οργάνωση και τις πολιτικές πρόσβασης της κρυφής μνήμης. Διαφορετικές ΚΜΕ μπορεί να χρησιμοποιούν διαφορετικές σχεδιάσεις κρυφής μνήμης, όπως διαφορετικά επίπεδα κρυφής μνήμης (Λ1, Λ2, Λ3), μεγέθη κρυφής μνήμης, συσχετιστικότητα, πολιτικές αντικατάστασης και στρατηγικές προφόρτωσης. Αυτές οι αρχιτεκτονικές αποκλίσεις μπορούν να επηρεάσουν

τον τρόπο με τον οποίο συμβαίνουν οι αστοχίες στην κρυφή μνήμη και τις επακόλουθες κυρώσεις τους.

Επιπλέον, οι αρχιτεκτονικές εξελίξεις και οι καινοτομίες σε νεότερες γενιές CPU συχνά εισάγουν βελτιστοποιήσεις με στόχο τη μείωση των αστοχιών της κρυφής μνήμης. Οι βελτιωμένες τεχνικές προφόρτωσης, οι εξυπνότεροι αλγόριθμοι πρόβλεψης ή οι αλλαγές στην ιεραρχία της κρυφής μνήμης μπορούν να επηρεάσουν τον τρόπο με τον οποίο εκδηλώνονται οι αστοχίες κρυφής μνήμης στις νεότερες CPU σε σύγκριση με τις παλαιότερες.

Η προσομοίωση μιας κρυφής μνήμης CPU χρησιμεύει ως απαραίτητο εργαλείο στη σύγχρονη πληροφορική, προσφέροντας κρίσιμες γνώσεις και βελτιστοποιήσεις. Μιμούμενοι τη συμπεριφορά των μηχανισμών προσωρινής αποθήκευσης, οι προσομοιώσεις επιτρέπουν τη σε βάθος ανάλυση των επιδόσεων της κρυφής μνήμης, βοηθώντας στο σχεδιασμό, την αξιολόγηση και τη βελτιστοποίηση των αρχιτεκτονικών κρυφής μνήμης. Αυτές οι προσομοιώσεις διευκολύνουν τη διερεύνηση ποικίλων διαμορφώσεων κρυφής μνήμης, πολιτικών αντικατάστασης και συσχετισμού, επιτρέποντας στους μηχανικούς να επιλέξουν τις πιο αποδοτικές ρυθμίσεις για συγκεκριμένες εφαρμογές. Επιπλέον, οι προσομοιώσεις κρυφής μνήμης προσφέρουν μια βαθύτερη κατανόηση του τρόπου με τον οποίο οι διάφοροι φόρτοι εργασίας αλληλεπιδρούν με την κρυφή μνήμη και επηρεάζουν τις αστοχίες της κρυφής μνήμης.

Η διαδικασία προσομοίωσης κρυφής μνήμης CPU μπορεί να είναι χρονοβόρα λόγω διαφόρων παραγόντων. Η πολυπλοκότητα που συνεπάγεται η προσομοίωση της συμπεριφοράς της κρυφής μνήμης απαιτεί σημαντικούς υπολογιστικούς πόρους και χρόνο. Ο χειρισμός μεγάλων ιχνών προτύπων προσπέλασης μνήμης για ακριβείς προσομοιώσεις αυξάνει τον χρόνο που απαιτείται για την επεξεργασία. Επιπλέον, η ενσωμάτωση περίπλοκων σχεδίων κρυφής μνήμης, όπως πολλαπλά επίπεδα κρυφής μνήμης (L1, L2, L3), ποικίλα μεγέθη κρυφής μνήμης, συστήματα συσχετισμού και προηγμένοι αλγόριθμοι κρυφής μνήμης, συμβάλλει στην υπολογιστική πολυπλοκότητα και τη διάρκεια των προσομοιώσεων. Ως εκ τούτου, αυτές οι προσομοιώσεις αποδεικνύονται εξαιρετικά χρονοβόρες και απαιτείται ένας καλύτερος τρόπος υπολογισμού του ποσοστού αστοχίας.

Η λύση σε αυτό το πρόβλημα είναι οι μηχανισμοί πρόβλεψης. Η σημασία ενός μοντέλου που μπορεί να προβλέψει με ακρίβεια το μισς ρατιο ενός φόρτου εργασίας χωρίς να έχει το βάρος της προσομοίωσης κάθε πτυχής του είναι τεράστια. Σχεδόν όλα τα υπάρχοντα μοντέλα πρόβλεψης του μισς ρατιο έχουν δημιουργηθεί για πολιτικές LRU ή τυχαίας αντικατάστασης [1, 2, 3]. Το μόνο υπάρχον μοντέλο [1] που μπορεί να χρησιμοποιήσει άλλες πολιτικές αντικατάστασης εκτός της LRU, πρέπει να γνωρίζει, πριν από την εκτέλεση, αναλυτικά όλες τις λεπτομέρειες διαμόρφωσης. Αυτό αφήνει ένα μεγάλο μέρος του προβλήματος αναπάντητο, αφού οι περισσότεροι σύγχρονοι επεξεργαστές CPU δεν χρησιμοποιούν την πολιτική αντικατάστασης LRU στις κρυφές μνήμες τους και, στις περισσότερες περιπτώσεις, δεν γνωρίζουμε ποια πολιτική αντικατάστασης χρησιμοποιείται.

Στην παρούσα διατριβή αντιμετωπίζουμε αυτό το πρόβλημα της πρόβλεψης των αστοχιών της κρυφής μνήμης σε οποιαδήποτε δεδομένη αρχιτεκτονική με τη χρήση μηχανικής μάθησης. Προτείνουμε έναν αριθμό διαφορετικών δικτύων που έχουν ως στόχο την όσο το δυνατόν ακριβέστερη εκτίμηση των αναλογιών αστοχίας στην κρυφή μνήμη. Εμπνευσμένοι από την ήδη υπάρχουσα πιθανολογική προσέγγιση σε αυτό το ζήτημα, γνωστή και ως Στατἄςηε 2.2, αποφασίσαμε να χρησιμοποιήσουμε τα ίδια δεδομένα που χρησιμοποιεί για την πρόβλεψή

του για να εκπαιδεύσουμε και να αξιολογήσουμε ένα δίκτυο μηχανικής μάθησης. Η σημερινή αύξηση της δημοτικότητας των δικτύων NLP μας παρακίνησε να προσθέσουμε έναν κλάδο μετασχηματιστή στα δίκτυα, ο οποίος χρησιμοποιεί αυτή τη νέα τεχνολογία κατανόησης του πηγαίου κώδικα, για να επιτύχουμε καλύτερα αποτελέσματα.

Δημιουργήσαμε συνολικά τρεις διαφορετικές αρχιτεκτονικές δικτύων που προβλέπουν το ποσοστό αστοχίας στην κρυφή μνήμη οποιουδήποτε προγράμματος για καθορισμένα μεγέθη κρυφής μνήμης με οποιαδήποτε πολιτική αντικατάστασης. Αυτά τα δίκτυα αποδίδουν πολύ καλά για οποιαδήποτε πολιτική αντικατάστασης. Εισάγαμε επίσης ένα που προβλέπει τα ποσοστά αστοχίας για οποιοδήποτε μέγεθος κρυφής μνήμης και οποιαδήποτε πολιτική αντικατάστασης, τη λειτουργία του οποίου παρουσιάσαμε στο LRU. Στην περίπτωση του LRU, όλα αυτά τα δίκτυα κατανοούν το πρόβλημα σημαντικά καλύτερα από την υπάρχουσα πιθανολογική προσέγγιση του ζητήματος.

Η παρούσα διατριβή είναι δομημένη σε πέντε κύρια κεφάλαια, καθένα από τα οποία συμβάλλει μοναδικά στη διερεύνηση και ανάλυση των αναλογιών αστοχίας. Το Κεφάλαιο 1 εισάγει τις θεμελιώδεις έννοιες και το θεωρητικό πλαίσιο που υποστηρίζει τη μελέτη, παρουσιάζοντας μια επισκόπηση του ιστορικού πλαισίου και της σημασίας της πρόβλεψης των αστοχιών της κρυφής μνήμης. Το κεφάλαιο 2 εμβαθύνει στην υπάρχουσα βιβλιογραφία, εξετάζοντας κριτικά προηγούμενες έρευνες που θα βοηθήσουν στην κατανόηση των ζητημάτων που προσπαθεί να αντιμετωπίσει η παρούσα διατριβή. Το κεφάλαιο 3 περιγράφει την ερευνητική μεθοδολογία που υιοθετήθηκε, περιγράφοντας λεπτομερώς την επιλεγμένη προσέγγιση, τις μεθόδους συλλογής δεδομένων και τις αρχιτεκτονικές δικτύου που χρησιμοποιήθηκαν. Επίσης, κατέχει ξεχωριστά τα αποτελέσματα για κάθε ένα από τα δίκτυα που θα προτείνουμε. Στο κεφάλαιο 4 παρουσιάζονται τα συλλεχθέντα αποτελέσματα που προέκυψαν από τα δίκτυά μας, προσφέροντας μια ανάλυση και σύγκριση των δεδομένων που συλλέχθηκαν. Τέλος, το Κεφάλαιο 5 συνθέτει τα ευρήματα, συζητά τις επιπτώσεις τους και προσφέρει συμπεράσματα μαζί με συστάσεις για μελλοντική έρευνα πρόβλεψης του ποσοστού αστοχίας στην κρυφή μνήμη.

Συνοψίζοντας, η παρούσα διατριβή έχει ως στόχο να χρησιμοποιήσει τις τρέχουσες εξελίξεις στη μηχανική μάθηση προκειμένου να προβλέψει τα ποσοστά αστοχίας κρυφής μνήμης οποιουδήποτε προγράμματος σε οποιαδήποτε αρχιτεκτονική κρυφής μνήμης. Με τη διερεύνηση πιθανών λύσεων σε αυτό το πρόβλημα, όπως η πιθανολογική προσέγγιση ή οι αλγόριθμοι NLP, η παρούσα μελέτη σκοπεύει να ρίξει φως στη δυσκολία και τις πολλές πτυχές της ακριβούς πρόβλεψης των ςαςηε μισσες. Τα επόμενα κεφάλαια θα εξετάσουν λεπτομερώς αυτές τις έννοιες. Το κεφάλαιο 2 θα επικεντρωθεί σε σχετικές εργασίες που έχουν γίνει, ενώ το κεφάλαιο 3 θα εμβαθύνει στην πρότασή μας για την επίλυση αυτού του ζητήματος. Αυτή η δομημένη προσέγγιση θα παρέχει μια ολοκληρωμένη κατανόηση της πρόβλεψης των αστοχιών της κρυφής μνήμης. Τώρα, ας προχωρήσουμε στη βαθύτερη εμβάθυνση σε αυτές τις περιοχές, ξεκινώντας με την πιθανολογική προσέγγιση StatCache.

Μέρος Ι

Θεωρητικό Μέρος

Κεφάλαιο 2

Θεωρητικό υπόβαθρο

2.1 Ιεραρχία κρυφών μνημών

Στην ιστορία της ανάπτυξης των υπολογιστών, οι ταχύτητες της CPU ξεπερνούσαν τις ταχύτητες πρόσβασης στη μνήμη. Αυτή η διαφορά οδηγούσε σε αδρανείς ΚΜΕ που περίμεναν δεδομένα από την κύρια μνήμη. Εισάγεται η κρυφή μνήμη, μια ιδέα που προτάθηκε για πρώτη φορά από τον Βρετανό επιστήμονα υπολογιστών Maurice Wilkes το 1965 [4]. Τα πρώιμα μοντέλα κρυφής μνήμης βελτίωσαν την καθυστέρηση πρόσβασης στα δεδομένα, αλλά το να γίνει η κύρια μνήμη εντελώς υψηλής ταχύτητας ήταν απαγορευτικά ακριβό. Οι ερευνητές διερεύνησαν καλύτερα σχέδια, οδηγώντας τελικά στην ιδέα των πολυεπίπεδων κρυφών μνημών. Αυτά τα πολυεπίπεδα μοντέλα κρυφής μνήμης, όπως οι κρυφές μνήμες τριών επιπέδων που βρίσκονται στα προϊόντα Core i7 της Ιντελ, επιτυγχάνουν μια ισορροπία μεταξύ κόστους και απόδοσης. Τώρα, οι CPUs μπορούν να αξιοποιήσουν μια ιεραρχία κρυφών μνημών, κάθε επίπεδο της οποίας λειτουργεί ως ρυθμιστικός χώρος μεταξύ του επεξεργαστή και της κύριας μνήμης, εξασφαλίζοντας αποτελεσματική ροή δεδομένων και ταχύτερη εκτέλεση.

Κάθε μία από αυτές τις κρυφές μνήμες έχει το δικό της μέγεθος και τη δική της εσωτερική δομή, ανεξάρτητα από τις κρυφές μνήμες άλλων επιπέδων. Υπάρχουν πολλοί τρόποι διαμόρφωσης μιας κρυφής μνήμης ως προς τη δομή της. Οι σημαντικοί που θα συζητήσουμε είναι το μέγεθος, η συσχετιστικότητα και η πολιτική αντικατάστασης. Για το μέγεθος, η μόνη εξάρτησή του είναι να είναι μεγαλύτερο από το μέγεθος της κρυφής μνήμης στα χαμηλότερα επίπεδα. Γενικά σημαίνει ότι το μέγεθος(L3 $>$ μέγεθος(L2) $>$ μέγεθος(L1). Αυτό οφείλεται στο γεγονός ότι η ταχύτητα των κρυφών μνήμης χαμηλότερων επιπέδων υποτίθεται ότι είναι υψηλότερη και επομένως απαιτείται μικρότερη κρυφή μνήμη. Η συσχέτιση αναφέρεται στην εσωτερική δομή αυτής της κρυφής μνήμης, συγκεκριμένα, ασχολείται με τις μεθόδους που χρησιμοποιούνται για να καθοριστεί πού μπορούν να αποθηκευτούν τα δεδομένα εντός της κρυφής μνήμης. Στην περίπτωσή μας, θα χρησιμοποιήσουμε set-associative caches, αλλά υπάρχουν επίσης fully-associative και direct mapping caches.

Τέλος, είναι σημαντικό να καθοριστεί η πολιτική αντικατάστασης της κρυφής μνήμης. Οι πολιτικές αντικατάστασης είναι αλγόριθμοι ή στρατηγικές που χρησιμοποιούνται για να αποφασιστεί ποιες εγγραφές της κρυφής μνήμης θα αντικατασταθούν ή θα εκδιωχθούν όταν πρέπει να φορτωθούν νέα δεδομένα σε μια πλήρη κρυφή μνήμη. Αυτές οι πολιτικές είναι ζωτικής σημασίας για τη διατήρηση της αποδοτικότητας και της απόδοσης του συστήματος κρυφής μνήμης, καθώς επηρεάζουν άμεσα τα ποσοστά επιτυχίας των προσπελάσεων της

κρυφής μνήμης. Η LRU είναι η πιο γνωστή πολιτική αντικατάστασης, η οποία αφαιρεί την καταχώρηση της κρυφής μνήμης που δεν έχει προσπελαστεί για το μεγαλύτερο χρονικό διάστημα. Οι σύγχρονοι επεξεργαστές χρησιμοποιούν πιο σύγχρονες πολιτικές αντικατάστασης που συχνά δεν είναι γνωστές στον χρήστη μιας CPU. Ως εκ τούτου, θα χρησιμοποιήσουμε μια σειρά από διαφορετικές πολιτικές αντικατάστασης σε αυτή τη διατριβή, για να δείξουμε ότι οι προβλέψεις παρουσιάζουν σταθερά υψηλή ποιότητα σε διάφορες μηχανές. Οι SHiP [5], SRRIP [6] και Mockingjay [7] είναι σύγχρονες πολιτικές αντικατάστασης που στοχεύουν στην πρόβλεψη της επαναχρησιμοποίησης μιας εγγραφής στην κρυφή μνήμη και αντικαθιστούν τη λιγότερο πιθανή να επαναχρησιμοποιηθεί.

2.2 StatCache

Έχουν γίνει στατιστικές προσπάθειες εκτίμησης των ποσοστών αστοχίας κρυφής μνήμης για πολιτικές αντικατάστασης LRU. Συγκεκριμένα, το StatCache [3] προσφέρει μια πιθανολογική προσέγγιση προς αυτόν τον στόχο χρησιμοποιώντας τις αποστάσεις επαναχρησιμοποίησης του εν λόγω προγράμματος.

Συμβολίζουμε ως προσπέλαση μνήμης τις διευθύνσεις ενός μπλοκ διευθύνσεων μεγέθους γραμμής κρυφής μνήμης.Έστω N ο αριθμός των προσπελάσεων μνήμης που συμβαίνουν κατά την εκτέλεση του προγράμματος. Μπορούμε να απαριθμήσουμε αυτές τις προσβάσεις από το 1 έως N. Για κάθε $i < j < N$ όπου i και j προσπελαύνουν τα ίδια δεδομένα και δεν έχουν συμβεί ενδιάμεσες προσπελάσεις, μπορούμε να πούμε ότι η απόσταση επαναχρησιμοποίησης αυτού είναι $i-j-1$. Αυτό σημαίνει ότι οι προσπελάσεις που συνέβησαν μεταξύ δύο διαδοχικών προσπελάσεων μιας διεύθυνσης μνήμης σε αυτά τα δεδομένα είναι $i-j-1$. Στη συνέχεια συγκεντρώνουμε αυτές τις προσβάσεις σε ένα ιστόγραμμα h. Σε αυτό το ιστόγραμμα h(i) υποδηλώνει τις προσπελάσεις που έχουν απόσταση επαναχρησιμοποίησης i.

Χρησιμοποιώντας τις αποστάσεις επαναχρησιμοποίησης, μπορούμε να καταλήξουμε στην εξής εξίσωση:

$RN \approx h(1)f(R) + h(2)f(2R) + h(3)f(3R) + ...$

όπου R είναι το ποσοστό αστοχίας, N είναι η μεγαλύτερη δυνατή απόσταση επαναχρησιμοποίησης, h είναι ένα ιστόγραμμα όπου το h(i) μας λέει ότι υπάρχουν h(i) αναφορές με απόσταση επαναχρησιμοποίησης i, και τέλος

Η συνάρτηση F(n) δηλώνει την πιθανότητα η γραμμή κρυφής μνήμης να μην παραμείνει στην κρυφή μνήμη μετά από n αστοχίες. Έτσι, υποθέτοντας ότι η κρυφή μνήμη είναι πλήρως συσχετιστική και έχει L γραμμές κρυφής μνήμης, η πιθανότητα μια γραμμή να μην παραμείνει στην κρυφή μνήμη μετά από ν αναδρομές είναι:

$f(n) = 1 - (1 - 1/L)^n$

Αυτή είναι ουσιαστικά η πιθανολογική προσέγγιση της StatCache στο ερώτημα. Κάθε απόσταση επαναχρησιμοποίησης μαζί με την πιθανότητα παραμονής της στην κρυφή μνήμη υπολογίζει έναν αριθμό αστοχιών στην κρυφή μνήμη που, αθροιζόμενες, δίνουν τις συνολικές αστοχίες της εκτέλεσης. Τα πλεονεκτήματα αυτής της μεθόδου εκτίμησης είναι ότι ο υπολογισμός της είναι απλός και γρήγορος. Το κύριο μειονέκτημα αυτής της μεθόδου είναι ότι λειτουργεί μόνο για την πολιτική αντικατάστασης LRU.

2.3 Άλλες χρήσεις Νευρωνικών Δικτύων

Τα βαθιά νευρωνικά δίκτυα έχουν αποδείξει τις ικανότητές τους να αναλύουν και να κατανοούν σύνθετα μοτίβα σε πολλά είδη προβλημάτων, όπως η ταξινόμηση εικόνων [8, 9]. Στην περίπτωσή μας, χρήσιμες είναι οι έρευνες που ασχολούνται με τον κώδικα υπολογιστών [10, 11].

Παράλληλα με αυτές, παρατηρείται επίσης μια αύξηση της δημοτικότητας των νευρωνικών δικτύων στα πλαίσια των αρχιτεκτονικών υπολογιστών.Για παράδειγμα, πολλές βελτιστοποιήσεις μεταγλωττιστών έχουν προκύψει από την εφαρμογή τεχνικών μηχανικής μάθησης. Στον προγραμματισμό εντολών, η συνάρτηση προτίμησης ενός προγραμματισμού έναντι ενός άλλου μπορεί να υπολογιστεί από τη χρονική διαφορά του αλγορίθμου RL [12]. Το μοντέλο που βασίζεται σε LSTM [13], παρακάμπτει τη χειροκίνητη μηχανική χαρακτηριστικών, αποκτώντας αυτόνομα ευρετικές λειτουργίες του μεταγλωττιστή από τον ακατέργαστο κώδικα.Αυτό επιτρέπει την κατασκευή κατάλληλων ενσωματώσεων για προγράμματα, ενώ ταυτόχρονα κατέχει τη διαδικασία βελτιστοποίησης.

Πολλά διαφορετικά δίκτυα μηχανικής μάθησης έχουν χρησιμοποιηθεί για την πρόβλεψη παρόμοιων προβλημάτων με το δικό μας [14].Dong *et al.*[15] χρησιμοποιούν τεχνητά νευρωνικά δίκτυα για την πρόβλεψη χαρακτηριστικών υψηλότερου επιπέδου (όπως αστοχίες σε cache read/write και instructions per cycle) από-χαμηλότερου επιπέδου χαρακτηριστικά (όπως cache associativity, capacity, latency) για ιεραρχίες κρυφών μνημών βασισμένες σε μη πτητική μνήμη. Άλλα δίκτυα μηχανικής μάθησης έχουν εισαχθεί για να βοηθήσουν στην πρόβλεψη αποτελεσματικής κατανομής πόρων [16] και χρονοπρογραμματισμού εργασιών [17], ώστε να επιλέγεται πάντα η διαδρομή των μέγιστων οδηγιών ανά κύκλο.

Πολυάριθμες εισαγωγές της μηχανικής μάθησης στη σφαίρα των αρχιτεκτονικών υπολογιστών, τόσο παρόμοιες όσο και διαφορετικές, έχουν προκαλέσει την περιέργειά μας. Αυτό το κίνητρο μας οδήγησε να διερευνήσουμε την επίλυση του προβλήματος πρόβλεψης του ποσοστού αστοχίας στην κρυφή μνήμη μέσα από το πρίσμα της μηχανικής μάθησης.

2.4 Ενσωματώσεις

Οι ενσωματώσεις είναι αναπαράσταση των αντικειμένων ως διανύσματα. Μια συνάρτηση μετατρέπει ένα αντικείμενο που είναι αναγνωρίσιμο από τον άνθρωπο σε ένα διάνυσμα αριθμών, το οποίο μπορεί να αναγνωριστεί από ένα πρόγραμμα με τη μικρότερη δυνατή απώλεια δεδομένων. Αυτή η μέθοδος αναπαράστασης χρησιμοποιείται συχνά στο NLP, όπου κάθε λέξη του λεξικού μετατρέπεται σε αριθμό και τροφοδοτείται σε ένα νευρωνικό δίκτυο.

Μια παρόμοια τεχνική έχει αναπτυχθεί για τη δημιουργία ενσωματώσεων κώδικα [18, 19] όπου τα σύμβολα του κώδικα χρησιμοποιούνται για τη μετατροπή μιας ακολουθίας εντολών κώδικα στο διάνυσμα ενσωμάτωσης. Κάθε εντολή παίρνει μια τιμή από το 0 έως το N όπου N είναι ο αυθαίρετος αριθμός συμβόλων που θέλουμε να λάβουμε υπόψη (τα υπόλοιπα τα απορρίπτουμε) και αυτό είναι γνωστό ως λεξιλόγιο. Στη συνέχεια, αυτή η μετατροπή των συμβόλων θα χρησιμοποιηθεί από κάποιο μηχανισμό για να μετατρέψει αυτούς τους πίνακες συμβόλων σε ένα διάνυσμα ποθ αναπαρηστά το πρόγραμμα με τη μικρότερη δυνατή απώλεια δεδομένων.

Στην περίπτωσή μας θα χρησιμοποιήσουμε το IR2Vec [18] για τη δημιουργία των ενσωματώσεων μας. Το IR2Vec δέχεται ως είσοδο την ενδιάμεση αναπαράσταση του κώδικα καθώς και ένα λεξιλόγιο, μετατρέπει το αρχείο IR σε έναν πίνακα αριθμών και στη συνέχεια τροφοδοτεί ένα δίκτυο, το οποίο έχει σχεδιαστεί για να καταγράφει τη σύνταξη και τη σημασιολογία. Έξοδος αυτής της διαδικασίας είναι ένα διάνυσμα μεγέθους 300, το οποίο υποτίθεται ότι θα δώσει στο μοντέλο μας ζωτικές πληροφορίες σχετικά με τις διαδικασίες του προγράμματος και τον τρόπο κατανόησης των αποστάσεων επαναχρησιμοποίησης.

Μέσω μάθησης χωρίς επίβλεψη οι συγγραφείς του IR2Vec δημιούργησαν ένα δίκτυο που αναγνωρίζει και αναπαριστά με ακρίβεια αυτά τα αρχεία IR. Το δίκτυό τους εκπαιδεύτηκε και δοκιμάστηκε στα βενσημαρκς SPEC CPU 17 και στη βιβλιοθήκη Boost. Τα πειραματικά τους αποτελέσματα εξασφαλίζουν την πρακτική βιωσιμότητα αυτού του δικτύου στο πρόβλημά μας, καθώς τα δικά μας πειραματικά δεδομένα προέρχονται επίσης από τις σουίτες συγκριτικών δοκιμών SPEC, όπως θα συζητήσουμε στη συνέχεια. Έτσι, θα χρησιμοποιήσουμε αυτό το δίκτυο και το λεξιλόγιο ως μαύρο κουτί για να μετατρέψουμε τον πηγαίο κώδικα του συνόλου δεδομένων μας. Η έξοδος αυτής της διαδικασίας είναι ένα διάνυσμα μεγέθους 300, το οποίο υποτίθεται ότι θα δώσει στο μοντέλο μας ζωτικές πληροφορίες σχετικά με τις διαδικασίες του προγράμματος και τον τρόπο κατανόησης των αποστάσεων επαναχρησιμοποίησης.

Μέρος II

Πρακτικό Μέρος

Κεφάλαιο 3

Προσέγγιση και μέθοδοι

3.1 Επισκόπηση

Ο στόχος μας σε αυτή τη διατριβή είναι να προβλέψουμε τα ποσοστά αστοχίας της κρυφής μνήμης δεδομένης ενός προγράμματος και μιας αρχιτεκτονικής κρυφής μνήμης. Για να χρησιμοποιήσουμε ένα πρόγραμμα ως είσοδο, πρέπει να μετατρέψουμε τα χαρακτηριστικά του σε χρήσιμα δεδομένα για να προβλέψει το δίκτυό μας. Για το σκοπό αυτό, σύμφωνα με το StatCache 2.2, χρειαζόμαστε τις αποστάσεις επαναχρησιμοποίησης κάθε προγράμματος στο σύνολο αναφοράς μας. Οι αποστάσεις επαναχρησιμοποίησης, όπως αναφέρθηκε προηγουμένως, είναι οι αποστάσεις μεταξύ διαδοχικών προσπελάσεων στην ίδια διεύθυνση στη μνήμη. Αυτές οι αποστάσεις εξαρτώνται κυρίως από το πρόγραμμα και τις εισόδους τουδιαφέρουν πολύ λίγο μεταξύ διαφορετικών αρχιτεκτονικών και επομένως θα καταγραφούν μόνο μία φορά από κάθε ίχνος ενός προγράμματος. Θα αποθηκευτούν σε ένα ιστόγραμμα και ένας κλάδος του δικτύου θα προσπαθήσει να προβλέψει τα ποσοστά αστοχίας χρησιμοποιώντας τα.

Στη συνέχεια, πρέπει να μετατρέψουμε τον κώδικα εισόδου σε διάνυσμα (ενσωμάτωση), ώστε το δίκτυο να μπορεί να ανακτήσει ακόμη περισσότερες πληροφορίες από τον κώδικα εισόδου σχετικά με τη δομή του προγράμματος, τη ροή των δεδομένων κ.λπ. Για να προβλέψουμε με ακρίβεια τα ποσοστά αστοχίας στην κρυφή μνήμη, πρέπει να ερμηνεύσουμε κατάλληλα τα μετασχηματισμένα δεδομένα. Το διάνυσμα ενσωμάτωσης μπορεί να τροφοδοτηθεί σε ένα πυκνό νευρωνικό δίκτυο, οπότε αυτό κάνουμε για τον κλάδο του μετασχηματιστή. Από την άλλη πλευρά, τα ιστογράμματα της απόστασης επαναχρησιμοποίησης είναι πιο πολύπλοκα στην κατανόηση- ένα απλό πυκνό νευρωνικό δίκτυο δεν είναι το πιο κατάλληλο δίκτυο για την πλήρη κατανόηση της πολυπλοκότητας αυτού του προβλήματος.

Για τον σκοπό αυτό, εισάγουμε τρία διαφορετικά μοντέλα που κατανοούν με διαφορετικό τρόπο τις αποστάσεις επαναχρησιμοποίησης. Το πρώτο είναι ένα ρηχό πολυεπίπεδο περσεπτρον που υπολογίζει την αναλογία αστοχίας αποκλειστικά με βάση τις αποστάσεις επαναχρησιμοποίησης. Στη συνέχεια, εισάγουμε ένα διπλό στρώμα LSTM και ένα δίκτυο CNN που έχει ως σκοπό την καλύτερη κατανόηση των αποστάσεων επαναχρησιμοποίησης δεδομένων των γειτονικών αποστάσεων επαναχρησιμοποίησης. Αυτά τα δύο δίκτυα υπολογίζουν καλύτερα τον λόγο αστοχίας στην κρυφή μνήμη, χρησιμοποιώντας τις αποστάσεις επαναχρησιμοποίησης καθώς και τον κλάδο μετασχηματισμού.

Τέλος, εισάγουμε ένα βαθύ MLP που λαμβάνει ως είσοδο τις αποστάσεις επαναχρησι-

μοποίησης και τις ενσωματώσεις, εισάγει το μέγεθος LLC ως παράμετρο και υπολογίζει τα ποσοστά μισς για άλλες αρχιτεκτονικές κρυφής μνήμης με την ίδια πολιτική αντικατάστασης.

3.2 Πειραματική οργάνωση

3.2.1 Προσομοίωση

Τα αρχικά δεδομένα εισόδου για το σύστημά μας αποτελούνται από τις σουίτες συγκριτικών δοκιμών SPEC 2007 και SPEC 2016 [20, 21].Τα SPEC είναι σύνολα δοκιμών αναφοράς που έχουν σχεδιαστεί για να παρέχουν ένα συγκριτικό μέτρο των επιδόσεων έντασης υπολογισμού σε ένα ευρύτατο πρακτικό φάσμα υλικού, χρησιμοποιώντας φόρτους εργασίας που έχουν αναπτυχθεί από εφαρμογές πραγματικών χρηστών. Αυτές οι σουίτες είναι ιδανικές για την παρουσίαση του στόχου μας, καθώς παρέχουν μια μεγάλη ποικιλία προβλημάτων και αλγορίθμων που καλύπτουν ένα ευρύ φάσμα υπολογιστικών εργασιών.

Για την προσομοίωση αυτών των προγραμμάτων, χρησιμοποιούμε το ChampSim, έναν προσομοιωτή επεξεργαστή που στοχεύει κυρίως στην όσο το δυνατόν ακριβέστερη προσομοίωση του υποσυστήματος μνήμης και της πρόβλεψης διακλαδώσεων [22]. Θα χρησιμοποιήσουμε αυτό το εργαλείο προσομοίωσης για τη συλλογή των δεδομένων χρόνου εκτέλεσης (αστοχίες και αποστάσεις επαναχρησιμοποίησης της κρυφής μνήμης, όπως εξηγείται στο επόμενο κεφάλαιο 3.2.2). Εκτελεί ένα πρόγραμμα σε μια προσομοιωμένη μηχανή που βασίζεται σε μια δεδομένη αρχιτεκτονική. Ως εκ τούτου, για να εκτελέσουμε το ChampSim, χρειαζόμαστε ίχνη από τα αρχεία αναφοράς του SPEC. Για το σκοπό αυτό, χρησιμοποιούμε τα ίχνη που δίνονται από το 3rd Data Prefetching Championship. Αυτά τα ίχνη αποτελούνται από 2 δισεκατομμύρια εντολές το καθένα και καλύπτουν μεγάλα τμήματα της εκτέλεσης ενός αρχείο αναφοράς. Κάθε αρχείο αναφοράς έχει από ένα έως έξι ίχνη στο όνομά του, καθένα από τα οποία είναι αντιπροσωπευτικό ενός συγκεκριμένου ποσοστού της εκτέλεσής του. Αργότερα, θα χρησιμοποιήσουμε αυτό το ποσοστό ως βάρος για να υπολογίσουμε τις αστοχίες κρυφής μνήμης ενός αρχείου αναφοράς ως σταθμισμένο μέσο όρο των ποσοστών αστοχίας των ιχνών του.

Η κρυφή μνήμη που παράγει τη μεγαλύτερη καθυστέρηση αστοχίας είναι η LLC. Αυτό καθιστά την πρόβλεψη για αυτήν πολύ πιο πολύτιμη από ό,τι για τις κρυφές μνήμες άλλου επιπέδου. Επομένως, στην προσέγγισή μας, διατηρούμε αμετάβλητες τις κρυφές μνήμες χαμηλότερου επιπέδου L1D και L2C με μεγέθη 48KB και 512KB, συσχετιστικότητα 12 και 16 αντίστοιχα και πολιτικές αντικατάστασης LRU. Πραγματοποιήσαμε τις προσομοιώσεις που αναφέρονται στα επόμενα κεφάλαια με ένα ευρύ φάσμα μεγεθών κρυφής μνήμης και τέσσερις πολιτικές αντικατάστασης για την LLC, ώστε να καλύψουμε ένα μεγάλο σύνολο αρχιτεκτονικών του πραγματικού κόσμου και να δείξουμε ότι είναι ευρέως χρησιμοποιήσιμη. Χρησιμοποιήσαμε τα μεγέθη κρυφής μνήμης που αναφέρονται στον πίνακα 3.1. Τα μεγέθη LLC των 768KB, 1536KB, 3072KB και 6144KB έχουν συσχετιστικότητα 12, ενώ οι κρυφές μνήμες μεγέθους 1024, 2048, 4096 και 8192KB έχουν συσχετιστικότητα 16.

Replacement policy	LLC μέγεθος [KB]
LRU	768, 1024, 1536, 2048, 3072, 4096, 6144, 8192
SHiP	1024, 2048, 4096, 8192
SRRIP	1024, 2048, 4096, 8192
Mockingjay	1024, 2048, 4096, 8192

Πίνακας 3.1: *Διαμορφώσεις κρυφής μνήμης LLC*

3.2.2 Σύνολο δεδομένων

Συλλογή αποστάσεων επαναχρησιμοποίησης

Η συλλογή των αποστάσεων επαναχρησιμοποίησης κατά τη διάρκεια του χρόνου εκτέλεσης είναι αρκετά απλή. Υλοποιήσαμε έναν αναλυτή αποστάσεων επαναχρησιμοποίησης που θα χρησιμοποιηθεί στο ChampSim. Του δίνεται μια διεύθυνση και μια κατάσταση και υπολογίζει την απόσταση επαναχρησιμοποίησης. Αυτό έγινε μέσω μιας απλής συνάρτησης που, κάθε φορά που καλείται, ελέγχει την τελευταία φορά που έγινε πρόσβαση σε αυτά τα δεδομένα και την αυξάνει στο ιστόγραμμα. Στη συνέχεια, η διεύθυνση των δεδομένων αποθηκεύεται για μελλοντικούς υπολογισμούς της απόστασης επαναχρησιμοποίησης και συνεχίζουμε. Αυτή η υλοποίηση είναι ελαφριά, αφού δεν προστίθενται παρά μόνο μερικές εντολές κάθε φορά που καλείται, και αποτελεσματική, αφού χρησιμοποιούνται αποδοτικές δομές δεδομένων, όπως οι χάρτες κατακερματισμού.

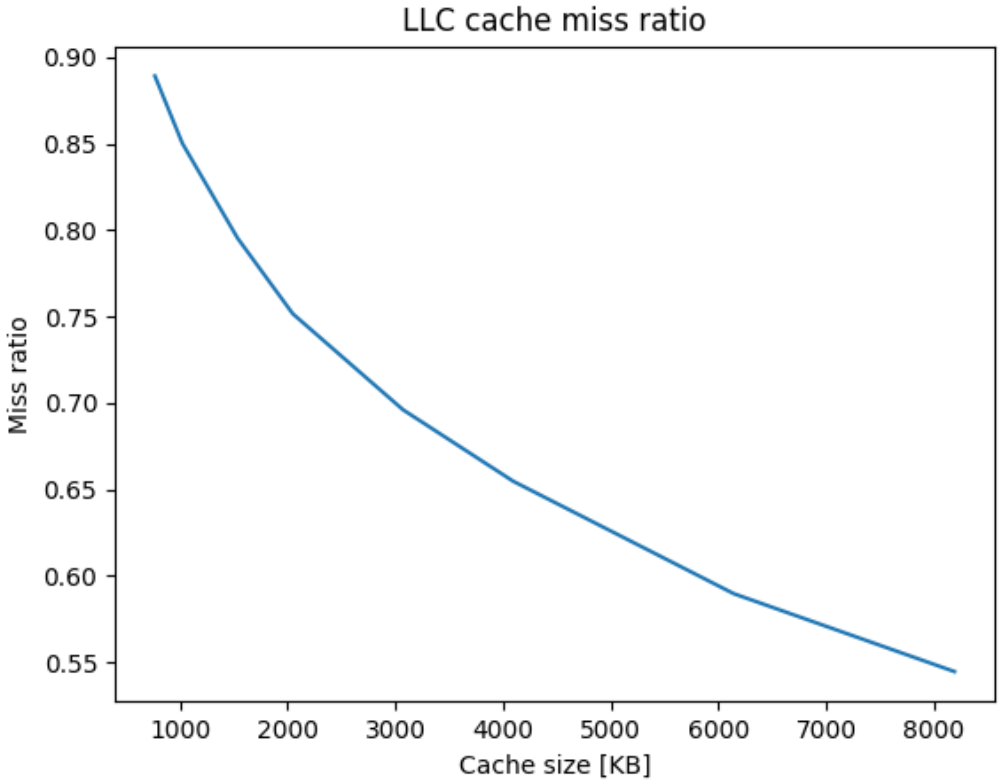
Για να χρησιμοποιήσουμε αυτόν τον αναλυτή της αποστάσεων επαναχρησιμοποίησης, τον προσθέσαμε στην πολιτική αντικατάστασης του L1D που εκτελεί αυτή την προαναφερθείσα διαδικασία κάθε ν-οστή φορά που καλείται. Αυτό το ν πρέπει να είναι αρκετά μικρό για να έχουμε μια αντιπροσωπευτική εικόνα των αποστάσεων επαναχρησιμοποίησης στο σύνολο δεδομένων μας, αλλά όχι πολύ μικρό, καθώς δεν θέλουμε να εκτελούμε τον αναλυτή αποστάσεων επαναχρησιμοποίησης με κάθε πρόσβαση σε δεδομένα, επειδή αυτό αυξάνει τον χρόνο επεξεργασίας. Στην περίπτωσή μας, χρησιμοποιούμε $\nu = 16$ προσπελάσεις. Αυτό είναι υπερβολικά σχολαστικό για 512κ προσπελάσεις, αλλά θέλουμε να διασφαλίσουμε ότι οι έξοδοι δεν επηρεάζονται από αυτό και έτσι θυσιάσαμε λίγο χρόνο επεξεργασίας.

Η διακύμανση των αστοχιών της κρυφής μνήμης καθ' όλη τη διάρκεια της εκτέλεσης του προγράμματος μπορεί να είναι ένα ζήτημα που επηρεάζει αρνητικά την εκτίμησή μας. Καθώς το πρόγραμμα περνάει από τις διάφορες φάσεις εκτέλεσης, ο λόγος των αστοχιών στην κρυφή μνήμη αλλάζει με την πάροδο του χρόνου. Σύμφωνα με το έγγραφο της StatCache [3], ο διαχωρισμός της εξόδου των αναλυτών σε παράθυρα με μικρότερο αριθμό προσπελάσεων είναι η λύση σε αυτό το ζήτημα. Η σκέψη πίσω από αυτό είναι ότι ο αριθμός των προσπελάσεων θα είναι αρκετά μικρός, έτσι ώστε ο λόγος αστοχίας κρυφής μνήμης κάθε παραθύρου να είναι πιθανότατα σταθερός με την πάροδο του χρόνου.

Για να το πετύχουμε αυτό, προσομοιώσαμε τα ίχνη και για κάθε 512κ προσπελάσεις, επιστρέψαμε την κατάσταση του ιστογράμματος της απόστασης επαναχρησιμοποίησης. Παρατηρήσαμε ότι αυτό το μέγεθος λειτουργεί καλύτερα στην περίπτωσή μας, καθώς δίνει στην προσομοίωση χρόνο για να γεμίσει ένα μέρος των κρυφών μνήμες και παρόλα αυτά να δειγματοληπτεί η εκτέλεση του προγράμματος αρκετά συχνά ώστε να μην παρατηρείται η διακύμανση. Τώρα, για κάθε ίχνος αναφοράς, έχουμε έναν αριθμό παραθύρων με 512κ

προσπελάσεις το καθένα. κάθε παράθυρο αποτελείται από τις αποστάσεις επαναχρησιμοποίησης που εμφανίστηκαν μέσα σε αυτές τις 512κ προσπελάσεις και τον λόγο αστοχίας LLC στον οποίο αντιστοιχούν αυτές. Ας είναι γνωστό ότι ορισμένα ίχνη έχουν πολύ λιγότερες από 512κ προσπελάσεις εντός της εκτέλεσής τους. Χρησιμοποιήσαμε ένα παράθυρο για κάθε ένα από αυτά τα ίχνη ανεξάρτητα, έτσι ώστε να μην χαθούν τα δεδομένα τους, αλλά κάποια από αυτά θα οδηγήσουν σε κάποια προβλήματα αργότερα λόγω της διαφορετικής κλίμακας τους.

Για τον υπολογισμό του συνολικού λόγου αστοχίας μιας συγκριτικής αξιολόγησης, πρέπει να υπολογιστεί ο μέσος όρος του λόγου αστοχίας κάθε παραθύρου σε ένα ίχνος. Αυτός ο λόγος αστοχιών υπολογίζεται από το *αστοχίες κρυφής μνήμης/προσπελάσεις κρυφής μνήμης* στην LLC κατά τη διάρκεια της περιόδου 512κ προσπελάσεων κάθε παραθύρου. Στη συνέχεια, ο λόγος αστοχιών κάθε ίχνους πολλαπλασιάζεται με ένα βάρος, το οποίο αντιπροσωπεύει το ποσοστό της συνολικής εκτέλεσης του αρχείο αναφοράς που προσομοιώνει αυτό το ίχνος.Το άθροισμα αυτών των αναλογιών αστοχίας πολλαπλασιαζόμενο με το βάρος του ίχνους κατασκευάζει έναν σταθμισμένο μέσο όρο, ο οποίος θα είναι η αναλογία αστοχίας ολόκληρου του αρχείου αναφοράς.



Σχήμα 3.1: *Μέσος όρος λόγων αστοχίας όλων των αρχείων αναφοράς με πολιτική αντικατάστασης LRU*

Όπως βλέπουμε, ο μέσος λόγος αστοχιών στην κρυφή μνήμη όλων των συγκριτικών δοκιμών βρίσκεται στο 90% για τη μικρότερη κρυφή μνήμη και συνεχίζει να μειώνεται σταδιακά, μέχρι το 55%.Αυτό είναι άμεση συνέπεια του γεγονότος ότι, κατά μέσο όρο, οι μεγαλύτε-

ρες κρυφές μνήμες οδηγούν σε χαμηλότερα ποσοστά αστοχίας, δεδομένου ότι μπορούν να αποθηκεύσουν περισσότερες πληροφορίες.

Για να συνοψίσουμε και να γίνουμε πιο ακριβείς, το σύνολο δεδομένων μας περιέχει από 1 έως 373 παράθυρα για κάθε ένα από τα 186 ίχνη που προκύπτουν από 47 συγκριτικά μέτρα.Κάθε παράθυρο περιέχει το ιστόγραμμα των αποστάσεων επαναχρησιμοποίησης και τα ποσοστά αστοχίας της κρυφής μνήμης που καταγράφηκαν κατά τη διάρκεια 512κ συνεχών προσπελάσεων μνήμης κατά τη διάρκεια της εκτέλεσης του αρχείου αναφοράς. Οι λόγοι αστοχίας κρυφής μνήμης είναι αριθμοί στο εύρος [0, 1] με βάση τις προσομοιωμένες τιμές του ChampSim για κάθε μία από τις διαμορφώσεις LLC που αναφέρονται στον πίνακα 3.1. Αυτές οι αποστάσεις επαναχρησιμοποίησης θα αποτελέσουν την είσοδο για τον πρώτο κλάδο του δικτύου μας και οι αναλογίες αστοχίας είναι η πραγματική έξοδος στην οποία θα προσαρμοστεί η έξοδος του δικτύου κατά την εκπαίδευση και θα συγκριθεί με αυτήν κατά τη δοκιμή.

Ενσωματώσεις

Όπως συζητήθηκε στην ενότητα 3.2, έχουμε μεταγλωττίσει τον κώδικα σε αρχεία IP. Θα χρησιμοποιήσουμε το προ-εκπαιδευμένο μοντέλο του IR2Vec για τη μετατροπή του κώδικα σε διανύσματα. Με αυτό, δημιουργούμε 47 διανύσματα 300 διαστάσεων, ένα που αντιπροσωπεύει κάθε σημείο αναφοράς. Αυτός είναι ο κλάδος μετασχηματισμού των δικτύων μας. Αυτό βοηθά το υπόλοιπο δίκτυο, μέσω του κώδικα του προγράμματος, να κατανοήσει τις ιδιότητες του προγράμματος, γεγονός που θα βοηθήσει στον υπολογισμό ενός πιο ακριβούς λόγου αστοχίας.

Το IR2Vec παράγει είτε μία ενσωμάτωση για κάθε συνάρτηση του προγράμματος είτε μία ενσωμάτωση για ολόκληρο το πρόγραμμα ταυτόχρονα. Το πλεονέκτημα της ύπαρξης μιας ενσωμάτωσης για κάθε συνάρτηση είναι η ακρίβεια ως προς το ποιο μέρος της εκτέλεσης θέλουμε να εστιάσουμε. Η ενσωμάτωση για όλο το πρόγραμμα θα μετατρέψει ολόκληρο το αρχείο αναφοράς σε ένα διάνυσμα 300 διαστάσεων. Στην περίπτωσή μας, δεν γνωρίζουμε τη συνάρτηση ή το μέρος του προγράμματος που εκτελείται σε κάθε ίχνος- έτσι, χρησιμοποιούμε ένα διάνυσμα που είναι η ενσωμάτωση ολόκληρου του αρχείου αναφοράς και ένα μόνο με το διάνυσμα της κύριας συνάρτησης για κάθε αρχείο αναφοράς.

Train-test split

Μια άλλη μέριμνα είναι ο διαχωρισμός του συνόλου δεδομένων, το οποίο προκύπτει από τα τμήματα 3.2.2 και 3.2.2, σε ένα σύνολο εκπαίδευσης και ένα σύνολο δοκιμής, το οποίο δεν είναι καθόλου τετριμμένο. το σύνολο δεδομένων μας αποτελείται από πλειάδες. Όπως περιγράφεται στις ενότητες 3.2.2 και 3.2.2, κάθε πλειάδα περιέχει το ιστόγραμμα της απόστασης επαναχρησιμοποίησης ενός παραθύρου, τα ποσοστά αστοχίας, καθώς και τις ενσωματώσεις του αρχείου αναφοράς από το οποίο προέρχεται.

Οι αποστάσεις επαναχρησιμοποίησης έχουν επιλεγεί από ένα αρχείο ιχνών κάθε φορά, οπότε όλα τα στοιχεία που προκύπτουν από το ίδιο αρχείο ιχνών θα έχουν παρόμοια δομή απόστασης επαναχρησιμοποίησης. Το ίδιο ισχύει και για τις ενσωματώσεις- τις συλλέξαμε από τον κώδικα που έχει μεταγλωττιστεί από τον συγκριτικό δείκτη και όλα τα αρχεία

ιχνών που έχουν παραχθεί από τον ίδιο συγκριτικό δείκτη με διαφορετικές διαμορφώσεις θα έχουν ως είσοδο το ίδιο αρχείο ενσωμάτωσης. Για παράδειγμα, τα ίχνη 435.gromacs-111B, 435.gromacs-134B, 435.gromacs-226B & 435.gromacs-228B έχουν όλα παραχθεί από το αρχείο αναφοράς 435.gromacs από το SPEC και συνεπώς έχουν την ίδια ενσωμάτωση σε κάθε στοιχείο του συνόλου δεδομένων. Λόγω αυτής της ομοιότητας, δεν μπορούμε να έχουμε στοιχεία ενός αρχείου αναφοράς τόσο στο σύνολο εκπαίδευσης όσο και στο σύνολο δοκιμής- αυτό θα διαστρέβλωνε τα δεδομένα μας, αφού το δίκτυο θα έχει εκπαιδευτεί σε αυτή τη δομή.

Η αρχική μας σκέψη για το διαχωρισμό εκπαίδευσης-δοκιμής ήταν να διαχωρίσουμε το 20%-30% των αρχείων αναφοράς του συνόλου δεδομένων για το σύνολο δοκιμών. Όμως, δεδομένου ότι το σύνολο δεδομένων μας είναι αρκετά μικρό, αποτελούμενο μόνο από 47 αρχεία αναφοράς, ο διαχωρισμός του 20%-30% των σημείων αναφοράς για ένα σύνολο δοκιμής δεν θα ήταν βέλτιστος. Θα είχε ως αποτέλεσμα ένα σύνολο δοκιμών που θα ήταν πολύ μικρό και, πιθανότατα, δεν θα ήταν ενδεικτικό όλων των πιθανών προγραμμάτων που θα μπορούσε να αντιμετωπίσει το δίκτυο. Για το λόγο αυτό, αποφασίσαμε να χρησιμοποιήσουμε μια μέθοδο που ονομάζεται leave-one-out cross-validation [23]. Ξεχωρίζουμε ένα αρχείο αναφοράς κάθε φορά και προσπαθούμε να προβλέψουμε τα ποσοστά αστοχίας του με το δίκτυο που εκπαιδεύτηκε στα άλλα 46 σημεία αναφοράς. Με αυτόν τον τρόπο, διατρέχουμε τα 47 σημεία αναφοράς και λαμβάνουμε 47 αποτελέσματα, ένα για κάθε αρχείο αναφοράς του συνόλου δεδομένων.

3.2.3 Μετρικές

Για την εκπαίδευση και τη δοκιμή των δικτύων, χρησιμοποιήσαμε τις βιβλιοθήκες TensorFlow στην Python. Οι χρόνοι επεξεργασίας των δικτύων προέρχονται από τις διαδικασίες εκπαίδευσης και δοκιμής που πραγματοποιήσαμε με τη χρήση μιας NVIDIA T4 GPU.

Χρησιμοποιήσαμε μια μετρική μέσου τετραγωνικού σφάλματος για τη διαδικασία εκπαίδευσης. Αυτό λειτουργεί επειδή, ανεξάρτητα από τη βαρύτητα του παραθύρου στο τελικό αποτέλεσμα, θέλουμε τη μικρότερη δυνατή απόκλιση από την εκτιμώμενη τιμή του. Λειτουργεί καλύτερα από έναν σταθμισμένο μέσο όρο (όπου το βάρος ενός παραθύρου θα είναι $trace\_weight * 1/windows\_per\_trace$) ή μια συνάρτηση μέσου απόλυτου σφάλματος, επειδή αναγκάζει το δίκτυο να μειώσει τις μεγαλύτερες αποκλίσεις. Αυτό, με τη σειρά του, ωθεί τη συνολική μέση απόκλιση να είναι μικρότερη.

Η εκπαίδευση έγινε με έναν βελτιστοποιητή Adam [24], μια στοχαστική μέθοδο για τη ρύθμιση των βαρών των δικτύων για βελτιστοποίηση. Ο ρυθμός μάθησης ήταν 0,001, το μέγεθος της παρτίδας ήταν 16 και εκπαιδεύσαμε το δίκτυο για 50 εποχές. Για αυτές τις μεταβλητές, δοκιμάσαμε πολλές διαφορετικές διαμορφώσεις και κρατήσαμε αυτήν εδώ, καθώς παρείχε τα πιο ακριβή αποτελέσματα.

Για να αξιολογήσουμε πόσο καλή είναι η πρόβλεψη ενός δικτύου, θα αξιολογήσουμε το απόλυτο σφάλμα του από την πραγματική αναλογία αστοχίας που υπολογίσαμε στο 3.2.2. Για να το υπολογίσουμε αυτό, πρέπει να υπολογίσουμε το μέσο όρο της εξόδου του δικτύου για κάθε παράθυρο σε ένα ίχνος. Στη συνέχεια, πρέπει να υπολογιστεί ένας σταθμισμένος μέσος όρος σε ένα σημείο αναφοράς με το βάρος του ίχνους και να αφαιρεθεί από την

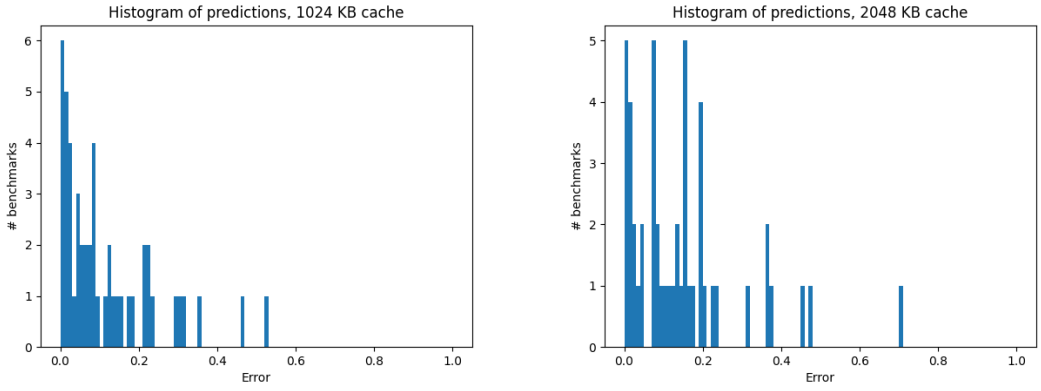
πραγματική τιμή. Η απόλυτη τιμή αυτής της αφαίρεσης θα αναφέρεται ως το απόλυτο σφάλμα πρόβλεψης. Για κάθε ένα από τα 47 σημεία αναφοράς στο σύνολο δεδομένων μας, θα υπολογιστεί ένα σφάλμα με το σύνολο εκπαίδευσης για τα άλλα 46 σημεία αναφοράς, όπως εξηγείται στο 3.2. Για να συγκρίνουμε αυτά τα σύνολα σφαλμάτων μεταξύ τους και με τις προαναφερθείσες προβλέψεις της StatCache, θα χρησιμοποιήσουμε έναν γεωμετρικό μέσο όρο. Πρέπει να αναφερθεί ότι με αυτόν τον τρόπο υπολογισμού του συνολικού αποτελέσματος μιας συγκριτικής αξιολόγησης χάνεται κάποια διακύμανση των δεδομένων στην ανάλυση. Οι διαφορές μεταξύ της πρόβλεψης και της πραγματικής τιμής ορισμένων ιχνών ποικίλλουν σημαντικά σε ορισμένα σημεία αναφοράς, έως και 70%. Επίσης, η διαφορά των παραθύρων μεταξύ πρόβλεψης και πραγματικής τιμής μπορεί να ποικίλλει εντός ενός ίχνους. Αυτές οι διαφορές εξισορροπούνται σε πολλαπλά παράθυρα και ίχνη.

3.3 Υπολογισμός StatCache

Οι έξοδοι της StatCache πρέπει να υπολογιστούν. Είναι σημαντικό να έχουμε μια ανταγωνιστική, σύγχρονη προσέγγιση πρόβλεψης που βασίζεται επίσης σε ιστογράμματα αποστάσεων επαναχρησιμοποίησης για να συγκρίνουμε τα αποτελέσματά μας. Η σύγκριση των δύο θα μας επιτρέψει να ποσοτικοποιήσουμε την επίδραση της χρήσης νευρωνικών δικτύων σε αντίθεση με μια αναλυτική προσέγγιση, δεδομένων των ίδιων δεδομένων εισόδου.

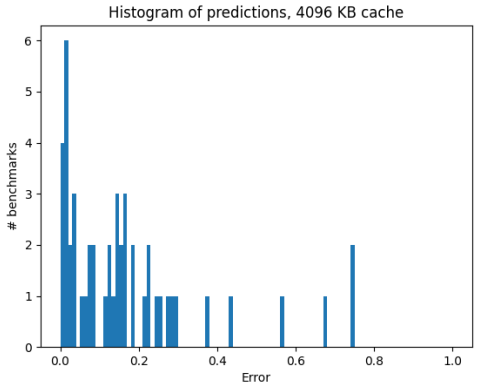
Το StatCache είναι μια πιθανολογική προσέγγιση του προβλήματος για την πολιτική αντικατάστασης LRU και οι τιμές του λαμβάνονται από τον τύπο που αναφέρεται στο κεφάλαιο 2.2. Κατά τον υπολογισμό αυτού του τύπου, λαμβάνουμε έναν λόγο αστοχίας για κάθε ένα από τα παράθυρα. Αφού τα υπολογίσουμε κατά μέσο όρο για κάθε ίχνος και υπολογίσουμε τον σταθμισμένο μέσο όρο για κάθε σημείο αναφοράς, όπως ακριβώς και προηγουμένως, υπολογίζουμε τον προβλεπόμενο λόγο αστοχίας για κάθε σημείο αναφοράς. Οι απόλυτες αποκλίσεις σφάλματος των προβλέψεων από τις πραγματικές τιμές, όπως απεικονίζονται στο σχήμα 3.2 για τα 47 αρχεία αναφοράς, φαίνεται να ακολουθούν μια αναδιπλωμένη κανονική κατανομή. Αυτό είναι αναμενόμενο αφού η προσέγγιση προσπαθεί να προβλέψει την τιμή του πραγματικού λόγου αστοχίας. Ως εκ τούτου, το απόλυτο σφάλμα αυτής της πρόβλεψης θα πρέπει να είναι το απόλυτο μιας κανονικής κατανομής γύρω από το μηδέν. Θα χρησιμοποιήσουμε τον γεωμετρικό μέσο όρο για να συνοψίσουμε τις κατανομές ως έναν αριθμό και να συγκρίνουμε τις εξόδους μεταξύ τους. Ο γεωμετρικός μέσος όρος των απόλυτων σφαλμάτων πρόβλεψης από το μοντέλο StatCache είναι 5,4%, 8,3%, 8,7% και 8,5% για τις κρυφές μνήμες μεγέθους 1MB, 2MB, 4MB και 8MB, αντίστοιχα.

Αυτή η αναδιπλωμένη κανονική κατανομή οδηγεί σε θηκογράμματα όπως αυτά που φαίνονται στο σχήμα 3.3. Η πορτοκαλί γραμμή αντιπροσωπεύει τη διάμεσο. Το ενδοτεταρτημοριακό εύρος (IQR) αντιπροσωπεύει το 25% των τιμών σε κάθε πλευρά της διαμέσου. Τα μουστάκια έχουν μέγιστο μήκος 1,5 * IQR και όλες οι τιμές εκτός αυτού του εύρους θεωρούνται ακραίες. Όλα τα θηκογράμματα των απόλυτων σφαλμάτων πρόβλεψης που θα συζητήσουμε σε αυτή τη διατριβή θα έχουν παρόμοια δομή για αυτό το πρόβλημα. Ο μέσος όρος είναι πολύ χαμηλός και έτσι θα έχουμε μικρά κουτιά και κάποιες ακραίες τιμές στο θηκογράμματα.

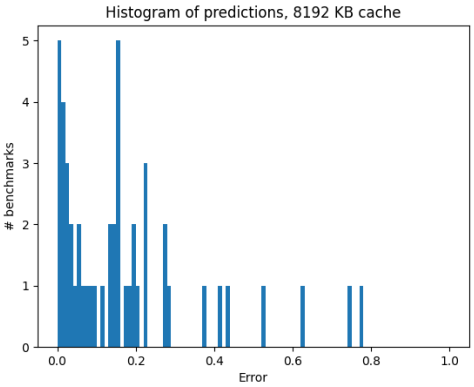


(α′) 1024KB

(β′) 2048KB



(γ′) 4096KB



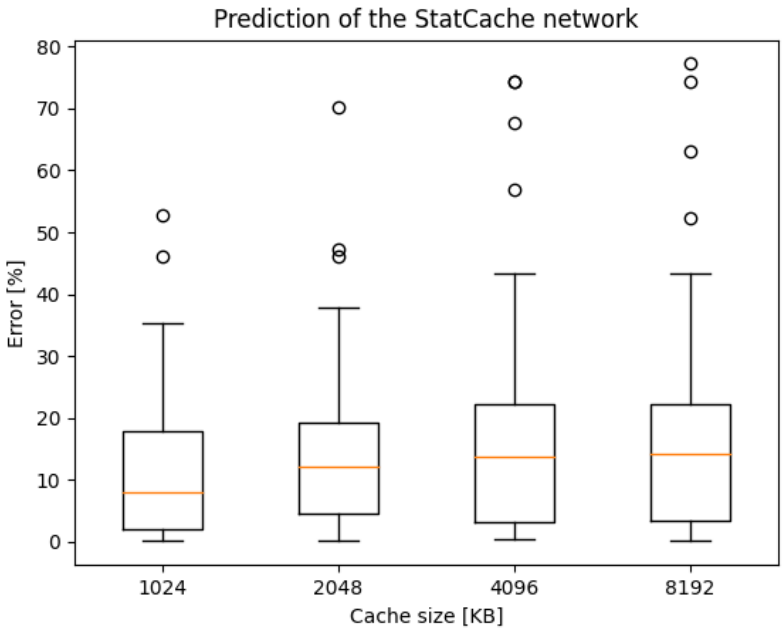
(δ′) 8192KB

Σχήμα 3.2: *Τα απόλυτα σφάλματα πρόβλεψης της StatCache για την LRU*

3.4 Ρηχό πολυεπίπεδο perceptron

Πρώτον, δημιουργήσαμε ένα απλό MLP για να προβλέψουμε τα ποσοστά αστοχίας αποκλειστικά από τις αποστάσεις επαναχρησιμοποίησης. Οι αποστάσεις επαναχρησιμοποίησης είναι αρκετά δεδομένα για την πρόβλεψη τέτοιων τιμών με μεγάλη ακρίβεια, ειδικά για την πολιτική αντικατάστασης LRU. Το StatCache προβλέπει τα ποσοστά αστοχίας με χαμηλό σφάλμα, γνωρίζοντας μόνο αυτά- επομένως, υποθέτουμε ότι υπάρχουν αρκετά δεδομένα για να κάνουμε μια αρχική πρόβλεψη και να δημιουργήσουμε ένα πρώτο μοντέλο γι′ αυτό.

Ένα στρώμα κανονικοποίησης μπορεί να ενισχύσει την ακρίβεια του μοντέλου μας. υπάρχουν τεράστιες αποκλίσεις μεταξύ των αριθμών στους κελιών του ιστογράμματος απόστασης επαναχρησιμοποίησης, καθώς αντικατοπτρίζουν τις πραγματικές προσπελάσεις στην κρυφή μνήμη. Μια απλή τυπική κλίμακα που προσαρμόζεται στο σύνολο εκπαίδευσης αποδεικνύεται πολύ χρήσιμη για την επίλυση αυτού του ζητήματος. Αυτός εκτελεί έναν απλό μετασχηματισμό $z = (x - u)/s$, όπου ξ είναι η σειρά μας, u είναι μια σειρά μέσων και s είναι η τυπική απόκλιση κάθε στήλης. Στη συνέχεια, εφαρμόζουμε αυτόν τον κλιμακωτή στο σύνολο εκπαίδευσης και στο σύνολο δοκιμής και συνεχίζουμε. Σημειώστε ότι αυτός ο κλιμακωτής πρέπει να εκπαιδευτεί μόνο στο σύνολο εκπαίδευσης για να μην δημιουργήσει ψευδώς καλύτερα αποτελέσματα. Ο υπολογισμός της τυπικής απόκλισης σε ολόκληρο το



Σχήμα 3.3: *Τα απόλυτα σφάλματα πρόβλεψης της StatCache.*

σύνολο δεδομένων θα μπορούσε να εισάγει μια μεροληψία που θα προκαλούσε καλύτερο αποτέλεσμα από το πραγματικό στο σύνολο δοκιμής.

Το μοντέλο που δημιουργήσαμε είναι ένα MLP με ένα κρυφό επίπεδο 512 νευρώνων. Λαμβάνει ως είσοδο το ιστόγραμμα της απόστασης επαναχρησιμοποίησης και παράγει τέσσερις εξόδους, μία για κάθε μέγεθος κρυφής μνήμης (1 MB, 2 MB, 4 MB και 8 MB). Αυτές οι τέσσερις έξοδοι είναι οι προβλέψεις του δικτύου για τις αστοχίες στην κρυφή μνήμη για το πώς θα ανταποκριθεί το μηχάνημα με τη συγκεκριμένη αρχιτεκτονική στο συγκεκριμένο πρόβλημα.

3.5 Δίκτυο LSTM

Ένα LSTM δέχεται ως είσοδο μια σειρά διαδοχικών τιμών, τις επαναλαμβάνει και υπολογίζει για κάθε τιμή μια έξοδο, διατηρώντας σε μια παράμετρο κρυφής κατάστασης τις τιμές που έχουν ήδη περάσει. Για το πρόβλημά μας, μετατρέψαμε τον πίνακα των αποστάσεων επαναχρησιμοποίησης σε σειρά και τον τροφοδοτήσαμε σε ένα δίκτυο LSTM. Έτσι, το δίκτυο μπορεί να υπολογίσει μια τιμή για κάθε έναν από τους κάδους του ιστογράμματος των αποστάσεων επαναχρησιμοποίησης, λαμβάνοντας υπόψη τις προηγούμενες τιμές που έχουν περάσει από τους κάδους μικρότερων αποστάσεων επαναχρησιμοποίησης.

Ομοίως, θέλουμε το δίκτυο να μπορεί να υπολογίζει μια τιμή για μια συγκεκριμένη τιμή του ιστογράμματος, δεδομένων των τιμών που ακολουθούν μετά από αυτήν από τους κάδους μεγαλύτερων αποστάσεων επαναχρησιμοποίησης. Για το λόγο αυτό, αντιστρέψαμε το ιστόγραμμα της απόστασης επαναχρησιμοποίησης και το τροφοδοτήσαμε σε ένα δεύτερο επίπεδο μονάδων LSTM. Για να λειτουργήσει αυτό το δεύτερο στρώμα ως συνέχεια του πρώτου στρώματος, το κελί εξόδου και η κρυφή κατάσταση πρέπει να αρχικοποιηθούν ως οι τελικές από το πρώτο. Τώρα, το δεύτερο στρώμα θα λειτουργεί ως συνέχεια του πρώτου στρώματος με ξεχωριστά βάρη.

Αρχικά, η σκέψη πίσω από αυτό το δίκτυο ήταν να αξιοποιήσει τον μηχανισμό της προσοχής. Αυτό θα μας ωφελούσε όσον αφορά την κατανόηση του πόσο σημαντικό είναι ένα χαρακτηριστικό και πού να εστιάσουμε όταν εξετάζουμε τα δεδομένα. Με αυτό το σκεπτικό, υλοποιήσαμε ένα στρώμα προσοχής αμέσως μετά τα δύο στρώματα LSTM για να τα συνδυάσουμε. Τα στρώματα προσοχής είναι μηχανισμοί προσοχής εμπνευσμένοι από τη γνωστική προσοχή του ανθρώπινου εγκεφάλου. Ανιχνεύουν τη σημασία κάθε σημείου δεδομένων μέσα στις εξόδους μέσω μιας συνάρτησης softmax [25].

Παρόλο που η διαίσθηση της χρήσης ενός τέτοιου δικτύου φαίνεται σωστή, η εκπαίδευση και η έξοδος δεν απέδωσαν τόσο καλά όσο αναμενόταν. Οι προβλέψεις του ήταν ελαφρώς χειρότερες από αυτές του StatCache και για το λόγο αυτό δεν θα τις παρουσιάσουμε. Αυτό είναι πιθανότατα συνέπεια του ότι δεν είχαμε αρκετά δεδομένα εκπαίδευσης. Ο πίνακας προσοχής υπολογίζεται από κάθε έξοδο κάθε μονάδας LSTM εντός των στρωμάτων LSTM. Αυτό θέτει ένα τεράστιο βάρος σε κάθε μία από αυτές και περιλαμβάνει πάρα πολλά βάρη για να βελτιστοποιηθούν με την εκπαίδευση.

Εναλλακτικά, όπως απεικονίζεται στο σχήμα 3.4, επιλέξαμε να χρησιμοποιήσουμε ένα πυκνό στρώμα περσεπτρον για να κατανοήσουμε τις εξόδους των στρωμάτων LSTM. Ένα στρώμα ρονσατενατε και ένα στρώμα φλαττεν μετατρέπουν τα σχήματα εξόδου από τα στρώματα LSTM σε σχήματα αποδεκτά από το πυκνό στρώμα. Αυτό το πυκνό δίκτυο κατανοεί καλύτερα το πρόβλημα και μπορεί να προβλέψει τα ποσοστά αστοχίας με μεγαλύτερη ακρίβεια.

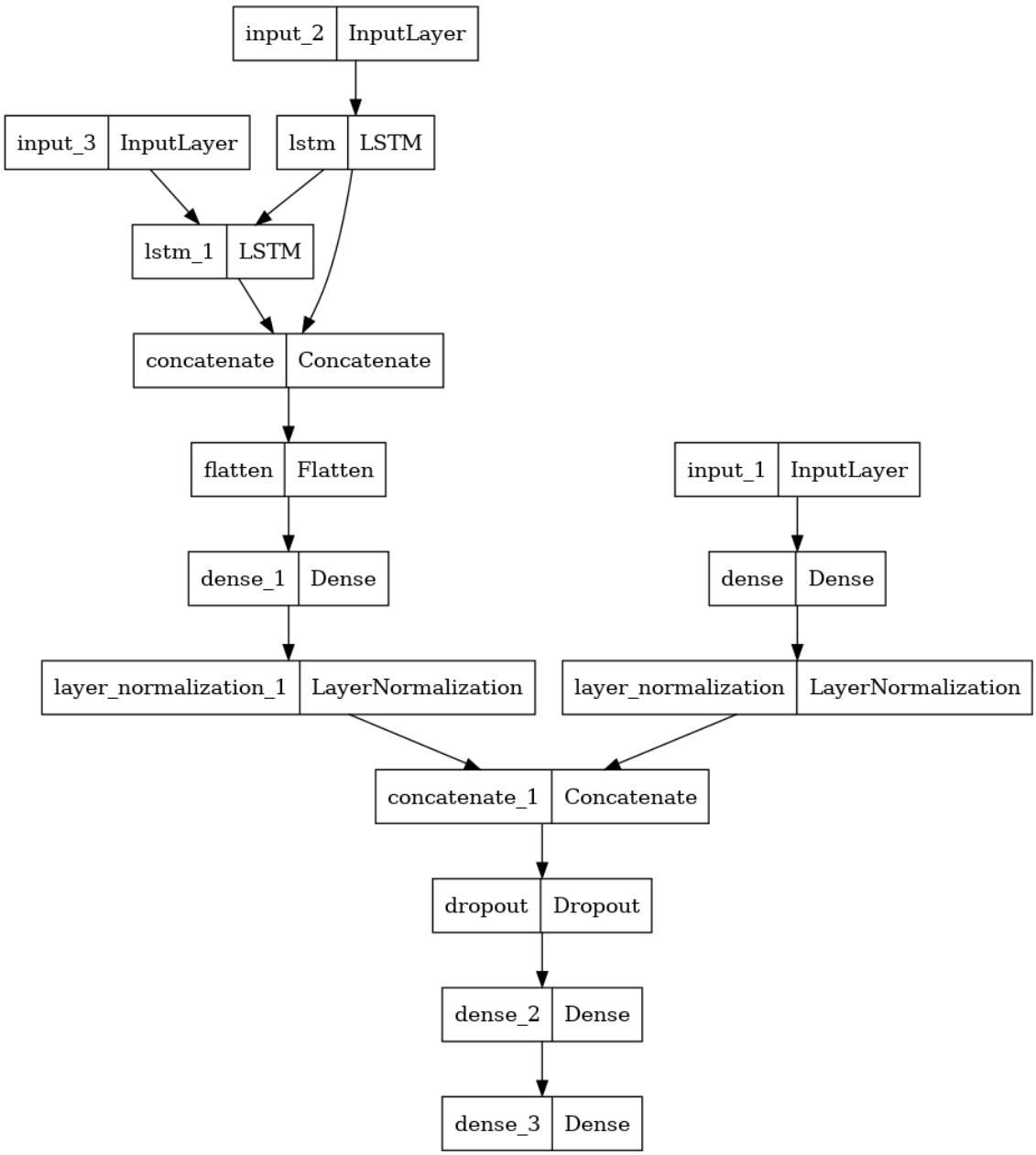
Για αυτό το δίκτυο, εκτός από τις αποστάσεις επαναχρησιμοποίησης, υλοποιήσαμε τον κλάδο μετασχηματισμού. Προσθέσαμε τις ενσωματώσεις του κώδικα σε ένα παράλληλο στρώμα περσεπτρονς, input_1 στο σχήμα 3.4. Οι ενσωματώσεις που προκύπτουν από το IR2Vec δεν είναι κανονικοποιημένες, οπότε μια κανονικοποίηση με παρόμοιο κλιμακωτή όπως χρησιμοποιήσαμε για τις αποστάσεις επαναχρησιμοποίησης μας παρέχει καλύτερα αποτελέσματα. Στη συνέχεια, συνδέσαμε όλα αυτά στο τελευταίο κρυφό στρώμα perceptron και, τέλος, στο στρώμα εξόδου. Όλα τα πυκνά νευρωνικά δίκτυα χρησιμοποιούν μια συνάρτηση ενεργο-

ποίησης relu εκτός από το στρώμα εξόδου, το οποίο χρησιμοποιεί τη σιγμοειδή συνάρτηση, δεδομένου ότι ο λόγος αστοχίας είναι μια πιθανότητα στο εύρος [0, 1].

Προσθέσαμε μερικά στρώματα κανονικοποίησης μεταξύ των μικρών πυκνών στρωμάτων των εισόδων και του μεγάλου πυκνού στρώματος της εξόδου. Αυτά φαίνεται να έχουν θετικό αντίκτυπο στο αποτέλεσμα του μοντέλου. Κάθε ένα από αυτά εκπαιδεύεται στα δεδομένα του συνόλου εκπαίδευσης και κανονικοποιεί τις εξόδους που δημιουργούνται από τα πυκνά νευρωνικά δίκτυα. Η κανονικοποίηση των στρωμάτων είναι ευεργετική, παρόλο που τα δεδομένα εισόδου είναι κανονικοποιημένα, διότι επιτρέπει ομαλότερες κλίσεις, ταχύτερη εκπαίδευση και καλύτερη ακρίβεια γενίκευσης. Δοκιμάσαμε όλες τις πιθανές τοποθεσίες αυτών των στρωμάτων στο δίκτυο και η καλύτερη απόδοση ήταν ακριβώς μετά τα μικρά, πυκνά στρώματα.

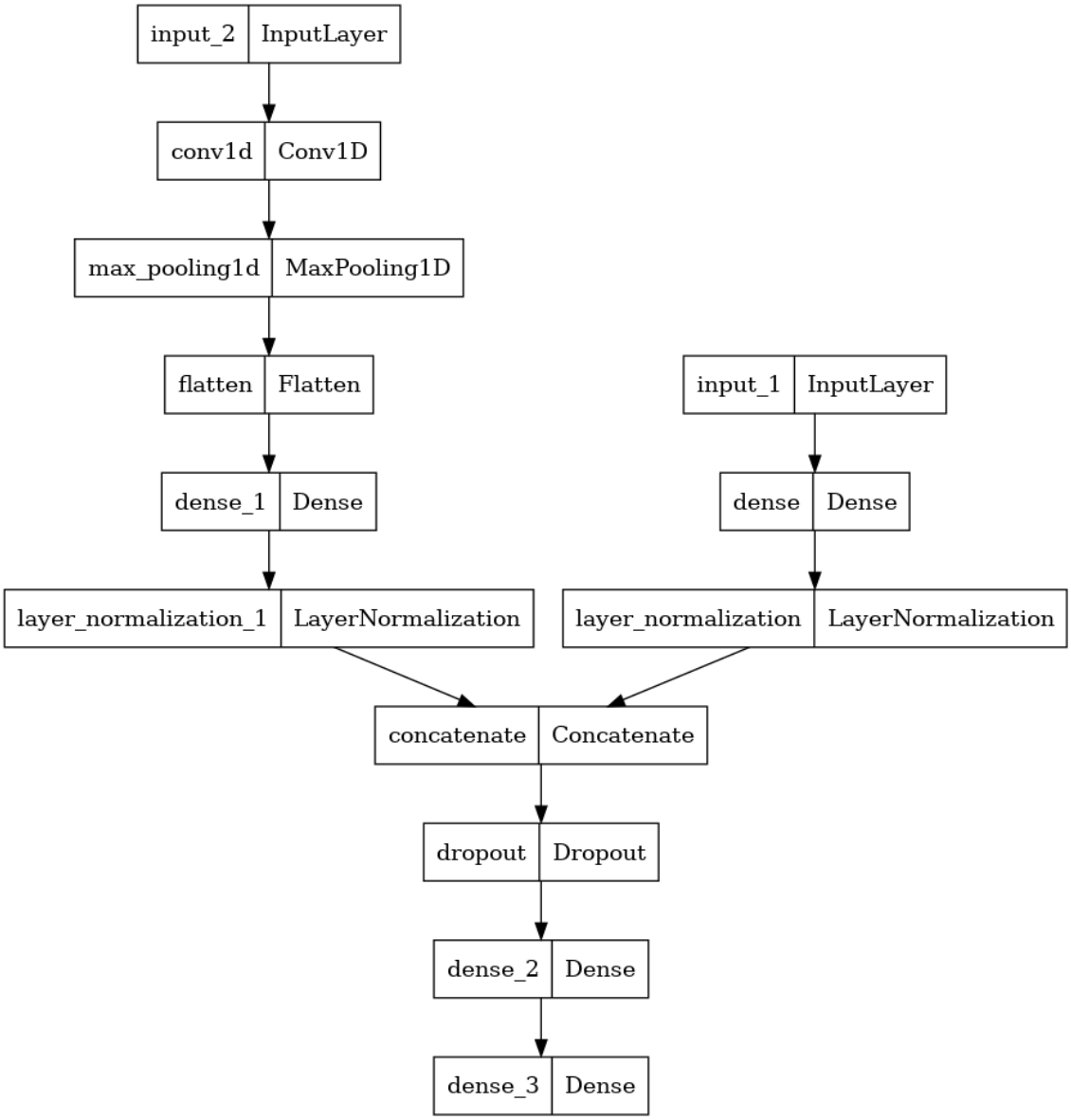
Όσον αφορά τις ιδιαιτερότητες του δικτύου, τα στρώματα LSTM μας είναι μεγέθους 4 μονάδων το καθένα. Πειραματιστήκαμε με πολλές τιμές για το δίκτυο στο εύρος 1-16 μονάδες LSTM και ανακαλύψαμε ότι οι 4 μονάδες ανά στρώμα φαίνεται να είναι η καλύτερη υπολογιστική τιμή. τα πυκνά στρώματα αμέσως μετά (dense και dense_1 στο σχήμα 3.4) έχουν μέγεθος 128 νευρώνων, ενώ το τελευταίο πυκνό στρώμα (dense_2) είναι μεγαλύτερο με μέγεθος 1024. Το μέγεθος των μικρότερων στρωμάτων δεν επηρεάζει πολύ την ακρίβεια- τα μεγέθη 128, 256 και 512 έχουν παρόμοια αποτελέσματα. Αυτό το μέγεθος του μεγαλύτερου στρώματος, από την άλλη πλευρά, λειτούργησε καλύτερα από άλλες τιμές όπως 512 και 2048.

Επιπλέον, χρησιμοποιούμε ένα στρώμα εγκατάλειψης. Αυτό το στρώμα θέτει τυχαία το 20% των εισόδων του στο μηδέν για να αποτρέψει την υπερβολική προσαρμογή.



Σχήμα 3.4: *Δομή του δικτύου LSTM. input_1 είναι οι ενσωματώσεις, input_2 το ιστόγραμμα αποστάσεων επαναχρησιμοποίησης και input_3 το ιστόγραμμα αποστάσεων επαναχρησιμοποίησης με αντίστροφη σειρά (από μεγαλύτερη τιμή προς την μικρότερη).*

3.6 Συνελικτικό νευρωνικό δίκτυο



Σχήμα 3.5: *Δομή του συνελικτικού νευρωνικού δικτύου. input_1 είναι οι ενσωματώσεις και input_2 είναι οι αποστάσεις επαναχρησιμοποίησης.*

Τα στρώματα συνέλιξης υπάρχουν για να βοηθούν το δίκτυο να εντοπίζει ομοιότητες μεταξύ χαρακτηριστικών που βρίσκονται κοντά το ένα στο άλλο, οι οποίες μπορεί να χαθούν ή να αγνοηθούν αν χρησιμοποιήσουμε ένα πυκνό δίκτυο. Οι αποστάσεις γειτονικής επαναχρησιμοποίησης με συνελίξεις θα μπορούσαν να οδηγήσουν σε καλύτερη απόδοση, καθώς η είσοδος στο πυκνό στρώμα θα είναι πιο κατατοπιστική.

Στη συνέχεια, αμέσως μετά, ένα στρώμα συγκέντρωσης έχει ως σκοπό τη μείωση της διαστατικότητας της προηγούμενης εξόδου. Το στρώμα συνελικτικού δικτύου έχει ν φίλτρα, παράγοντας έτσι ν αποτελέσματα ανά είσοδο ιστογράμματος. Η είσοδος ιστογράμματος είναι ένας πίνακας σχήματος (896,1)- επομένως, ο πίνακας εξόδου θα είναι $((896 - n)/step, n)$. Χρησιμοποιούμε το στρώμα συσσώρευσης μεγίστου για να μειώσουμε αυτές τις διαστάσεις

οριζόντια σε $((896 - n)/(step * K), n)$, όπου K είναι το μέγεθος του πυρήνα. Υπάρχουν εναλλακτικές λύσεις στο στρώμα συσσώρευσης για τη μείωση των διαστάσεων, μερικές από τις πιο δημοφιλείς είναι η συγκέντρωση μεγίστου και μέσου ή η χρήση μεγαλύτερου βήματος. Σε αυτό το πρόβλημα, ένα μικρό βήμα και το στρώμα συσσώρευσης μεγίστου υπολόγισαν τα καλύτερα αποτελέσματα, οπότε χρησιμοποιήσαμε αυτά.

Και πάλι, όπως ακριβώς και στην εφαρμογή LSTM, η έξοδος αυτού του στρώματος συγκέντρωσης συνδέεται με ένα πυκνό στρώμα MLP με relu και ένα στρώμα κανονικοποίησης. Το υπόλοιπο δίκτυο είναι το ίδιο όπως και προηγουμένως.

Αξίζει να σημειωθεί ότι, με αυτό το δίκτυο, οι μη κλιμακούμενες αποστάσεις επαναχρησιμοποίησης λειτουργούν καλύτερα. Ο κλιμακωτής στις μη κλιμακωτές αποστάσεις επαναχρησιμοποίησης που χρησιμοποιήσαμε προηγουμένως δεν βελτιώνει καθόλου την πρόβλεψή μας. Ακριβώς το αντίθετο, η πρόβλεψη χειροτερεύει επειδή η κλιμάκωση επηρεάζει την έξοδο της συνέλιξης- αυτό έχει ως αποτέλεσμα μικρότερες διαφορές μεταξύ των τιμών που προκύπτουν από τη συνέλιξη και κατά συνέπεια χειρότερη κατανόηση. Το δίκτυο προτιμά τις μη κλιμακωμένες αποστάσεις επαναχρησιμοποίησης που περνούν από αυτό, οι οποίες στη συνέχεια θα περάσουν από το πρώτο στρώμα perceptron. Στη συνέχεια, ένα στρώμα κανονικοποίησης θα κανονικοποιήσει τα αποτελέσματα αυτού του πυκνού στρώματος, αντί να έχει προ-κλιμακωμένα δεδομένα. Δοκιμάστηκαν όλα τα πιθανά σημεία για την εισαγωγή στρωμάτων κανονικοποίησης, με όλους τους δυνατούς συνδυασμούς, και απλά ένα στρώμα μετά το περσεπτρον παράγει τις καλύτερες προβλέψεις.

Όσον αφορά τις ιδιαιτερότητες του δικτύου, το CNN δίκτυό μας αποτελείται από φίλτρα που επαναλαμβάνουν τις αποστάσεις επαναχρησιμοποίησης εισόδου με ένα βήμα και υπολογίζουν τη συνέλιξη πάνω σε έναν πυρήνα. Πειραματιστήκαμε με πολλούς συνδυασμούς για αυτούς τους τρεις αριθμούς (αριθμός φίλτρου, βήμα και μέγεθος πυρήνα) στις περιοχές $[1, 20]$ για κάθε αριθμό. Η δοκιμή αυτή κατέληξε στο συμπέρασμα ότι ο καλύτερος συνδυασμός είναι 6 φίλτρα, μέγεθος πυρήνα 8 και στριδε 4. Τα πυκνά στρώματα αμέσως μετά (dense και dense_1 στο σχήμα 3.5) έχουν μέγεθος 512 νευρώνων, ενώ το τελευταίο πυκνό στρώμα (dense_2) είναι μεγαλύτερο με μέγεθος 1024. Το μέγεθος των μικρότερων στρωμάτων επηρεάζει λίγο την ακρίβεια- τα μεγέθη 128 και 256 έχουν γεωμετρικό μέσο όρο απόλυτων σφαλμάτων πρόβλεψης περίπου 0,5% λιγότερο. Αυτό το μέγεθος του μεγαλύτερου στρώματος, από την άλλη πλευρά, λειτούργησε καλύτερα από άλλες τιμές όπως 512 και 2048.

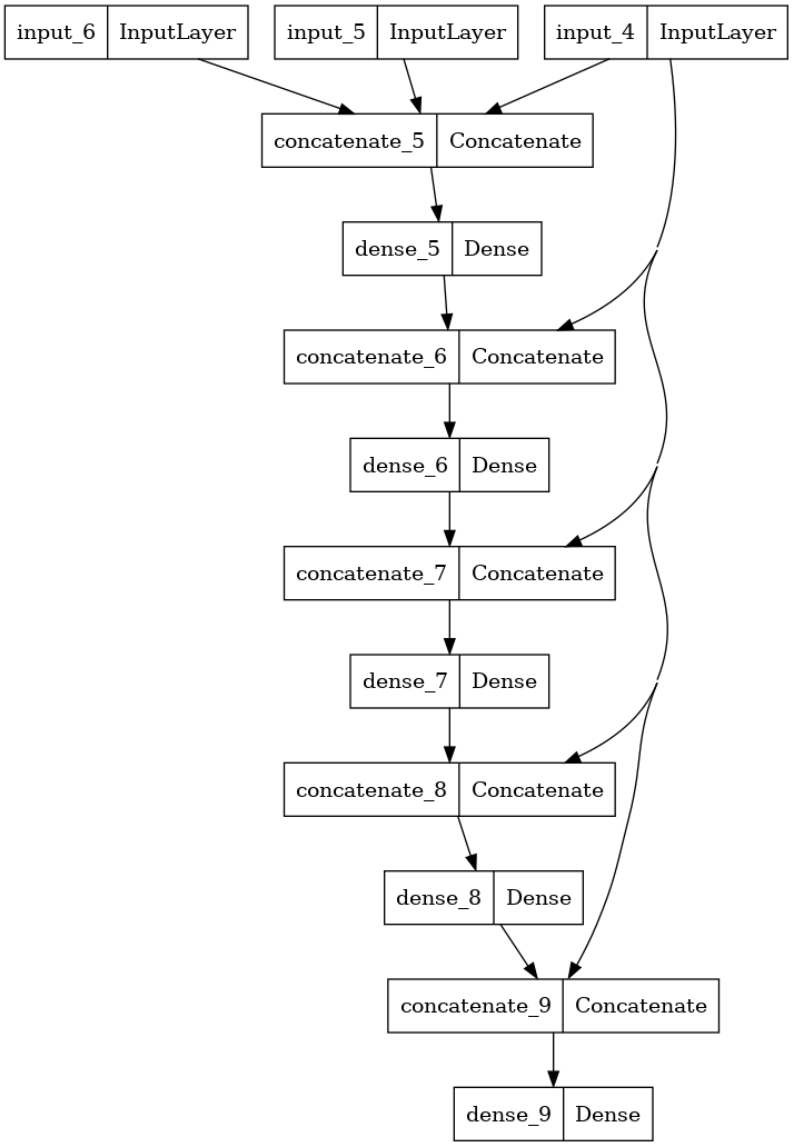
Η εκπαίδευση αυτού του δικτύου χρειάζεται κατά μέσο όρο 6 ms/step ή 3 δευτερόλεπτα/εποχή, κάτι που είναι πολύ πιο γρήγορο σε σύγκριση με τα 50 ms/step ή 15 δευτερόλεπτα/εποχή του δικτύου LSTM.

3.7 Βαθύ νευρωνικό δίκτυο

Αφού εκτιμήσαμε την ποιότητα των προηγούμενων αποτελεσμάτων και για τα 4 μεγέθη κρυφής μνήμης (1, 2, 4 και 8 MB), θελήσαμε να δούμε αν θα μπορούσαμε να δημιουργήσουμε ένα δίκτυο που να προβλέπει το ποσοστό αστοχίας σε οποιοδήποτε μέγεθος κρυφής μνήμης LLC. Αυτό έχει ως στόχο την ακριβή εκτίμηση του ποσοστού αστοχίας για κάθε μέγεθος κρυφής μνήμης και, τελικά, να δείξει ποια είναι η καλύτερη αρχιτεκτονική κρυφής

μνήμης για ένα συγκεκριμένο πρόβλημα. Για τον σκοπό αυτό, προσομοιώσαμε ξανά τα ίχνη, χρησιμοποιώντας ενδιάμεσα μεγέθη κρυφής μνήμης για να δημιουργήσουμε ένα μεγαλύτερο σύνολο δεδομένων. Αυτό περιέχει τα δεδομένα για τα μεγέθη κρυφής μνήμης LLC [768, 1024, 1536, 2048, 3152, 4096, 6044, 8192] KB με την πολιτική αντικατάστασης LRU.

Αυτό το δίκτυο μπορεί να χρησιμοποιηθεί με δύο διαφορετικούς τρόπους. Εάν έχουμε ένα πρόγραμμα για το οποίο έχουμε τα δεδομένα σε ένα ή περισσότερα μηχανήματα, μπορούμε να το προσθέσουμε στο σύνολο εκπαίδευσης και να προβλέψουμε πόσο υψηλό ποσοστό αστοχίας θα έχει με άλλα μεγέθη κρυφής μνήμης. Ο δεύτερος τρόπος χρήσης είναι για ένα νέο πρόγραμμα που δεν έχει δει το δίκτυο. Αυτός ο δεύτερος τρόπος θα είναι γενικά πιο χρήσιμος από τα τρία προηγούμενα δίκτυα, καθώς θα μπορεί να προβλέψει την αναλογία αστοχίας για οποιοδήποτε μέγεθος κρυφής μνήμης και δεν θα περιορίζεται μόνο σε αυτά στα οποία έχει εκπαιδευτεί.



Σχήμα 3.6: *Βαθύ νευρωνικό δίκτυο για την πρόβλεψη άλλων μεγεθών κρυφής μνήμης. input_4 είναι το μέγεθος της κρυφής μνήμης, input_5 είναι οι αποστάσεις επαναχρησιμοποίησης και input_6 είναι οι ενσωματώσεις.*

Για αυτό το δίκτυο, όπως φαίνεται στο σχήμα 3.6, χρησιμοποιούμε 4 κρυφά στρώματα. Θέλουμε κάθε στρώμα να κάνει προβλέψεις ανάλογα με το μέγεθος της κρυφής μνήμης- γι' αυτό τροφοδοτούμε το μέγεθος εισόδου σε κάθε κρυφό στρώμα συνδιάζοντάς το με την έξοδο του προηγούμενου στρώματος. Κάθε ένα από τα κρυφά στρώματα έχει 512 νευρώνες- τα τέσσερα πρώτα χρησιμοποιούν τη συνάρτηση ενεργοποίησης ρελυ, ενώ το τελευταίο στρώμα και το στρώμα εξόδου χρησιμοποιούν τη σιγμοειδή συνάρτηση ενεργοποίησης. Αυτή η διαμόρφωση δοκιμάστηκε και φάνηκε να αποδίδει καλύτερα. Περισσότερα κρυφά στρώματα ή περισσότεροι νευρώνες δεν απέδιδαν καλύτερη πρόβλεψη, ίσως λόγω της δημιουργίας πολλών μεταβλητών.

Και οι τρεις είσοδοι περιέχουν μια κανονικοποιημένη έκδοση των αποστάσεων επαναχρησιμοποίησης, των ενσωματώσεων και των μεγεθών της κρυφής μνήμης. Οι αποστάσεις επαναχρησιμοποίησης και οι ενσωματώσεις κανονικοποιούνται με την ίδια μέθοδο όπως και στα άλλα δίκτυα. Το μέγεθος της εισόδου διαιρέθηκε με 512KB, με αποτέλεσμα να προκύψουν τιμές που κυμαίνονται από 1,5 - 16,0 για κάθε μία από τις κρυφές μνήμες.

Κεφάλαιο 4

Σύγκριση αποτελεσμάτων

4.1 Πρόβλεψη για άγνωστα προβλήματα

4.1.1 Πρόβλεψη για την πολιτική αντικατάστασης LRU

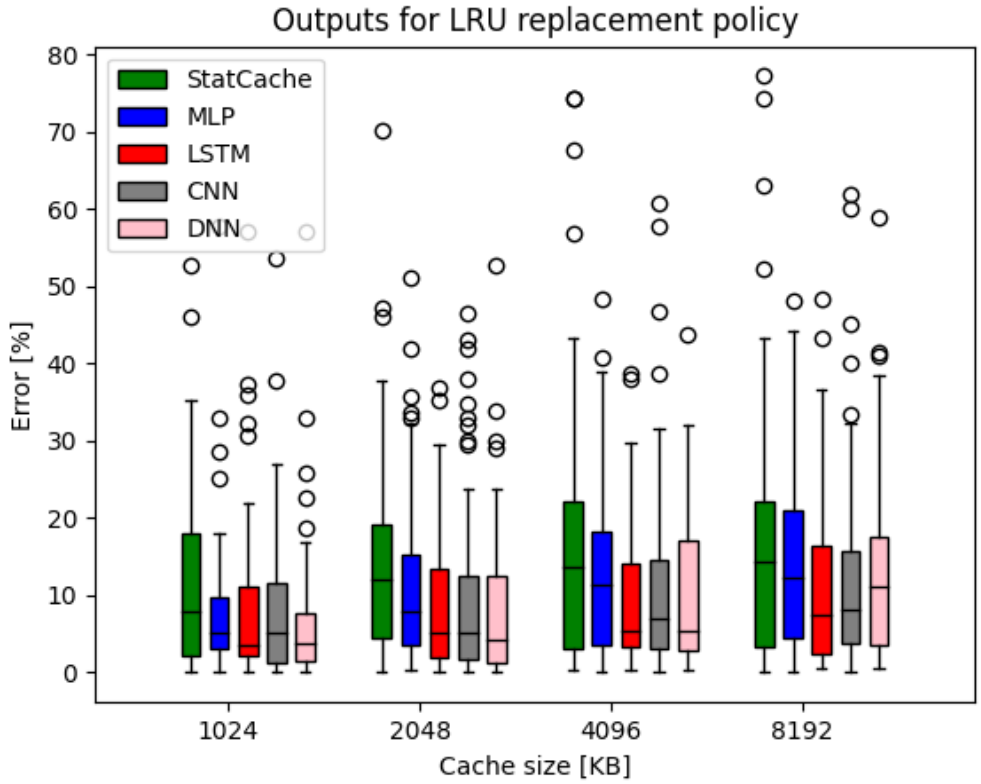
Το πρόβλημα που πρόκειται να συζητήσουμε είναι πολύ εύκολο να περιγραφεί. Προσπαθούμε να προβλέψουμε τα ποσοστά αστοχίας οποιασδήποτε εφαρμογής που συναντούν τα δίκτυά μας. Για να το δείξουμε αυτό χρησιμοποιούμε την προαναφερθείσα μέθοδο leave-one-out cross-validation όπου διαχωρίζουμε ένα βενσημαρκ, εκπαιδεύουμε το δίκτυο στα άλλα βενσημαρκς και προσπαθούμε να το προβλέψουμε. Με αυτόν τον τρόπο θα έχουμε την καλύτερη δυνατή πρόβλεψη για κάθε φόρτο εργασίας χωρίς να τον έχουμε δει προηγουμένως.

Η αναζήτηση της ακριβέστερης πρόβλεψης του ποσοστού αστοχίας μας οδήγησε στην εισαγωγή τεσσάρων συνολικά μοντέλων: των δικτύων MLP, CNN, LSTM και DNN. Κάθε ένα από αυτά τα μοντέλα αντιπροσωπεύει μια ξεχωριστή προσέγγιση για την όσο το δυνατόν ακριβέστερη πρόβλεψη των αναλογιών αστοχίας στην κρυφή μνήμη. Ας συγκρίνουμε και ας συζητήσουμε τις συνολικές προβλέψεις των δικτύων μας. Για το σκοπό αυτό, θα επιλέξουμε μια πολιτική αντικατάστασης και θα συγκρίνουμε τα σφάλματα πρόβλεψης που έχουμε υπολογίσει με κάθε ένα από τα δίκτυά μας.

Η LRU είναι η μόνη πολιτική αντικατάστασης για την οποία μπορούμε να συγκρίνουμε τόσο το Στατᾶσηε όσο και τα δίκτυα DNN. Το Στατᾶσηε επινοήθηκε και λειτουργεί μόνο με την πολιτική αντικατάστασης LRU. Το δίκτυό μας DNN είναι πιθανότατα ικανό να προβλέψει οποιαδήποτε πολιτική αντικατάστασης, αλλά θα χρειαζόμασταν περισσότερα δεδομένα για άλλες πολιτικές αντικατάστασης για να επιβεβαιώσουμε ότι πράγματι προβλέπει οποιοδήποτε μέγεθος κρυφής μνήμης. Το δίκτυο DNN από το οποίο θα παρουσιάσουμε αποτελέσματα είναι το DNN που έχει εκπαιδευτεί στα 4 μεγέθη κρυφής μνήμης (1MB, 2MB, 4MB και 8MB) με την ίδια μέθοδο διασταυρούμενης επικύρωσης leave-one-out όπως και τα άλλα δίκτυα.

Είναι σαφές ότι όλα τα δίκτυά μας ξεπερνούν σημαντικά τις προβλέψεις της StatCache. Οι ακραίες τιμές, τα πλαίσια και οι διάμεσοι των απόλυτων σφαλμάτων πρόβλεψης στο θηκόγραμμα του 4.1 είναι αισθητά υψηλότερα για την πρόβλεψη της StatCache. Είναι επίσης σαφώς αξιοσημείωτο ότι οι γεωμετρικοί μέσοι των σφαλμάτων πρόβλεψης στον πίνακα 4.1 είναι σαφώς χειρότεροι από οποιοδήποτε άλλο δίκτυο στα τρία μικρότερα μεγέθη κρυφής μνήμης και ίσοι με τους χειρότερους για την κρυφή μνήμη μεγέθους 8MB.

Είναι επίσης σαφώς ορατό ότι για μεγαλύτερα μεγέθη κρυφής μνήμης, τα απόλυτα σφάλ-



Σχήμα 4.1: *Σφάλματα πρόβλεψης των δικτύων MLP, LSTM, CNN, DNN και StatCache για LLC με πολιτική αντικατάστασης LRU*

ματα πρόβλεψης γίνονται μεγαλύτερα. Για μεγαλύτερα μεγέθη κρυφής μνήμης, ο μέσος λόγος αστοχίας μειώνεται δραστικά. Αυτό μας οδηγεί στο συμπέρασμα ότι για μεγαλύτερες κρυφές μνήμες, παρόλο που είναι ευκολότερο να προβλέψουμε ένα εύρος όπου θα είναι τα ποσοστά αστοχίας, είναι δυσκολότερο να προβλέψουμε τα ακριβή ποσοστά αστοχίας.

Το ρηχό MLP είναι συνολικά το πιο αδύναμο από τα δίκτυά μας. Το γεγονός ότι δεν λαμβάνει τόσες πληροφορίες όσο τα άλλα δίκτυα, λόγω του ότι δεν έχει τον κλάδο του μετασχηματιστή, καθώς και η χειρότερη κατανόηση των αποστάσεων επαναχρησιμοποίησης, έχουν ως αποτέλεσμα μεγαλύτερα σφάλματα από τα άλλα δίκτυα. Τα σφάλματα πρόβλεψής του έχουν υψηλότερες διαμέσους, μεγαλύτερα κουτιά και πολλές ακραίες τιμές σε σύγκριση με τα σφάλματα πρόβλεψης των δικτύων CNN και LSTM. Αυτό το βλέπουμε και από το γεγονός ότι παράγει σταθερά το δεύτερο μεγαλύτερο σφάλμα πρόβλεψης στον πίνακα 4.1. Ο στόχος αυτού του δικτύου ήταν να δείξει ότι είναι δυνατόν με ένα απλό ΜΛΠ να παράγει παρόμοια, αν όχι καλύτερα, αποτελέσματα από το StatCache χρησιμοποιώντας μόνο τις αποστάσεις επαναχρησιμοποίησης. Τα αποτελέσματα δείχνουν ότι επιτύχαμε με επιτυχία αυτόν τον στόχο.

Τα άλλα τρία δίκτυα έχουν πρόσβαση στον μετασχηματισμένο κώδικα του προγράμματος, οπότε οι προβλέψεις τους είναι αναμφισβήτητα καλύτερες από αυτές του MLP. Για να αναλύσουμε τις προβλέψεις τους με μεγαλύτερη ακρίβεια, θα εξετάσουμε τα ιστογράμματα των απόλυτων σφαλμάτων πρόβλεψης 4.2. Δεδομένου ότι τα ιστογράμματα είναι παρόμοια

για άλλα μεγέθη κρυφής μνήμης ως προς το σχήμα, θα παρουσιάσουμε μόνο αυτά από το μέγεθος της κρυφής μνήμης 4MB που είναι αντιπροσωπευτικά.

Network	**Cache size [MB]**			
	1	**2**	**4**	**8**
StatCache	5.4%	8.3%	8.7%	8.5%
MLP	4.1%	7.2%	6.3%	8.5%
LSTM	3.6%	3.9%	5.3%	6.4%
CNN	3.9%	4.6%	5.5%	6.2%
DNN	3.6%	4.0%	6.0%	8.1%

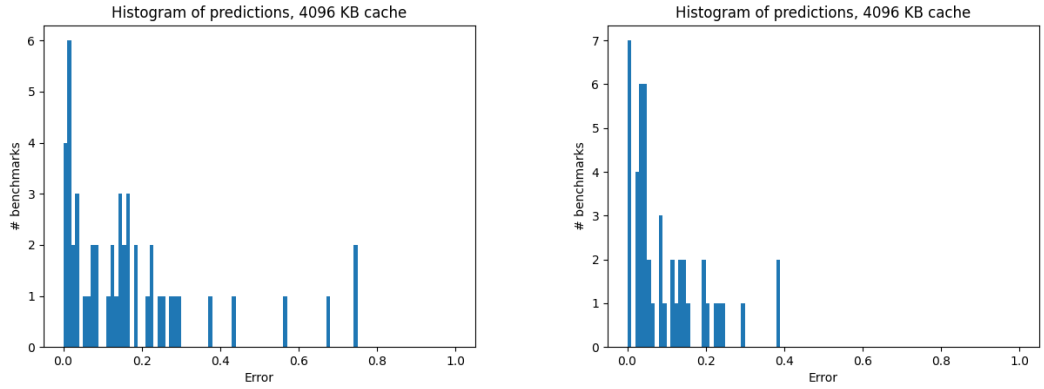
Πίνακας 4.1: *Γεωμετρικός μέσος όρος των απόλυτων σφαλμάτων πρόβλεψης για τις κρυφές μνήμες με πολιτική αντικατάστασης LRU.*

Είναι προφανές ότι η πρόβλεψη της StatCache είναι χειρότερη από οποιαδήποτε άλλη πρόβλεψη. Πολλές μεγάλες ακραίες τιμές, λιγότερα σημεία αναφοράς κοντά στο μηδενικό σφάλμα και μια σχεδόν ομοιόμορφη κατανομή στο εύρος των τιμών της καθιστούν σαφές γιατί οι προβλέψεις του StatCache είναι χειρότερες. Η υψηλότερη στήλη για όλα τα άλλα ιστογράμματα είναι η μικρότερη. Τα απόλυτα σφάλματα πρόβλεψης των δικτύων LSTM και CNN έχουν πολύ παρόμοια σχήματα κοντά στο μηδέν, με πολλά σημεία αναφοράς πολύ κοντά και λιγότερα πιο μακριά. Αυτό που τα διαφοροποιεί είναι ότι το CNN έχει πολλές ακραίες τιμές. Αυτό είναι επίσης ορατό στο 4.1, όπου τα απόλυτα σφάλματα πρόβλεψης του δικτύου CNN έχουν πολύ περισσότερες ακραίες τιμές από τα σφάλματα πρόβλεψης του δικτύου LSTM.

Η αξιολόγηση της πρόβλεψης του δικτύου DNN σε σύγκριση με τα άλλα είναι λίγο πιο δύσκολη. Τα απόλυτα σφάλματα πρόβλεψής του φαίνεται να έχουν μια πιο ομοιόμορφη κατανομή στο εύρος από 0 έως 0,2 με πολύ λίγες ακραίες τιμές. Αυτό είναι ελαφρώς χειρότερο από τα απόλυτα σφάλματα πρόβλεψης του δικτύου LSTM. Όταν εξετάζουμε το αθροιστικό θηκόγραμμα 4.1, οι προβλέψεις του φαίνεται να είναι καλύτερες από αυτές του LSTM για μικρότερες τιμές του μεγέθους της κρυφής μνήμης. Οι γεωμετρικοί μέσοι όροι και για τις δύο αυτές περιπτώσεις είναι σχεδόν ίσοι. Αυτό σημαίνει ότι το δίκτυο έχει την ικανότητα να προβλέπει τα ποσοστά αστοχίας της κρυφής μνήμης εξίσου καλά με το δίκτυο LSTM, αν όχι καλύτερα, για μικρά μεγέθη κρυφής μνήμης. Αυτό σημαίνει επίσης ότι αυτό το δίκτυο με δομή DNN έχει αδυναμία για μεγαλύτερα μεγέθη κρυφής μνήμης.

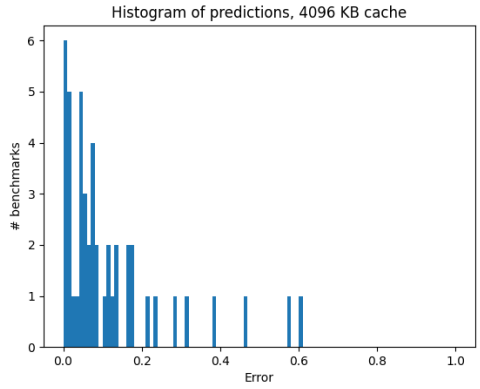
4.1.2 Πρόβλεψη οποιουδήποτε μεγέθους κρυφής μνήμης για την πολιτική αντικατάστασης LRU

Αφού αναλύσαμε τα αποτελέσματα όλων των δικτύων σε αυτά τα τέσσερα μεγέθη κρυφής μνήμης από τα δίκτυά μας, θέλουμε να παρουσιάσουμε την ευελιξία του δικτύου DNN. Το δίκτυο DNN δέχεται ως είσοδο την παράμετρο του μεγέθους της κρυφής μνήμης και μπορεί να κατανοήσει την αναλογία αστοχίας οποιουδήποτε μεγέθους κρυφής μνήμης. Ως εκ τούτου, έχουμε τα μισς ρατιος περισσότερων μεγεθών κρυφής μνήμης στην LLC για να δείξουμε ότι αυτό το δίκτυο DNN μπορεί πράγματι να τα προβλέψει.

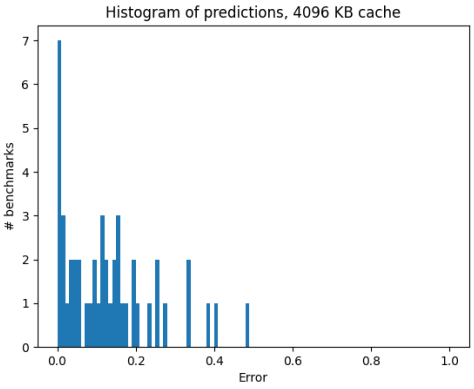


(α΄) StatCache

(β΄) LSTM



(γ΄) CNN



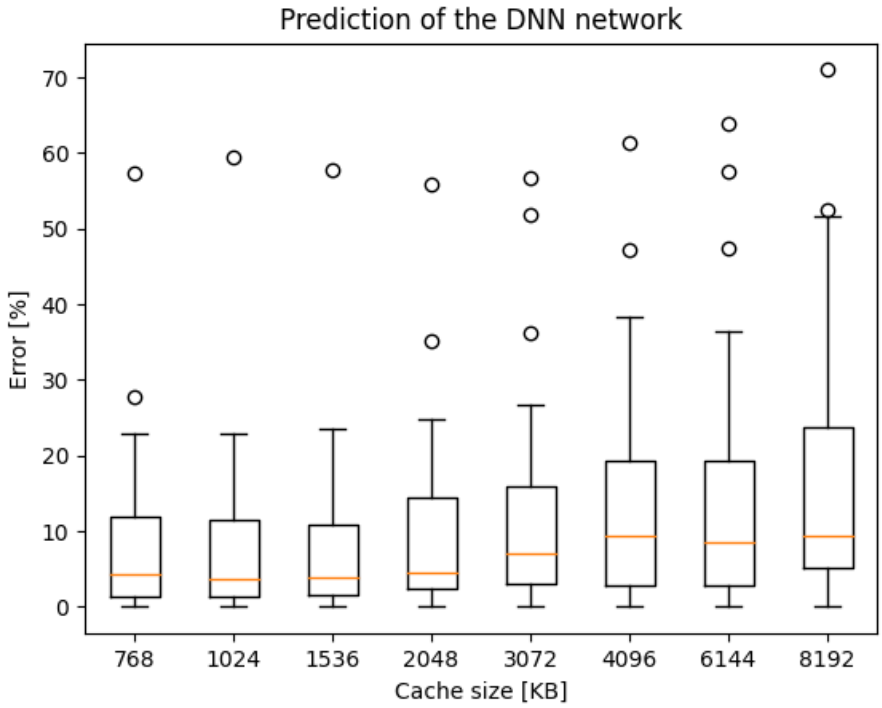
(δ΄) DNN

Σχήμα 4.2: *StatCache, LSTM, CNN, and DNN absolute prediction errors for LLC of size 4MB with LRU replacement policy*

Σε αυτή την ενότητα, προσπαθούμε να προβλέψουμε οποιοδήποτε άγνωστο πρόγραμμα για οποιοδήποτε μέγεθος κρυφής μνήμης με τη δεδομένη πολιτική αντικατάστασης. Έχουμε τα μεγέθη LLC [768, 1024, 1536, 2048, 3152, 4096, 6044, 8192] KB και πολιτική αντικατάστασης LRU. Θα χρησιμοποιήσουμε τη μέθοδο της διασταυρούμενης επικύρωσης (χρoσσ-αλιδατιον) για το σύνολο δοκιμών μας, όπως ακριβώς και στο κεφάλαιο 3.4, όπου θα διατρέξουμε τα αρχεία αναφοράς και θα τα προβλέψουμε με ένα δίκτυο που εκπαιδεύτηκε στα άλλα 46 αρχεία αναφοράς. Στο τέλος, θα έχουμε 47 προβλέψεις, μία για κάθε σημείο αναφοράς, με το δίκτυο να εκπαιδεύεται κάθε φορά σε όλα τα άλλα σημεία αναφοράς.

Για λιγότερα από 3 μεγέθη κρυφής μνήμης στο σύνολο εκπαίδευσης, η πρόβλεψη δεν έχει καμία αξία, καθώς το δίκτυο δεν μπορεί να κατανοήσει την παράμετρο του μεγέθους. Όταν το δίκτυο εκπαιδεύεται σε ένα μέγεθος κρυφής μνήμης, η παράμετρος μεγέθους δεν αλλάζει στο σύνολο εκπαίδευσης, οπότε το δίκτυο απλώς θα της αποδίδει τυχαίες τιμές. Όταν χρησιμοποιούνται δύο μεγέθη κρυφής μνήμης στο σύνολο εκπαίδευσης, το δίκτυο το κατανοεί ως δυαδικό πρόβλημα και προσπαθεί να εφαρμόσει σε αυτό μια σιγμοειδή συνάρτηση. Αυτό καθιστά τις προβλέψεις αναξιόπιστες. Έτσι, θα παρουσιάσουμε τις προβλέψεις με 3, 4 και 6 μεγέθη κρυφής μνήμης στο σύνολο εκπαίδευσης.

Το δίκτυο εκπαιδεύτηκε με τα μεγέθη της κρυφής μνήμης 1MB, 2MB και 8MB στο



Σχήμα 4.3: *Απόλυτα σφάλματα πρόβλεψης του δικτύου DNN, εκπαιδευμένα σε μεγέθη κρυφής μνήμης 1MB, 2MB και 8MB.*

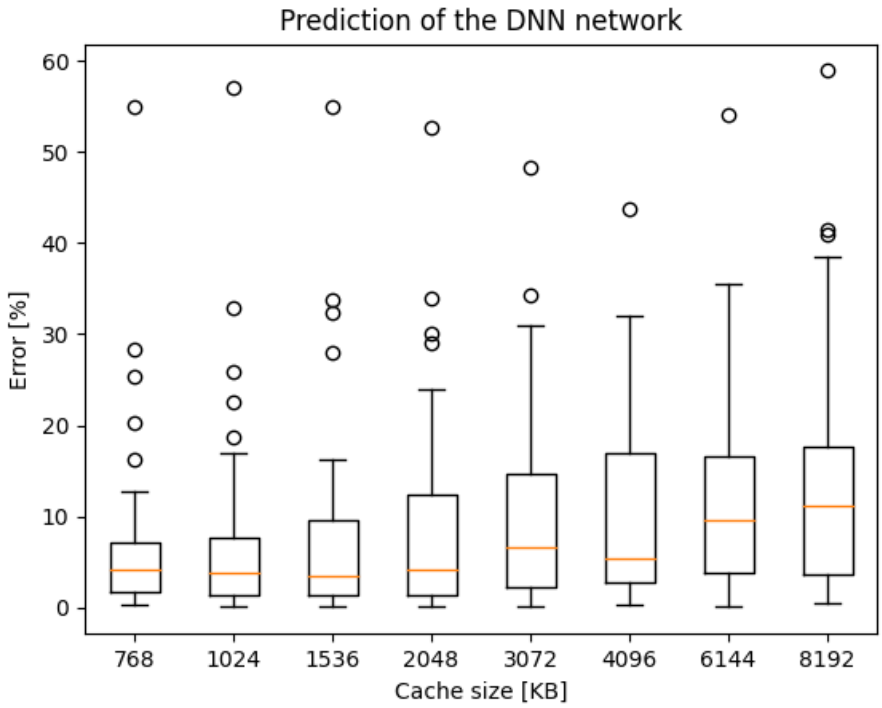
σύνολο εκπαίδευσης. Όπως βλέπουμε στο σχήμα 4.3, τα απόλυτα σφάλματα πρόβλεψης για τις χαμηλότερες κρυφές μνήμες είναι πολύ καλά. Οι διάμεσοι είναι χαμηλοί και σταθεροί. Κυρίως υπάρχει μόνο μία ακραία τιμή για αυτές τις κρυφές μνήμες, και αυτή είναι η πρόβλεψη για το σημείο αναφοράς 416.gamess- ο λόγος αστοχίας του προβλέπεται πάντα υψηλότερος και θα είναι και στο μέλλον, για καλό λόγο, η πρόβλεψη αστοχίας. Αυτό το σημείο αναφοράς αποτελείται από ένα ίχνος, το οποίο έχει μόνο 17κ προσπελάσεις στην LLC, πράγμα που σημαίνει ότι το δίκτυο διαβάζει ένα παράθυρο αποστάσεων επαναχρησιμοποίησης με 17κ προσπελάσεις και προσπαθεί να προβλέψει το μισς ρατιο του. Η πλειονότητα των παραθύρων επαναχρησιμοποιούμενων αποστάσεων στα οποία εκπαιδεύεται το δίκτυό μας αποτελείται από 512κ προσπελάσεις. Έτσι, το δίκτυο βλέπει τον χαμηλό αριθμό αποστάσεων επαναχρησιμοποίησης και υποθέτει πολλές προσπελάσεις που συμβαίνουν μόνο μία φορά και επομένως δεν έχουν αποστάσεις επαναχρησιμοποίησης. Αυτά έχουν ως αποτέλεσμα αστοχίες (σολδ μισσες) και επομένως το δίκτυο υπολογίζει υψηλότερο ποσοστό αστοχίας. Μπορούμε με ασφάλεια να αγνοήσουμε αυτή την έξοδο, δεδομένου ότι, με μια προσομοίωση 30 φορές μεγαλύτερης διάρκειας, η έξοδος πιθανότατα θα πλησίαζε περισσότερο την πραγματική τιμή.

Όσον αφορά τις μεγαλύτερες LLC, οι προβλέψεις χειροτερεύουν σταδιακά. Η ακρίβεια του δικτύου για αυτές τις κρυφές μνήμες είναι σημαντικά χειρότερη από εκείνη του δικτύου LSTM, όπως υποδηλώνουν οι γεωμετρικοί μέσοι όροι τους 4.2. Οι γεωμετρικοί μέσοι των σφαλμάτων πρόβλεψης είναι κοντά σε αυτά που προβλέπει το LSTM για μικρότερα μεγέθη κρυφής μνήμης και χειροτερεύουν δραστικά για μεγαλύτερα μεγέθη κρυφής μνήμης.

Train-set cache sizes [MB]	**Cache size [KB]**							
	768	**1024**	**1536**	**2048**	**3072**	**4096**	**6144**	**8192**
1, 2, 8	3.0%	3.1%	3.8%	5.0%	5.8%	6.2%	6.5%	8.5%
1, 2, 4, 8	3.8%	3.6%	3.1%	4.0%	5.1%	6.0%	6.9%	8.1%
0.75, 1, 2, 4, 6, 8	2.5%	3.4%	4.1%	3.5%	4.2%	5.3%	7.2%	9.7%
LSTM Network	-	3.6%	-	3.9%	-	5.3%	-	6.4%

Πίνακας 4.2: *Γεωμετρικός μέσος όρος των απόλυτων σφαλμάτων πρόβλεψης του δικτύου DNN, εκπαιδευμένο σε διαφορετικά μεγέθη κρυφής μνήμης. Μαζί με τον γεωμετρικό μέσο όρο των απόλυτων σφαλμάτων πρόβλεψης του δικτύου LSTM, για σύγκριση.*

Στη συνέχεια, προσθέσαμε τα δεδομένα της κρυφής μνήμης μεγέθους 4MB στο σύνολο εκπαίδευσης. Τώρα έχουμε τα δεδομένα για τις LLC μεγέθους 1, 2, 4 και 8 MB στο σύνολο εκπαίδευσης. Αυτό σημαίνει ότι το δίκτυο εκπαιδεύεται στα ίδια ακριβώς δεδομένα με τα προηγούμενα δίκτυα που παρουσιάσαμε.

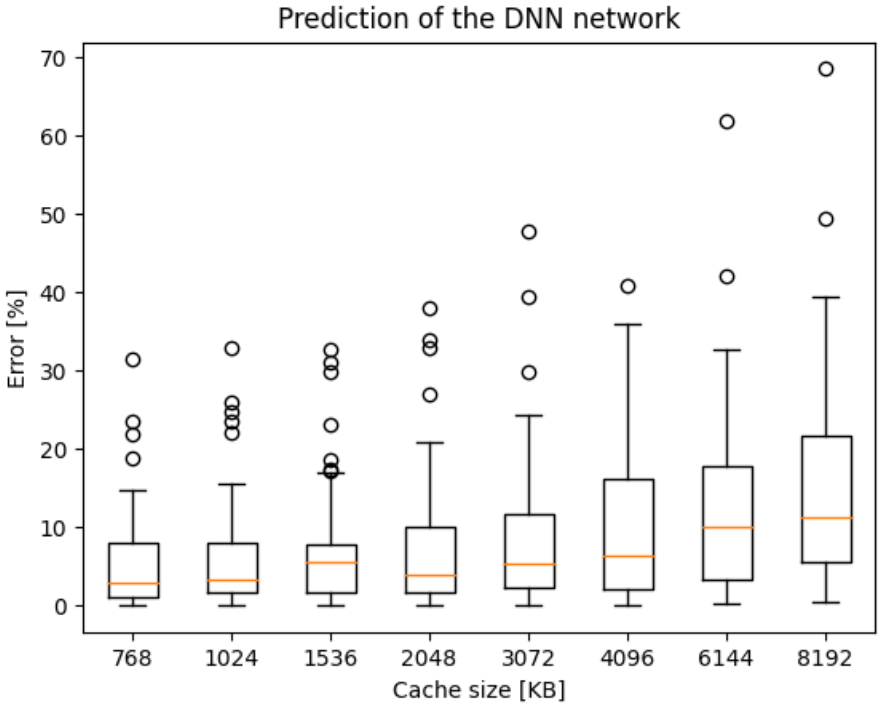


Σχήμα 4.4: *Απόλυτα σφάλματα πρόβλεψης του δικτύου DNN, εκπαιδευμένα σε μεγέθη κρυφής μνήμης 1MB, 2MB, 4MB και 8MB.*

Είναι ενδιαφέρον πώς αυτή η προσθήκη επηρέασε πολύ λίγο την πρόβλεψη της κρυφής μνήμης μεγέθους 4MB, στο σχήμα 4.4 ή στον πίνακα 4.2. Προφανώς, το δίκτυο μπορούσε ήδη να προβλέψει τα ποσοστά αστοχίας για το μέγεθος της κρυφής μνήμης 4MB αρκετά καλά όταν εκπαιδεύτηκε σε τρία μεγέθη κρυφής μνήμης, επομένως, η προσθήκη του το άλλαξε πολύ λίγο.

Αυτή η προσθήκη βοήθησε το δίκτυό μας να κατανοήσει καλύτερα ορισμένες από τις μικρότερες κρυφές μνήμες. Όταν εκπαιδεύσαμε για πρώτη φορά ένα δίκτυο, το εκπαιδεύσαμε μόνο για το μέγεθος της κρυφής μνήμης 2MB. Η γνώση μόνο της κρυφής μνήμης 2 MB μας έδωσε μια πρόβλεψη με γεωμετρικό μέσο όρο απόλυτων σφαλμάτων πρόβλεψης 4,7% για τη στήλη 2MB. Είναι ενδιαφέρον πώς ο γεωμετρικός μέσος όρος της κρυφής μνήμης μεγέθους 2MB (γραμμή 2 του πίνακα 4.2) είναι χαμηλότερος από αυτόν που προβλέφθηκε με μόνο αυτή την κρυφή μνήμη στο σύνολο εκπαίδευσης. Αυτό σημαίνει ότι το δίκτυο απέκτησε κάποια εικόνα για τα ποσοστά αστοχίας της κρυφής μνήμης μεγέθους 2MB από την προσθήκη της κρυφής μνήμης μεγέθους 4MB στο σύνολο εκπαίδευσης.

Τέλος, εκπαιδεύσαμε το δίκτυο σε έξι μεγέθη κρυφής μνήμης: 0,75, 1, 2, 4, 6 και 8 MB μεγέθους LLC. Αυτό γίνεται απλώς για να δούμε πώς θα συμπεριφερθεί το δίκτυο με μια τέτοια προσθήκη μεγεθών κρυφής μνήμης.



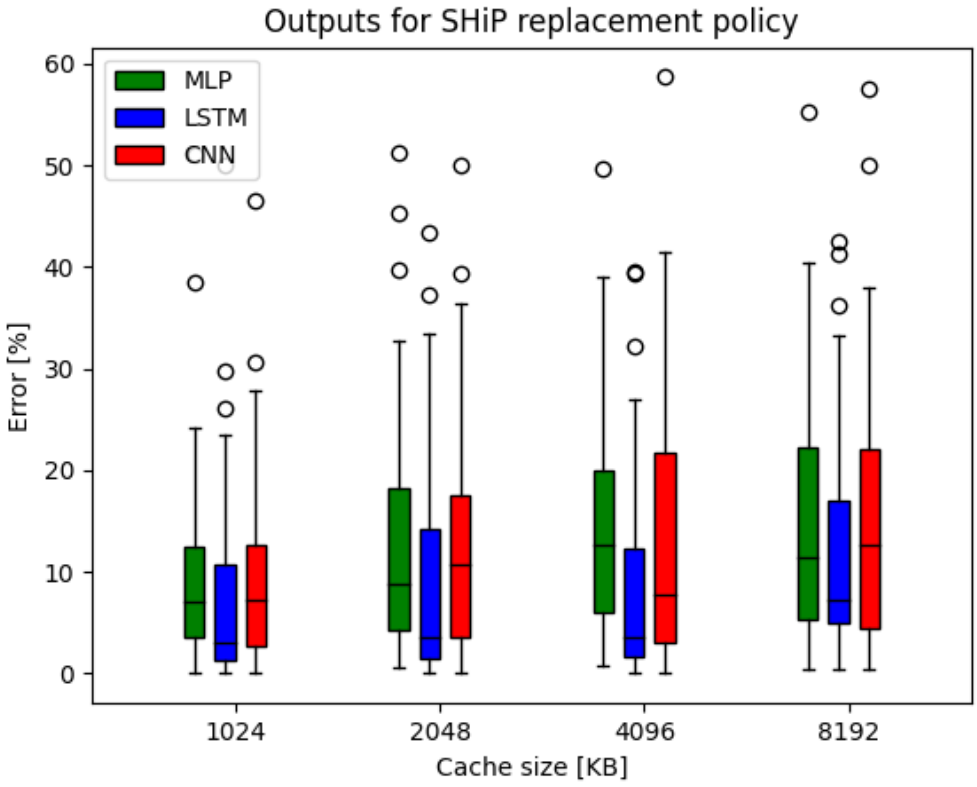
Σχήμα 4.5: *Απόλυτα σφάλματα πρόβλεψης του δικτύου DNN, εκπαιδευμένα σε μεγέθη κρυφής μνήμης 0.75, 1, 2, 4, 6 και 8 MB.*

Οι προβλέψεις αυτού του συνόλου εκπαίδευσης στο σχήμα 4.5 φαίνονται πολύ παρόμοιες με τα σφάλματα των προηγούμενων προβλέψεων. Τα μεγέθη και τα σχήματα των κουτιών στο θηκόγραμμα είναι αρκετά παρόμοια με εκείνο που εκπαιδεύτηκε με 4 μεγέθη κρυφής μνήμης. Η μόνη διαφορά είναι ότι οι προβλέψεις του 416.gamess έγιναν λίγο καλύτερες για μικρότερα μεγέθη κρυφής μνήμης, αλλά όχι αρκετά.

Στην τρίτη γραμμή του πίνακα 4.2 βλέπουμε ότι η πρόβλεψη της κρυφής μνήμης 768 KB έγινε ακόμη καλύτερη, καθώς και η πρόβλεψη για την κρυφή μνήμη μεγέθους 2MB. Τώρα, ο γεωμετρικός μέσος όρος των απόλυτων σφαλμάτων πρόβλεψης για την κρυφή μνήμη μεγέθους 2MB είναι χαμηλότερος από την πρόβλεψη του δικτύου LSTM που αναφέραμε στην αρχή. Αυτό σημαίνει ότι πιθανότατα για οποιοδήποτε μέγεθος κρυφής μνήμης μικρότερο από 4MB αυτό το δίκτυο DNN θα έχει παρόμοια αποτελέσματα με το δίκτυο LSTM. Είναι επίσης ενδιαφέρον ότι ο γεωμετρικός μέσος όρος των απόλυτων σφαλμάτων πρόβλεψης για τις κρυφές μνήμες μεγέθους 6 και 8 MB έχει αυξηθεί. Το δίκτυο εστιάζει περισσότερο στην ακριβή πρόβλεψη των μικρών μεγεθών κρυφής μνήμης και προφανώς δεν μπορεί να κατανοήσει με την ίδια ακρίβεια τα μεγαλύτερα.

4.1.3 Πρόβλεψη πολιτικής αντικατάστασης SHiP

Για τις άλλες τρεις πολιτικές αντικατάστασης, έχουμε μόνο τα σφάλματα πρόβλεψης από τα δίκτυα MLP, CNN, και LSTM. Η πρόβλεψη του StatCache έχει σχεδιαστεί για να περιγράφει τη συμπεριφορά των κρυφών μνήμης LRU και, επομένως, δεν μπορεί να εφαρμοστεί σε άλλες πολιτικές αντικατάστασης κρυφής μνήμης. Το δίκτυο DNN έχει εκπαιδευτεί και δοκιμαστεί μόνο για την πολιτική αντικατάστασης LRU- για άλλες πολιτικές αντικατάστασης, δεν έχουμε αρκετά δεδομένα για αρκετά μεγέθη κρυφής μνήμης ώστε να αξιολογήσουμε με ακρίβεια την απόδοσή του σε διάφορα μεγέθη κρυφής μνήμης.



Σχήμα 4.6: *Σφάλματα πρόβλεψης των δικτύων MLP, LSTM, CNN για LLC με πολιτική αντικατάστασης SHiP.*

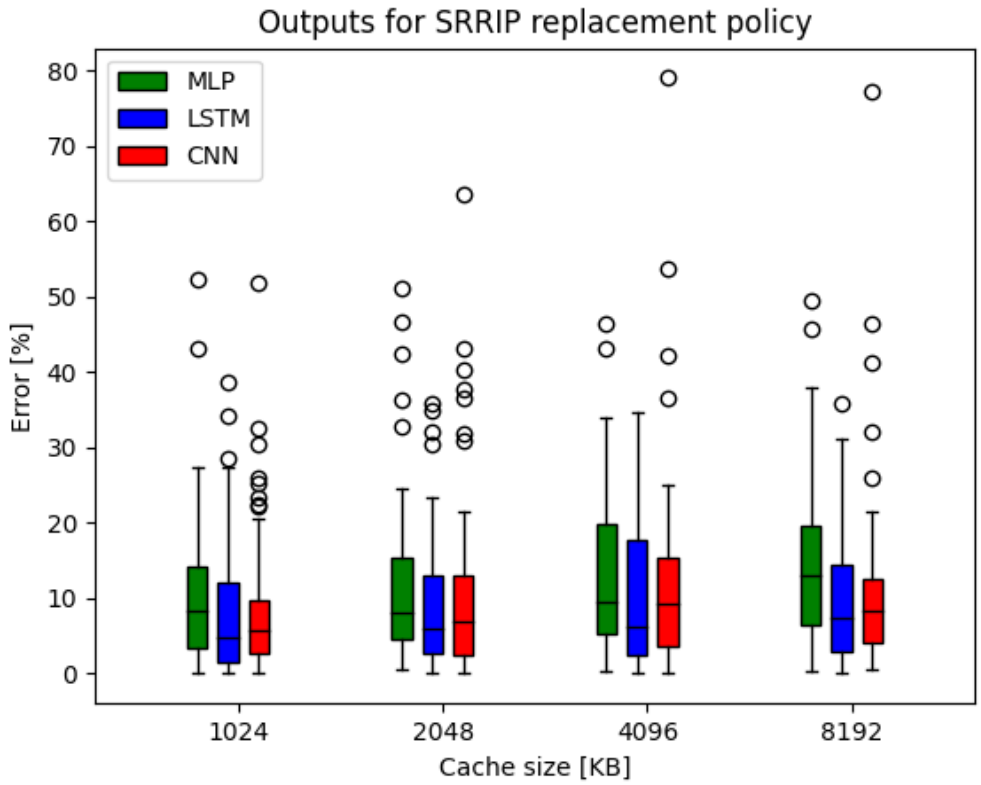
Network	**Cache size [MB]**			
	1	**2**	**4**	**8**
MLP	5.6%	7.7%	9.8%	8.6%
LSTM	2.7%	4.1%	3.7%	7.6%
CNN	5.3%	7.3%	5.8%	9.4%

Πίνακας 4.3: *Γεωμετρικός μέσος όρος των σφαλμάτων πρόβλεψης για τις κρυφές μνήμες με πολιτική αντικατάστασης SHiP.*

Η καλύτερη πρόβλεψη όταν εξετάζουμε τα σφάλματα πρόβλεψης, σχήμα 4.6, ή τους γεωμετρικούς μέσους όρους τους, πίνακας 4.3, είναι σαφώς η πρόβλεψη LSTM. Όλοι οι

μέσοι όροι είναι πιο κοντά στο μηδέν από οποιονδήποτε άλλο και τα θηκογράμματα φαίνονται καλύτερα για όλα τα μεγέθη της κρυφής μνήμης. Οι γεωμετρικοί μέσοι του δικτύου είναι τουλάχιστον 2% καλύτεροι από τους γεωμετρικούς μέσους του μικρότερου από τα άλλα δύο δίκτυα. Επίσης, δεν αποκλίνουν περισσότερο από 1% από τους γεωμετρικούς μέσους των σφαλμάτων πρόβλεψης του LSTM για την πολιτική αντικατάστασης LRU. Αυτό σημαίνει ότι έχουν το ίδιο εύρος με αυτά που προβλέπονται για LRU και δείχνουν τη σταθερότητα αυτού του δικτύου παρά την αλλαγή της πολιτικής αντικατάστασης. Τα σφάλματα πρόβλεψης από το δίκτυο CNN για αυτή την πολιτική αντικατάστασης είναι πολύ παρόμοια με τα σφάλματα πρόβλεψης του δικτύου MLP, γεγονός που δείχνει ότι το δίκτυο CNN παρουσιάζει κάποια αδυναμία για αυτή την πολιτική αντικατάστασης. Αυτό εξακολουθεί να εγείρει το ερώτημα, τι κάνει το δίκτυο LSTM να προβλέπει τόσο πολύ καλύτερα από τα άλλα δύο σε αυτή την πολιτική αντικατάστασης.

4.1.4 Πρόβλεψη πολιτικής αντικατάστασης SRRIP



Σχήμα 4.7: *Σφάλματα πρόβλεψης των δικτύων MLP, LSTM, CNN για LLC με πολιτική αντικατάστασης SRRIP.*

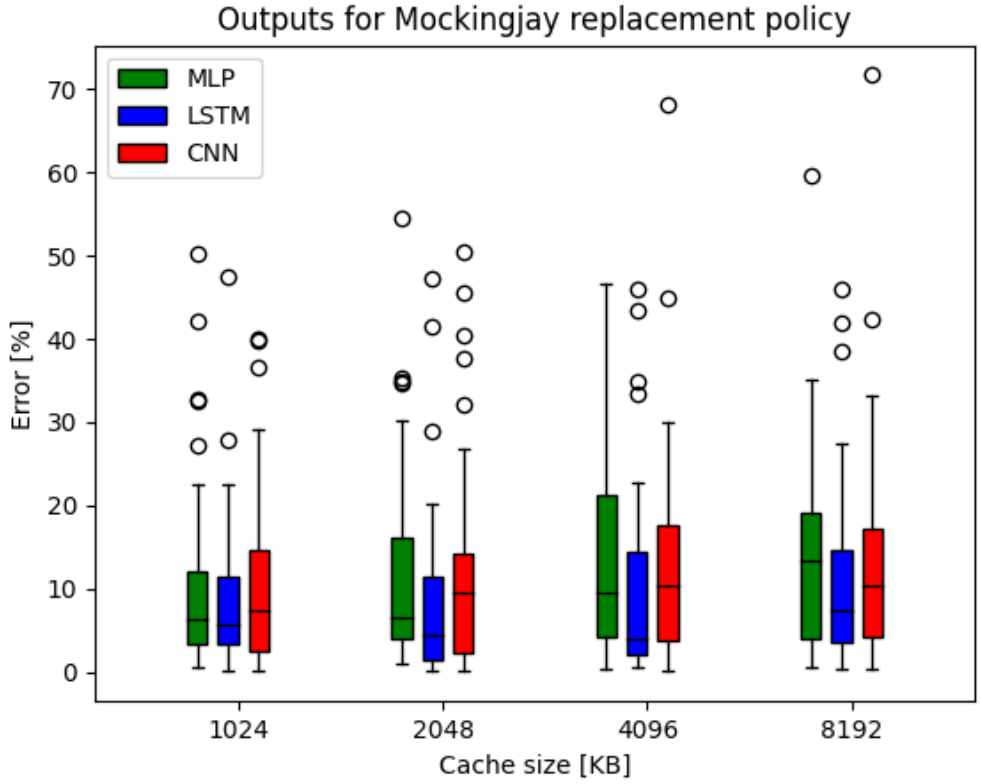
Και πάλι, οι προβλέψεις του δικτύου LSTM φαίνεται να είναι οι ισχυρότερες. Τα σφάλματα πρόβλεψης από το δίκτυο CNN είναι καλύτερα από αυτά που είχαμε για το SHiP και μοιάζουν με τα σφάλματα πρόβλεψης του δικτύου LSTM. Έχουν μικρότερα πλαίσια στο θηκόγραμμα, που σημαίνει ότι το σφάλμα πρόβλεψης είναι αρκετά σταθερό. Παρόλα αυτά, τα σφάλματα πρόβλεψης του δικτύου LSTM είναι καλύτερα όσον αφορά τους μέσους και

Network	**Cache size [MB]**			
	1	**2**	**4**	**8**
MLP	5.7%	8.0%	8.6%	9.3%
LSTM	3.1%	4.6%	5.5%	5.7%
CNN	4.7%	5.2%	6.4%	7.4%

Πίνακας 4.4: *Γεωμετρικός μέσος όρος των σφαλμάτων πρόβλεψης για τις κρυφές μνήμες με πολιτική αντικατάστασης SRRIP.*

τους γεωμετρικούς μέσους. Αυτό που αξίζει επίσης να αναφερθεί είναι ότι πρόκειται για τον χαμηλότερο γεωμετρικό μέσο των σφαλμάτων πρόβλεψης που είχαμε για την κρυφή μνήμη μεγέθους 8MB.

4.1.5 Πρόβλεψη πολιτικής αντικατάστασης Mockingjay



Σχήμα 4.8: *Σφάλματα πρόβλεψης των δικτύων MLP, LSTM, CNN για LLC με πολιτική αντικατάστασης Mockingjay.*

Network	**Cache size [MB]**			
	1	**2**	**4**	**8**
MLP	6.2%	7.8%	9.0%	8.8%
LSTM	4.9%	3.6%	5.3%	6.2%
CNN	5.3%	6.2%	7.6%	8.3%

Πίνακας 4.5: *Γεωμετρικός μέσος όρος των σφαλμάτων πρόβλεψης για τις κρυφές μνήμες με πολιτική αντικατάστασης Mockingjay.*

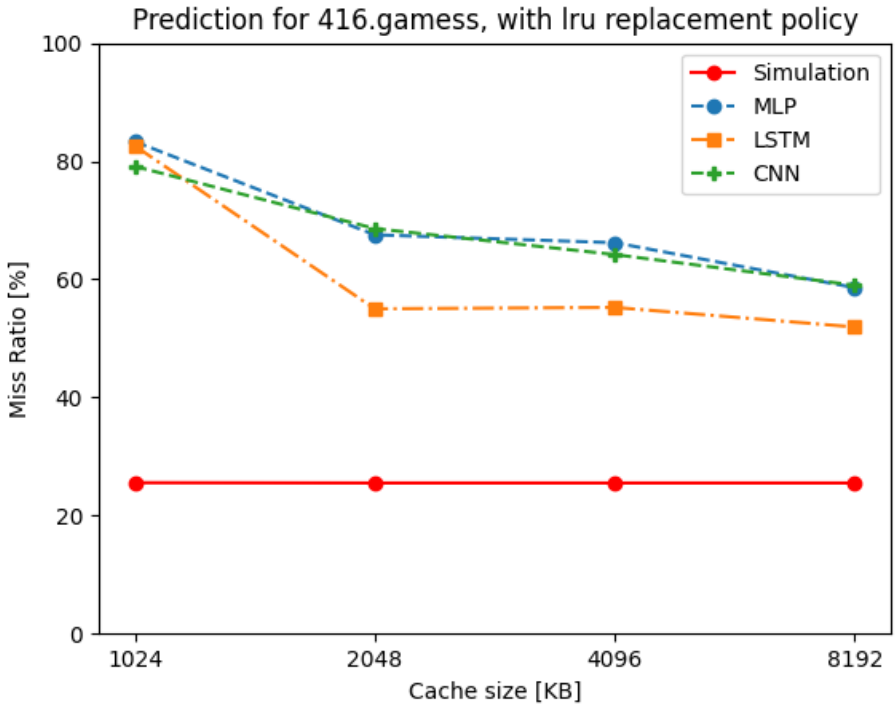
Τέλος, υπάρχει η πολιτική αντικατάστασης του Mockingjay. Όπως και προηγουμένως, το δίκτυο MLP είναι σαφώς χειρότερο από τα άλλα δύο δίκτυα στην πρόβλεψη των κρυφών μνήμης με αυτή την πολιτική αντικατάστασης. Το δίκτυο LSTM υπερέχει σημαντικά έναντι των άλλων δικτύων. Κάθε πλαίσιο στο θηκόγραμμα των σφαλμάτων πρόβλεψης από το δίκτυο LSTM είναι μικρότερο, χαμηλότερο και έχει μικρότερο μέσο όρο από τα άλλα.

Το Mockingjay είναι μια πολιτική αντικατάστασης κρυφής μνήμης που βασίζεται στις αποστάσεις επαναχρησιμοποίησης. Τις παρακολουθεί, υπολογίζει την πιθανότητα επαναχρησιμοποίησης και στη συνέχεια αποφασίζει αν ένα αντικείμενο θα παραμείνει στην κρυφή μνήμη. Αυτό θα μας οδηγούσε στο συμπέρασμα ότι μια διαδικασία όπως ένα διπλό στρώμα

LSTM βοηθά το δίκτυο να κατανοήσει καλύτερα τις αποστάσεις επαναχρησιμοποίησης και πώς θα συμπεριφερθεί η κρυφή μνήμη με αυτή την πολιτική αντικατάστασης. Αλλά το δίκτυο LSTM δεν προβλέπει το Mockingjay σημαντικά καλύτερα από οποιαδήποτε άλλη πολιτική αντικατάστασης. Αυτό μας οδηγεί στο συμπέρασμα ότι υπάρχουν ακόμη περιθώρια βελτίωσης όσον αφορά αυτό το δίκτυο.

4.1.6 Ανάλυση συγκεκριμένων εφαρμογών

Αυτή η ενότητα της διατριβής έχει ως στόχο να αναλύσει και να συγκρίνει τα αποτελέσματα των προαναφερθέντων δικτύων. Θα συγκρίνουμε και θα συζητήσουμε τα πλεονεκτήματα και τις αδυναμίες των δικτύων μας, δίνοντας έμφαση στα σχετικά πλεονεκτήματά τους σε πραγματικές εφαρμογές. Ας ξεκινήσουμε τη σύγκριση των αποτελεσμάτων αξιολογώντας τις προβλέψεις των δικτύων μας για ορισμένα συγκεκριμένα αρχεία αναφοράς.



Σχήμα 4.9: *Προβλέψεις δικτύου για 416.gamess με πολιτική αντικατάστασης LRU.*

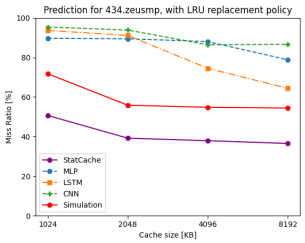
Πρώτα απ' όλα, ας ασχοληθούμε με το σημείο αναφοράς με τις περισσότερες λανθασμένες προβλέψεις στο σύνολο δεδομένων μας. Το σημείο αναφοράς 416.games αποτελείται από ένα ίχνος με 17κ προσπελάσεις μέσα σε αυτό. Η πλειονότητα των παραθύρων στα οποία εκπαιδεύεται το δίκτυό μας αποτελείται από 512κ προσπελάσεις. Όταν έρχεται αντιμέτωπο με ένα παράθυρο αυτού του μεγέθους, το δίκτυό μας υποθέτει ότι οι περισσότερες προσπελάσεις που συμβαίνουν είναι cold misses και, ως εκ τούτου, δεν έχουν αποστάσεις επαναχρησιμοποίησης. Ως εκ τούτου, η πρόβλεψη είναι πολύ υψηλότερη από την πραγματική τιμή. Όπως βλέπουμε παραπάνω, στο σχήμα 4.9, τα τρία πρώτα δίκτυα προβλέπουν πολύ υψηλότερο ποσοστό αστοχίας από το πραγματικό. Αυτός ο τύπος αναφοράς μπορεί να αγνοηθεί με ασφάλεια, καθώς, με μια μεγαλύτερη προσομοίωση, το αποτέλεσμα θα είναι πιο ακριβές.

Περνάμε σε μερικά από τα χειρότερα προβλεπόμενα σημεία αναφοράς από όλα τα δίκτυά μας. Στο σχήμα 4.10 παρουσιάζουμε τρία από τα σημεία αναφοράς για τα οποία έχουμε τις μεγαλύτερες αποκλίσεις μεταξύ της προβλεπόμενης και της προσομοιωμένης τιμής. Αυτά τα τρία σημεία αναφοράς είναι τα 401.bzip2, 434.zeusmp και 456.hmmer. Όλα τους είναι ιδιαίτερα εντατικά προγράμματα με εκατομμύρια προσπελάσεις το καθένα. Παρουσιάζουμε τις εξόδους των δικτύων μας για τις τέσσερις πολιτικές αντικατάστασης. Για την LRU, δεδομένου ότι διαχωρίσαμε τις προβλέψεις των StatCache, MLP, CNN και LSTM από τα δίκτυα DNN για να διατηρήσουμε τη σαφήνεια. Οι προβλέψεις των DNN (διαγράμματα d, e και f) επισημαίνονται ως DNN-i, όπου i είναι ο αριθμός των μεγεθών της κρυφής μνήμης εκπαίδευσης του δικτύου. Είναι σαφές ότι όλα τα δίκτυα DNN-1 είναι τυχαία και αυτό οφείλεται στο γεγονός ότι το δίκτυο δεν κατανοεί την παράμετρο μεγέθους, καθώς έχει εκπαιδευτεί μόνο σε μία παράμετρο μεγέθους.

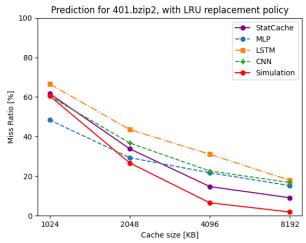
Το 434.zeusmp είναι ένα πρόγραμμα που προσομοιώνει αστροφυσικά φαινόμενα με βάση τον κώδικα υπολογιστικής ρευστοδυναμικής ZEUS-MP. Είναι σαφές ότι η πρόβλεψη από το StatCache για αυτό το σημείο αναφοράς είναι η μόνη που είναι χαμηλότερη από τις πραγματικές προσομοιωμένες αναλογίες αστοχίας.Όλες οι προβλέψεις από τα δίκτυα που προτείναμε φαίνεται να συμφωνούν ότι ο λόγος αστοχίας θα πρέπει να είναι σημαντικά υψηλότερος από τον προσομοιωμένο.Παρόλο που αυτό ισχύει για τα περισσότερα δίκτυα, το LSTM φαίνεται να έχει κάνει μια γενναία προσπάθεια να προβλέψει με ακρίβεια την αναλογία αστοχίας για την πολιτική αντικατάστασης Mockingjay (διάγραμμα m).

Το 401.bzip2 είναι ένας αλγόριθμος συμπίεσης που συμπιέζει μια σειρά αρχείων καθ' όλη τη διάρκεια της εκτέλεσής του. Η πρόβλεψη της StatCache για αυτό το αρχείο αναφοράς είναι πολύ ακριβής, ξεπερνώντας τα δίκτυά μας κατά πολύ.Η λανθασμένη πρόβλεψη των δικτύων μας για αυτό το αρχείο αναφοράς δεν είναι τόσο κακή όσο για τα άλλα δύο αρχεία αναφοράς.Η πρόβλεψη των ποσοστών αστοχίας είναι πάντα ακριβής για το μικρότερο μέγεθος κρυφής μνήμης και στη συνέχεια, καθώς τα μεγέθη κρυφής μνήμης γίνονται μεγαλύτερα, αποκλίνει.

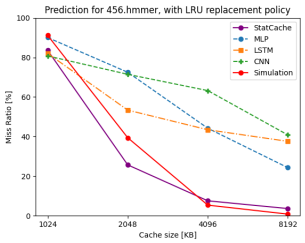
Τέλος, το αρχείο 456.hmmer είναι ένας υπολογισμός των προφίλ κρυφών μοντέλων Μάρκοβ για να γίνει ευαίσθητη αναζήτηση σε βάσεις δεδομένων χρησιμοποιώντας στατιστικές περιγραφές της συναίνεσης μιας οικογένειας ακολουθιών. Και πάλι, η πρόβλεψη της Statcache είναι πολύ καλύτερη για αυτό το σημείο αναφοράς. Η απότομη πτώση της αναλογίας αστοχίας για μεγάλα μεγέθη κρυφής μνήμης φαίνεται να μπερδεύει τα δίκτυά μας, αφού προβλέπουν πάντα υψηλότερα για τα μεγαλύτερα μεγέθη κρυφής μνήμης, παρόλο που για τα μικρότερα τα βρίσκουν σωστά. Είναι ενδιαφέρον πώς το δίκτυο DNN (διάγραμμα f) δημιουργεί φαινομενικά το ίδιο σχήμα πρόβλεψης όταν εκπαιδεύεται με 1, 2 ή 3 μεγέθη κρυφής μνήμης, από το τέταρτο και πάνω. Από την άλλη πλευρά, φαίνεται να κατανοεί καλύτερα το σχήμα των συγκριτικών τιμών και να κινείται για να το ξεπεράσει. Το LSTM φαίνεται επίσης να έχει το σωστό σχήμα σε σύγκριση με τα άλλα δύο δίκτυα, αλλά δεν φαίνεται να καταλαβαίνει ακριβώς την απότομη καμπύλη.



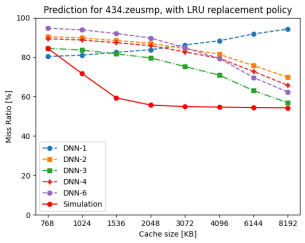
(α΄) 434.zeusmp, LRU



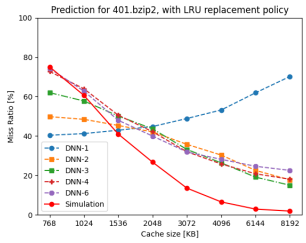
(β΄) 401.bzip2, LRU



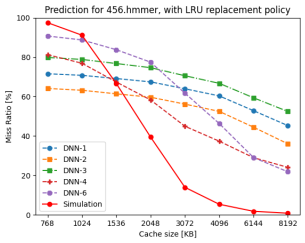
(γ΄) 456.hmmer, LRU



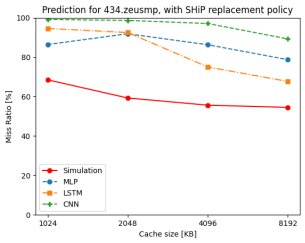
(δ΄) 434.zeusmp, LRU



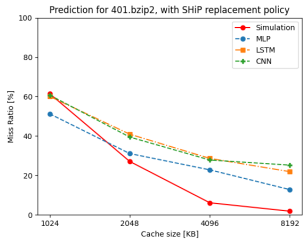
(ε΄) 401.bzip2, LRU



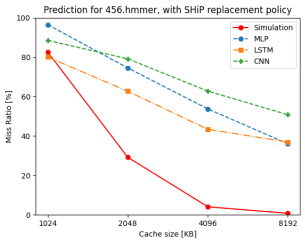
(ϛ΄) 456.hmmer, LRU



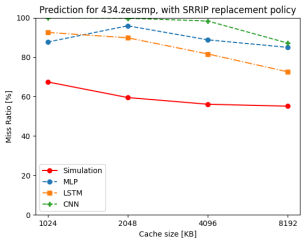
(ζ΄) 434.zeusmp, SHiP



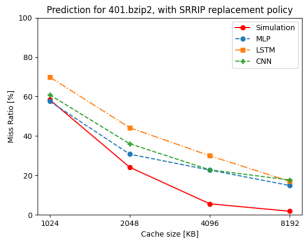
(η΄) 401.bzip2, SHiP



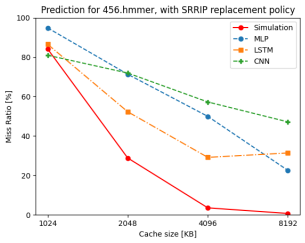
(θ΄) 456.hmmer, SHiP



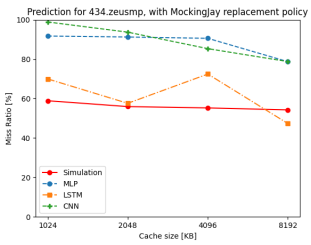
(ι΄) 434.zeusmp, SRRIP



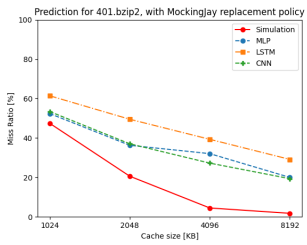
(ια΄) 401.bzip2, SRRIP



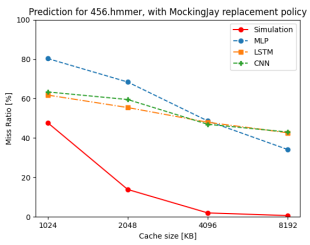
(ιβ΄) 456.hmmer, SRRIP



(ιγ΄) 434.zeusmp, Mockingjay

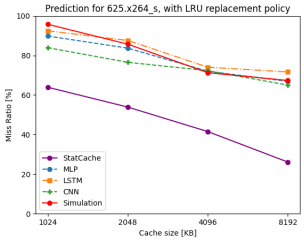


(ιδ΄) 401.bzip2, Mockingjay

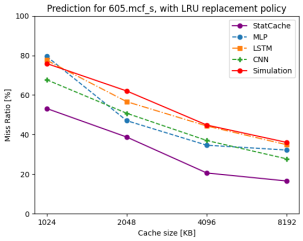


(ιε΄) 456.hmmer, Mockingjay

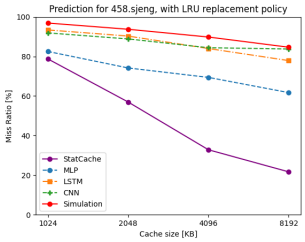
Σχήμα 4.10: *Μερικές από τις χειρότερες προβλέψεις αρχείων αναφοράς.*



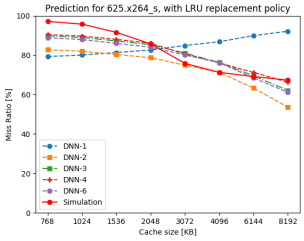
(α') 625.x264_s, LRU



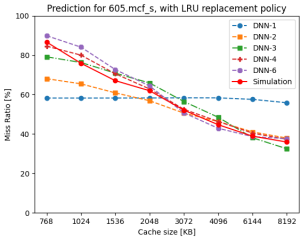
(β') 605.mcf_s, LRU



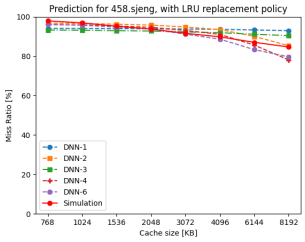
(γ') 458.sjeng, LRU



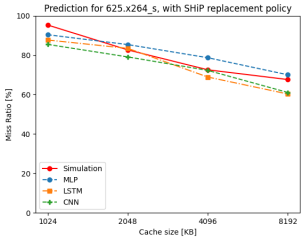
(δ') 625.x264_s, LRU



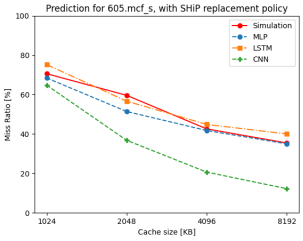
(ε') 605.mcf_s, LRU



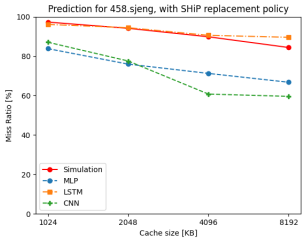
(ς') 458.sjeng, LRU



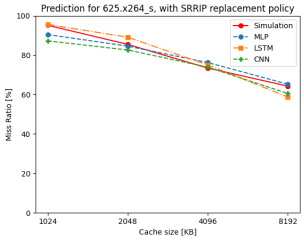
(ζ) 625.x264_s, SHiP



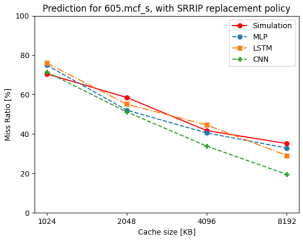
(η') 605.mcf_s, SHiP



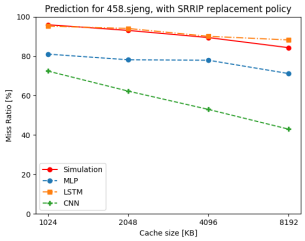
(θ') 458.sjeng, SHiP



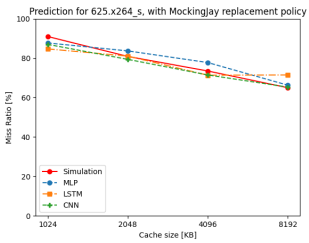
(ι') 625.x264_s, SRRIP



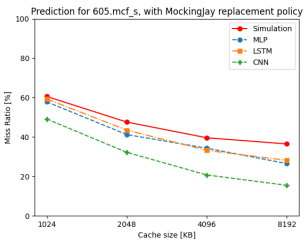
(ια') 605.mcf_s, SRRIP



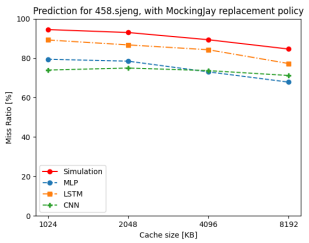
(ιβ') 458.sjeng, SRRIP



(ιγ') 625.x264_s, Mockingjay



(ιδ') 605.mcf_s, Mockingjay



(ιε') 458.sjeng, Mockingjay

Σχήμα 4.11: *Μερικές από τις καλύτερες προβλέψεις αρχείων αναφοράς.*

Ορισμένα από τα καλύτερα προβλεπόμενα αρχεία παρουσιάζονται στο σχήμα 4.11. Αυτή τη φορά, οι προβλέψεις από τα δίκτυα είναι πολύ κοντά στα ποσοστά αστοχίας της κρυφής μνήμης της προσομοίωσης.

Τα τρία αρχεία αναφορά που παρουσιάζουμε έχουν μερικές από τις καλύτερες προβλέψεις από τα δίκτυά μας .Προφανώς δεν είναι όλα τα αρχεία αναφοράς με καλές προβλέψεις, απλώς μερικά ενδεικτικά. Είναι ενδιαφέρον πώς, για σχεδόν όλες αυτές τις προβλέψεις, η καλύτερη πρόβλεψη είναι αυτή του δικτύου LSTM. Κατανοεί αυτά τα προβλήματα σχεδόν τέλεια, έχοντας ελάχιστες αποκλίσεις από τις πραγματικές τιμές. Δείχνει πραγματικά πώς η καλύτερη κατανόηση του προβλήματος εφαρμόζεται στην πρόβλεψη.

Το CNN και το MLP συνολικά φαίνεται να έχουν μικρές αποκλίσεις από τις τιμές της προσομοίωσης, μεγαλύτερες από αυτές του δικτύου LSTM. Το δίκτυο DNN φαίνεται να κατανοεί επίσης το πρόβλημα, και με περισσότερα δεδομένα, οι προβλέψεις γίνονται επίσης καλύτερες. Και πάλι, έχουμε το μη εκπαιδευμένο διάγραμμα DNN-1 που εμφανώς δεν καταλαβαίνει την παράμετρο του μεγέθους.

Τέλος, το StatCache υπερτερεί σημαντικά σε αυτές τις προβλέψεις. Οι προβλέψεις του StatCache δεν είναι κακές για όλα αυτά τα σημεία αναφοράς. Μπορεί να είναι κακές για το σημείο αναφοράς 458.sjeng, αλλά για τα υπόλοιπα, οι προβλέψεις των δικτύων μας είναι πολύ πιο ακριβείς.

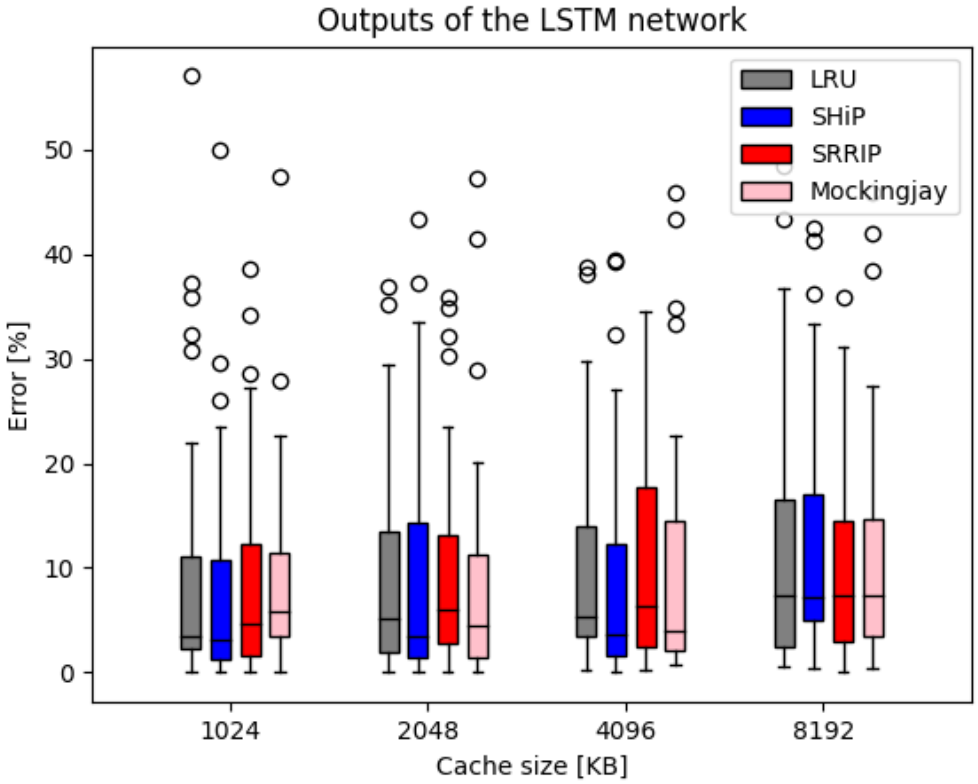
4.1.7 Συμπεράσματα

Συμπερασματικά, το δίκτυο LSTM προβλέπει σταθερά τα πιο ακριβή αποτελέσματα από τα δίκτυα που προτείναμε. Όπως βλέπουμε στο σχήμα 4.12 τα απόλυτα σφάλματα πρόβλεψής του είναι εξίσου καλά για οποιαδήποτε πολιτική αντικατάστασης. Τα αποτελέσματά του έδειξαν ότι μπορεί να προβλέψει με συνέπεια οποιοδήποτε μέγεθος κρυφής μνήμης στο οποίο εκπαιδεύεται με πολύ χαμηλούς γεωμετρικούς μέσους όρους σφαλμάτων πρόβλεψης. Έδειξε επίσης ότι είναι πολύ συνεπές στις προβλέψεις της, καθώς οι γεωμετρικοί μέσοι των σφαλμάτων πρόβλεψης στον πίνακα 4.6 δεν διαφέρουν πολύ στις διαφορετικές πολιτικές αντικατάστασης.

Πολιτική αντικατάστασης	1MB	2MB	4MB	8MB
LRU	3.6%	3.9%	5.3%	6.4%
SHiP	2.7%	4.1%	3.7%	7.6%
SRRIP	3.1%	4.6%	5.5%	5.7%
Mockingjay	4.9%	3.6%	5.3%	6.2%

Πίνακας 4.6: *Γεωμετρικός μέσος όρος των απόλυτων σφαλμάτων πρόβλεψης του δικτύου LSTM*

Μια απλούστερη και λιγότερο δαπανηρή εναλλακτική λύση σε αυτό είναι το δίκτυο CNN, το οποίο έχει τη δυνατότητα να προβλέπει με μεγάλη ακρίβεια και εκπαιδεύεται πολύ πιο γρήγορα, με περίπου 20% του χρόνου εκπαίδευσης του LSTM. Αλλά είναι επίσης ασυνεπές για ορισμένες πολιτικές αντικατάστασης, με παρόμοιες προβλέψεις με το MLP, πράγμα που δεν είναι βέλτιστο. Το δίκτυο DNN παρουσιάζει τις περισσότερες δυνατότητες, καθώς έχει τις καλύτερες προβλέψεις για μικρότερα μεγέθη κρυφής μνήμης στο πλαίσιο της πολιτικής αντικατάστασης LRU, αλλά εξακολουθεί να έχει περιθώρια βελτίωσης στις μεγαλύτερες κρυφές μνήμες. Συνολικά, τα αποτελέσματα αυτά υπογραμμίζουν τις διαφορετικές περιπλοκές του προβλήματος και δείχνουν ότι είναι πράγματι δυνατή η ακριβής πρόβλεψή του.



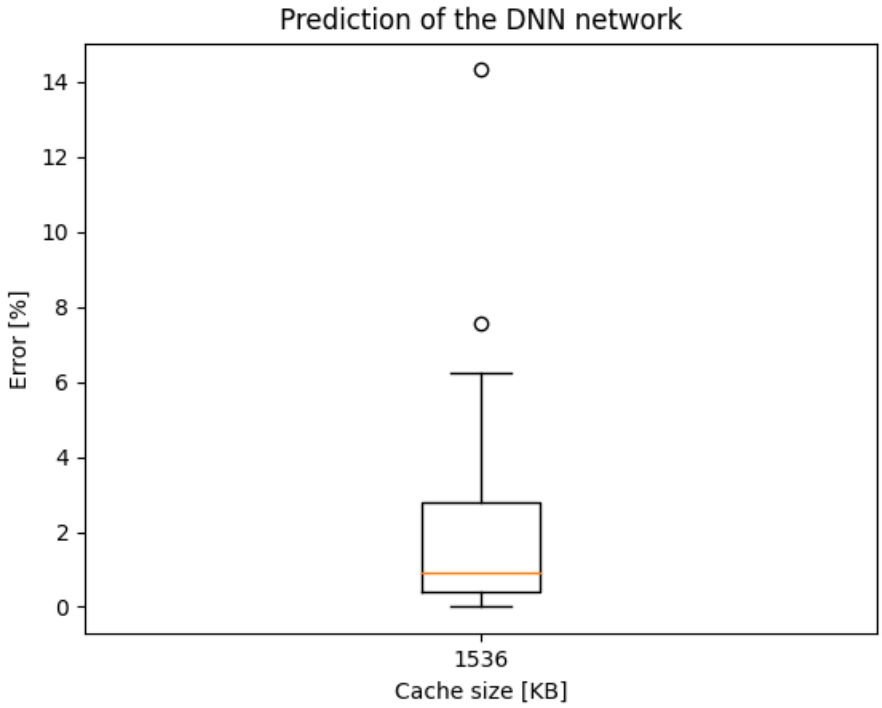
Σχήμα 4.12: *Απόλυτα σφάλματα πρόβλεψης του δικτύου LSTM*

4.2 Πρόβλεψη γνωστού προγράμματος

Υπάρχει και μια άλλη χρήση αυτού του δικτύου. Ας υποθέσουμε ότι έχουμε μερικά μηχανήματα που έχουν μια συγκεκριμένη αρχιτεκτονική. Θέλουμε να προβλέψουμε πώς θα συμπεριφερθεί ένα πρόγραμμα που έχει εκτελεστεί σε αυτές τις μηχανές, όταν εκτελεστεί από μηχανές με την ίδια πολιτική αντικατάστασης και άλλα μεγέθη LLC. Για να μοντελοποιήσουμε αυτή την περίπτωση χρήσης, θα χρησιμοποιήσουμε τις προσομοιώσεις της σουίτας αναφοράς για ορισμένα μεγέθη κρυφής μνήμης και θα μεταφέρουμε τα υπόλοιπα στο σύνολο δοκιμών. Με αυτόν τον τρόπο, θα είναι σαν να έχουμε μηχανές με τα μεγέθη κρυφής μνήμης LLC του συνόλου εκπαίδευσης και πολιτική αντικατάστασης LRU και να προβλέπουμε τα ποσοστά αστοχίας για τα μεγέθη κρυφής μνήμης εντός του συνόλου δοκιμής. Κάθε φορά, ολόκληρο το σύνολο των συγκριτικών δοκιμών διατηρείται στο σύνολο εκπαίδευσης και προσπαθούμε να προβλέψουμε όλες τις συγκριτικές δοκιμές ταυτόχρονα.

Αρχικά, μετακινήσαμε τα δεδομένα για ένα μέγεθος κρυφής μνήμης από το σύνολο εκπαίδευσης στο σύνολο δοκιμής για να δούμε αν και πόσο καλά το δίκτυο μπορεί να το προβλέψει με βάση τα υπόλοιπα αρχεία αναφοράς. Πιο συγκεκριμένα, το σύνολο δοκιμής αποτελείται από τη στήλη για μέγεθος κρυφής μνήμης 3MB και το σύνολο εκπαίδευσης από τα άλλα 7 μεγέθη κρυφής μνήμης [768, 1024, 1536, 2048, 4096, 6044, 8192] KB.

Οι προβλέψεις είναι πολύ καλύτερες από ό,τι θα μπορούσε να κάνει οποιοδήποτε άλλο δίκτυο. Ο γεωμετρικός μέσος όρος των παραπάνω σφαλμάτων είναι 0,9%, όπως φαίνεται



Σχήμα 4.13: *Απόλυτα σφάλματα πρόβλεψης του DNN για 1 μέγεθος κρυφής μνήμης, εκπαιδευμένο με ποσοστά αστοχίας 7 μεγεθών κρυφής μνήμης.*

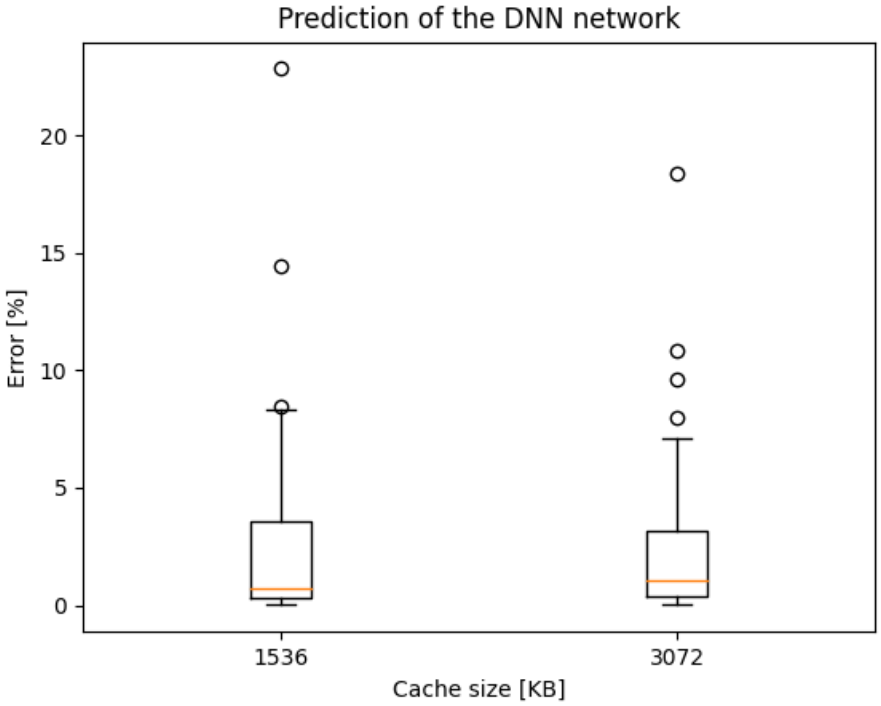
στην τελευταία γραμμή του πίνακα 4.7. Ο γεωμετρικός μέσος όρος των απόλυτων σφαλμάτων πρόβλεψης του LSTM που είχε προβλεφθεί προηγουμένως για το LRU είναι 3,9% και 5,3% για τα δύο γειτονικά μεγέθη κρυφής μνήμης 2MB και 4MB, αντίστοιχα. Αυτό σημαίνει ότι αυτή η πρόβλεψη είναι αναμφισβήτητα πολύ καλύτερη από την καλύτερη πρόβλεψη που είχαμε μέχρι τώρα. Ήταν αναμενόμενο ότι αυτό το αποτέλεσμα θα ήταν καλό, δεδομένου ότι το δίκτυο έχει ένα πολύ μεγάλο σύνολο δεδομένων για να εκπαιδευτεί και μπορεί να κατανοήσει πολύ καλά το πρόβλημα.

Αυτό το αποτέλεσμα είναι μια καλή ένδειξη ότι το δίκτυό μας είναι κατάλληλο για την πρόβλεψη αυτού του είδους προβλήματος. Προκάλεσε επίσης το ερώτημα για το πόσα λίγα μεγέθη κρυφής μνήμης χρειάζονται στο σύνολο εκπαίδευσης για να έχουμε μια ακριβή πρόβλεψη, αφού η προσομοίωση επτά μεγεθών κρυφής μνήμης για την πρόβλεψη του όγδοου δεν είναι ένα ρεαλιστικό πρόβλημα.

Train-set cache sizes [MB]	**Cache sizes [KB]**							
	768	**1024**	**1536**	**2048**	**3072**	**4096**	**6144**	**8192**
1, 2, 4, 8	1.4%	-	0.8%	-	0.9%	-	1.1%	-
0.75, 1, 2, 4, 6, 8	-	-	0.8%	-	0.8%	-	-	-
0.75,1,1.5,2,4,6,8	-	-	-	-	0.9%	-	-	-
LSTM Network	-	3.6%	-	3.9%	-	5.3%	-	6.4%

Πίνακας 4.7: *Γεωμετρικός μέσος όρος των απόλυτων σφαλμάτων πρόβλεψης του δικτύου DNN.*

Το αμέσως επόμενο βήμα για να απαντηθεί αυτό το ερώτημα είναι να αφαιρεθεί άλλο ένα μέγεθος κρυφής μνήμης και να παρατηρηθεί πώς συμπεριφέρεται το σφάλμα των προβλέψεων.



Σχήμα 4.14: *Απόλυτα σφάλματα πρόβλεψης του DNN για 2 μεγέθη κρυφής μνήμης, εκπαιδευμένο με ποσοστά αστοχίας 6 μεγεθών κρυφής μνήμης*

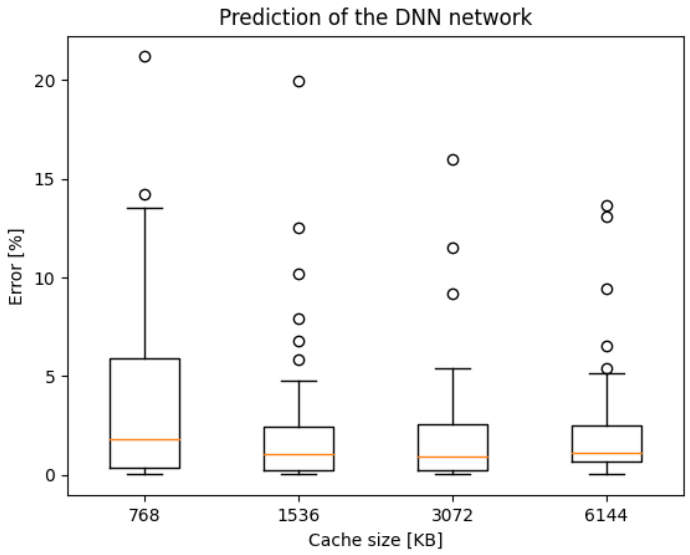
Ο γεωμετρικός μέσος όρος για τις δύο στήλες μεγέθους 1,5MB και 3MB είναι 0,8% και 0,8%, αντίστοιχα. Βλέπουμε ότι αυτοί οι γεωμετρικοί μέσοι είναι εξίσου καλοί με αυτούς που παρατηρήσαμε προηγουμένως.

Είναι σαφώς ορατό ότι, για κάθε μέγεθος κρυφής μνήμης, υπάρχει ένα μακράν χειρότερο προβλεπόμενο σημείο αναφοράς. Αυτή είναι η πρόβλεψη για το αρχείο αναφοράς 416.gamess- το ποσοστό αστοχίας του προβλέπεται πάντα υψηλότερο και θα προβλέπεται και στο μέλλον, για καλό λόγο. Αυτό το αρχείο αναφοράς αποτελείται από ένα ίχνος, το οποίο έχει μόνο 17κ προσπελάσεις στην LLC, πράγμα που σημαίνει ότι το δίκτυο διαβάζει ένα παράθυρο των αποστάσεων επαναχρησιμοποίησης με 17κ προσπελάσεις και προσπαθεί

να προβλέψει το ποσοστό αστοχίας του. Η πλειονότητα των παραθύρων αποστάσεων επαναχρησιμοποίησης στα οποία εκπαιδεύεται το δίκτυό μας αποτελείται από 512κ προσπελάσεις. Έτσι, το δίκτυο βλέπει τον χαμηλό αριθμό αποστάσεων επαναχρησιμοποίησης και υποθέτει πολλές προσπελάσεις που συμβαίνουν μόνο μία φορά και επομένως δεν έχουν αποστάσεις επαναχρησιμοποίησης. Αυτά έχουν ως αποτέλεσμα αστοχίες (cold misses) και επομένως το δίκτυο υπολογίζει υψηλότερο ποσοστό αστοχίας. Μπορούμε με ασφάλεια να αγνοήσουμε αυτή την έξοδο, δεδομένου ότι, με μια προσομοίωση 30 φορές μεγαλύτερης διάρκειας, η έξοδος πιθανότατα θα πλησίαζε περισσότερο την πραγματική τιμή.

Τέλος, κρατάμε τις τέσσερις πιο πιθανές κρυφές μνήμες LLC που θα έχει μια πραγματική αρχιτεκτονική και προσπαθούμε να προβλέψουμε τις άλλες τέσσερις τιμές. Δηλαδή, κρατάμε τις κρυφές μνήμες LLC των 1024, 2048, 4096 και 8192 KB στο σύνολο εκπαίδευσης και προσπαθούμε να προβλέψουμε τη συμπεριφορά των αρχείων αναφοράς που προσομοιώθηκαν με LLC μεγέθους 768, 1536, 3152 και 6044 KB. Με αυτή την είσοδο, υπολογίζουμε παρόμοια αποτελέσματα όπως προηγουμένως. Η κλίμακα των γεωμετρικών μέσων στην τρίτη σειρά του πίνακα 4.7 είναι παρόμοια με εκείνη για τα έξι μεγέθη κρυφής μνήμης στο σύνολο εκπαίδευσης.

Μπορούμε να παρατηρήσουμε στο 4.15 ότι τα σφάλματα των ακραίων τιμών εξακολουθούν να υφίστανται. Αξίζει επίσης να αναφέρουμε ότι η πρόβλεψη για 768 KB LLC είναι σαφώς, αλλά όχι πολύ χειρότερη από τις άλλες τρεις προβλέψεις. Αυτό είναι αποτέλεσμα του ότι δεν βρίσκεται μεταξύ των μεγεθών της κρυφής μνήμης που έχει δει το δίκτυο. Για παράδειγμα, το δίκτυο γνωρίζει την έξοδο για κρυφές μνήμες μεγέθους 1MB και 2MB και, επομένως, μπορεί να κατανοήσει καλύτερα πώς θα συμπεριφερόταν μια μηχανή για μια κρυφή μνήμη μεγέθους 1536 KB. Για τα 768 KB, είναι πιο δύσκολο να προεκτείνει τη συμπεριφορά του προγράμματος, και ως εκ τούτου αυξάνονται τα σφάλματα. Αυτό το σφάλμα εξακολουθεί να είναι πολύ χαμηλότερο από το καλύτερο σφάλμα που είχαμε προβλέποντας την κρυφή μνήμη μεγέθους 1MB προηγουμένως.



Σχήμα 4.15: *Απόλυτα σφάλματα πρόβλεψης του DNN για 4 μεγέθη κρυφής μνήμης, εκπαιδευμένο με ποσοστά αστοχίας 4 μεγεθών κρυφής μνήμης*

Και πάλι, η ύπαρξη μικρότερων μεγεθών κρυφής μνήμης στο σύνολο εκπαίδευσης δεν έχει νόημα, επομένως δεν θα τα παρουσιάσουμε.

Συνολικά, είναι σαφές ότι με αυτή τη χρήση, το δίκτυο προβλέπει το πρόβλημα με απόλυτη ακρίβεια. Οι προβλέψεις είναι αδιαμφισβήτητα καλύτερες από οποιεσδήποτε άλλες προβλέψεις που είχαν γίνει προηγουμένως χωρίς να γνωρίζουμε το πρόβλημα. Το μειονέκτημα αυτού του δικτύου είναι ότι πρέπει να εκτελεστεί και να εκπαιδευτεί σε όλες αυτές τις διαφορετικές αρχιτεκτονικές κρυφής μνήμης. Επομένως, εξαρτάται από την εκάστοτε εφαρμογή αν αυτό αξίζει τον κόπο.

Κεφάλαιο 5

Κατακλείδα

5.1 Συμπεράσματα

Στην παρούσα διατριβή, παρουσιάζουμε έναν αριθμό δικτύων μηχανικής μάθησης που προσφέρουν λύσεις για το πρόβλημα της πρόβλεψης των ποσοστών αστοχίας σε διαφορετικές αρχιτεκτονικές κρυφής μνήμης. Η γνώση των ποσοστών αστοχίας της κρυφής μνήμης ενός προγράμματος σε διαφορετικές αρχιτεκτονικές είναι σημαντική, καθώς επιτρέπει στον χρήστη να γνωρίζει ποια είναι η καλύτερη μηχανή για να εκτελέσει το εν λόγω πρόγραμμα. Η προσομοίωση μιας διαδικασίας είναι μια δύσκολη και χρονοβόρα διαδικασία και η παράλειψή της μέσω μιας πρόβλεψης είναι μια πολύ αποτελεσματική λύση σε αυτό το πρόβλημα.

Παρουσιάζουμε τέσσερα δίκτυα μηχανικής μάθησης που κατανοούν το πρόβλημα με τον δικό τους μοναδικό τρόπο. Εμπνευσμένοι από την ήδη υπάρχουσα πιθανολογική προσέγγιση του StatCache σε αυτό το ζήτημα, αποφασίσαμε να χρησιμοποιήσουμε τα ίδια δεδομένα που χρησιμοποιεί για τις προβλέψεις του για να εκπαιδεύσουμε και να αξιολογήσουμε ένα δίκτυο μηχανικής μάθησης. Αρχικά, ένα απλό δίκτυο MLP δείχνει ότι είναι αρκετά απλό να προβλέψει τους λόγους αστοχίας με παρόμοια ακρίβεια όπως η πιθανολογική μέθοδος της StatCache. Αυτό το δίκτυο βασίζεται στις αποστάσεις επαναχρησιμοποίησης μιας εκτέλεσης του προγράμματος και εφαρμόζεται σε οποιοδήποτε μέγεθος κρυφής μνήμης και σε οποιαδήποτε πολιτική αντικατάστασης. Προβλέπει τα ποσοστά αστοχίας οποιουδήποτε προγράμματος με οποιαδήποτε πολιτική αντικατάστασης με έναν γεωμετρικό μέσο όρο παρόμοιο με αυτόν που εισάγει η StatCache.

Η πορεία προς την όσο το δυνατόν ακριβέστερη κατανόηση των δεδομένων εισόδου μας οδήγησε στην εισαγωγή δύο νέων δικτύων. Μαζί με τις NLP ενσωματώσεις των προγραμμάτων ως είσοδο, αυτά προσπαθούν να κατανοήσουν διαφορετικά τις αποστάσεις επαναχρησιμοποίησης και έτσι να υπολογίσουν ακριβέστερες προβλέψεις του προβλήματος. Το πρώτο είναι το δίκτυο LSTM, το οποίο, όπως υποδηλώνει και το όνομά του, κατανοεί τις αποστάσεις επαναχρησιμοποίησης μέσω ενός στρώματος LSTM. Αυτό το δίκτυο είναι το πιο χρονοβόρο για την εκπαίδευσή του, αλλά παράγει μακράν τα καλύτερα αποτελέσματα από όλα τα δίκτυά μας. Παράγει σταθερές, ακριβείς προβλέψεις των αναλογιών αστοχίας για οποιαδήποτε πολιτική αντικατάστασης και μέγεθος κρυφής μνήμης που εκπαιδεύεται. Το δίκτυο CNN λαμβάνει τις αποστάσεις επαναχρησιμοποίησης ως είσοδο σε ένα στρώμα CNN που έχει ως σκοπό την κατανόηση των αποστάσεων επαναχρησιμοποίησης μέσω μιας συνέλιξης των τιμών

που βρίσκονται σε κοντινή απόσταση. Η εκπαίδευση αυτού του δικτύου είναι σημαντικά λιγότερο δαπανηρή, αλλά έχει το μειονέκτημα ότι δεν είναι τόσο ακριβές όσο το δίκτυο LSTM. Οι προβλέψεις του είναι ως επί το πλείστον ελαφρώς λιγότερο ακριβείς από τις προβλέψεις του δικτύου LSTM, αλλά για ορισμένες πολιτικές αντικατάστασης, όπως η SHiP, είναι ακόμη χειρότερες, παρόμοιες με τις προβλέψεις του δικτύου MLP. Τα δύο αυτά δίκτυα αποτελούν τα βασικά βήματα που κάνει η παρούσα διατριβή για την ακριβή κατανόηση του προβλήματος.

Τέλος, παρουσιάζουμε ένα βαθύ νευρωνικό δίκτυο που λειτουργεί για οποιοδήποτε μέγεθος κρυφής μνήμης. Αυτό, όπως και τα προηγούμενα, χρησιμοποιεί τις αποστάσεις επαναχρησιμοποίησης και τις ενσωματώσεις του κώδικα για την κατανόηση του προβλήματος. Η διαφορά είναι ότι αυτό χρησιμοποιεί το μέγεθος της κρυφής μνήμης ως είσοδο και, όπως δείξαμε, είναι σε θέση να κατανοήσει το πρόβλημα για οποιοδήποτε μέγεθος κρυφής μνήμης. Αυτό το δίκτυο που προτείνουμε είναι πολύ ισχυρό για μικρότερα μεγέθη κρυφής μνήμης και έχει μια μικρή αδυναμία για μεγαλύτερα. Αυτό το δίκτυο παράγει επίσης πολύ ακριβή αποτελέσματα εάν γνωρίζει ήδη την πραγματική τιμή ενός προβλήματος για ορισμένα μεγέθη κρυφής μνήμης και τότε μπορεί να προβλέψει την αναλογία αστοχιών οποιουδήποτε μεγέθους κρυφής μνήμης με αθέατη ακρίβεια. Αυτό το δίκτυο μπορεί να αποτελέσει εφαλτήριο για περαιτέρω έρευνα προς την κατεύθυνση παρόμοιων αρχιτεκτονικών που κατανοούν καλύτερα το πρόβλημα.

5.2 Συζήτηση

Η προηγούμενη ενότητα αποτελεσμάτων, που παρουσιάστηκε στο κεφάλαιο 4, παρουσίασε μια ολοκληρωμένη ανάλυση των εμπειρικών δεδομένων που συγκεντρώθηκαν στην παρούσα μελέτη. Αυτή η ενότητα συζήτησης αποσκοπεί στη βαθύτερη κατανόηση της σημασίας αυτών των αποτελεσμάτων σε σχέση με το ευρύτερο πλαίσιο των αρχιτεκτονικών μηχανών. Με την εξέταση αυτών των ευρημάτων μέσα από διάφορους φακούς, η ενότητα αυτή επιδιώκει να διευκρινίσει το νόημα, τη σημασία και τις πιθανές επιπτώσεις τους στο πεδίο των αρχιτεκτονικών υπολογιστών. Επιπλέον, η συζήτηση αυτή θα αναλύσει κριτικά τις επιπτώσεις αυτών των αποτελεσμάτων σε σχέση με την υπάρχουσα βιβλιογραφία, με στόχο να συνεισφέρει νέες προοπτικές και δρόμους για περαιτέρω διερεύνηση στο πεδίο.

Τα μοντέλα που προτείνουμε στην παρούσα διατριβή αμφισβητούν τα αποτελέσματα προηγούμενων μηχανισμών πρόβλεψης που αποτέλεσαν τις καλύτερες εκτιμήσεις για το πρόβλημα αυτό. Αποδεικνύονται καλύτερα από τις προβλέψεις του Στατἄσηε με σημαντική διαφορά και προβλέπουν επίσης την έξοδο οποιασδήποτε πολιτικής αντικατάστασης. Αυτό σημαίνει ότι η πρόβλεψη των αστοχιών της κρυφής μνήμης σε μια μηχανή μπορεί πλέον να περιγραφεί ως θέμα εκπαίδευσης ενός μοντέλου μηχανικής μάθησης στις εξόδους μιας μηχανής και να το αφήσουμε να προβλέψει τα ποσοστά αστοχιών. Η εισαγωγή ενός τέτοιου δικτύου δεν θα απλοποιούσε μόνο τη διαδικασία απόφασης για το ποια είναι η καλύτερη αρχιτεκτονική για ένα πρόβλημα- θα παρέλειπε επίσης πολλές ώρες επεξεργασίας ή προσομοίωσης.

Για την περαιτέρω πλαισίωση αυτών των ευρημάτων, είναι επιτακτική ανάγκη να εξετάσουμε τους ευρύτερους παράγοντες που μπορεί να επηρεάσουν τα δίκτυά μας. Η εισαγωγή προσαρμοσμένων ιχνών για τα αρχεία αναφοράς, και μαζί με αυτό, οι ενσωματώσεις του συ-

γκεκριμένου τμήματος, θα επηρέαζαν πολύ τα αποτελέσματά μας.Η πρόβλεψη ενός δικτύου που γνωρίζει ποιο μέρος του προγράμματος εκτελείται σε κάθε ίχνος, όπως συζητείται στην ενότητα 3.2.2, μπορεί να βελτιώσει σημαντικά τις προβλέψεις από αυτά τα δίκτυα.Επιπλέον, θα μπορούσε να γίνει περαιτέρω διεύρυνση προς δεδομένα που προκύπτουν από παράλληλη επεξεργασία ή δεδομένα από πραγματικές μηχανές.Η πρόβλεψη της αναλογίας αστοχίας των διεργασιών που εκτελούνται σε πολλαπλούς πυρήνες μέσα σε μια μηχανή θα εισάγει πολύ μεγαλύτερη δυσκολία στο πρόβλημα και θα πρέπει να αποτελέσει αντικείμενο περαιτέρω διερεύνησης.Τα δεδομένα που χρησιμοποιήσαμε για την παρούσα διατριβή είναι τα δεδομένα που παράγονται από ένα πρόγραμμα προσομοίωσης. Θα ήταν ενδιαφέρον να δούμε πώς συγκρίνονται αυτά τα αποτελέσματα με τα δίκτυα που εκπαιδεύτηκαν σε ένα πραγματικό μηχάνημα, δεδομένου ότι το καθένα έχει τις δικές του δυσκολίες.

Επιπλέον, υπάρχουν δυνατότητες βελτίωσης των δικτύων μας. Όπως συζητήθηκε προηγουμένως, τα δίκτυα LSTM και CNN παρουσιάζουν δυνατότητες να έχουν ακόμη καλύτερα αποτελέσματα. Η εισαγωγή περισσότερων κριτηρίων αναφοράς στο σύνολο δεδομένων ή μια σημαντική αλλαγή στην αρχιτεκτονική τους θα μπορούσε να βελτιώσει ακόμη περισσότερο τις προβλέψεις τους. Το δίκτυο DNN είναι ένα δίκτυο το οποίο δεν έχει εξερευνηθεί από εμάς στο ίδιο βάθος με τα άλλα δίκτυα. Διαφορετικές αρχιτεκτονικές και υλοποιήσεις δικτύων θα μπορούσαν να μειώσουν σημαντικά τα σφάλματά του. Θα μπορούσε ακόμη και να επεκταθεί ώστε να έχει μια πρόσθετη είσοδο, το ποσοστό αστοχίας της κρυφής μνήμης μιας εκτέλεσης σε μια κρυφή μνήμη και στη συνέχεια να κάνει εκτιμήσεις για άλλα μεγέθη κρυφής μνήμης.

Συνοψίζοντας, η εις βάθος ανάλυση και η ερμηνεία των ευρημάτων που παρουσιάζονται σε αυτή τη συζήτηση υπογραμμίζουν την πολυπλοκότητα που ενυπάρχει στην πρόβλεψη των αστοχιών στην κρυφή μνήμη. Εν τέλει, η μελέτη αυτή αποτελεί ένα σκαλοπάτι στη συνεχή προσπάθεια να διαλευκανθούν οι πολυπλοκότητες της αρχιτεκτονικής των υπολογιστών, ενθαρρύνοντας την περαιτέρω επιστημονική έρευνα για την προώθηση της κατανόησης αυτού του τομέα¨.

Μέρος III

Επίλογος

Παραρτήματα

Βιβλιογραφία

[1] N. Beckmann και D. Sanchez. *Modeling cache performance beyond LRU*. *2016 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, 2016.

[2] X. Liu Q. Wang και M. Chabby. *Featherlight Reuse-distance Measurement*. *IEEE International Symposium on High Performance Computer Architecture*, 2019.

[3] E. Berg και E. Hagersten. *StatCache: a probabilistic approach to efficient and accurate data locality analysis*. *IEEE International Symposium on - ISPASS Performance Analysis of Systems and Software*, 2004.

[4] Maurice V. Wilkes. *Slave Memories and Dynamic Storage Allocation*. *IEEE Trans. Electron. Comput.*, 14(2):270–271, 1965.

[5] Carole Jean Wu, Aamer Jaleel, Will Hasenplaugh, Margaret Martonosi, Simon C. Steely και Joel Emer. *SHiP: Signature-based Hit Predictor for high performance caching*. *2011 44th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, σελίδες 430–441, 2011.

[6] Aamer Jaleel, Kevin B. Theobald, Simon C. Steely και Joel Emer. *High performance cache replacement using re-reference interval prediction (RRIP)*. ISCA '10, σελίδα 60–71, New York, NY, USA, 2010. Association for Computing Machinery.

[7] Ishan Shah, Akanksha Jain και Calvin Lin. *Effective Mimicry of Belady's MIN Policy*. *2022 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, σελίδες 558–572, 2022.

[8] Alex Krizhevsky, Ilya Sutskever και Geoffrey E Hinton. *ImageNet Classification with Deep Convolutional Neural Networks*. *Advances in Neural Information Processing Systems*F. Pereira, C.J. Burges, L. Bottou και K.Q. Weinberger, επιμελητές, τόμος 25. Curran Associates, Inc., 2012.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren και Jian Sun. *Deep Residual Learning for Image Recognition*. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, σελίδες 770–778, 2016.

[10] Miltiadis Allamanis, Earl T. Barr, Christian Bird και Charles Sutton. *Learning Natural Coding Conventions*. FSE 2014, New York, NY, USA, 2014. Association for Computing Machinery.

[11] Miltiadis Allamanis και Charles Sutton. *Mining Idioms from Source Code*. *Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering*, FSE 2014, New York, NY, USA, 2014. Association for Computing Machinery.

[12] Amy McGovern. *Building a Basic Block Instruction Scheduler with Reinforcement Learning and Rollouts*. σελίδες 141–160, 2002.

[13] C. Cummins, P. Petoumenos, Z. Wang και H. Leather. *End-to-End Deep Learning of Optimization Heuristics*. *26th International Conference on Parallel Architectures and Compilation Techniques (PACT)*, 2017.

[14] Nan Wu και Yuan Xie. *A Survey of Machine Learning for Computer Architecture and Systems*. *ACM Comput. Surv.*, 55(3), 2022.

[15] Xiangyu Dong, Norman P. Jouppi και Yuan Xie. *A circuit-architecture co-optimization framework for exploring nonvolatile memory hierarchies*. *ACM Trans. Archit. Code Optim.*, 10(4), 2013.

[16] Ramazan Bitirgen, Engin Ipek και Jose F. Martinez. *Coordinated management of multiple interacting resources in chip multiprocessors: A machine learning approach*. *2008 41st IEEE/ACM International Symposium on Microarchitecture*, σελίδες 318–329, 2008.

[17] Daniel Nemirovsky, Tugberk Arkose, Nikola Markovic, Mario Nemirovsky, Osman Unsal και Adrian Cristal. *A Machine Learning Approach for Performance Prediction and Scheduling on Heterogeneous CPUs*. σελίδες 121–128, 2017.

[18] S. VenkataKeerthy, Rohit Aggarwal, Shalini Jain, Maunendra Sankar Desarkar, Ramakrishna Upadrasta και Y. N. Srikant. *IR2VEC: LLVM IR Based Scalable Program Embeddings*. *ACM Trans. Archit. Code Optim.*, 17(4), 2020.

[19] Yulei Sui, Xiao Cheng, Guanqin Zhang και Haoyu Wang. *Flow2Vec: Value-Flow-Based Precise Code Embedding*. *Proc. ACM Program. Lang.*, 4(ΟΟΠΣΛΑ), 2020.

[20] John L. Henning. *SPEC CPU2006 Benchmark Descriptions*. *SIGARCH Comput. Archit. News*, 34(4):1–17, 2006.

[21] James Bucek, Klaus Dieter Lange και Jóakimv. Kistowski. *SPEC CPU2017: Next-Generation Compute Benchmark*. *Companion of the 2018 ACM/SPEC International Conference on Performance Engineering*, ICPE '18, σελίδα 41–42, New York, NY, USA, 2018. Association for Computing Machinery.

[22] Nathan Gober, Gino Chacon, Lei Wang, Paul V. Gratz, Daniel A. Jimenez, Elvira Teran, Seth Pugsley και Jinchun Kim. *The Championship Simulator: Architectural Simulation for Education and Competition*, 2022.

[23] Geoffrey Webb, Claude Sammut, Claudia Perlich, Tamás Horváth, Stefan Wrobel, Kevin Korb, William Noble, Christina Leslie, Michail Lagoudakis, Novi Quadrianto,

Wray Buntine, Lise Getoor, Galileo Namata, Jiawei Jin, Jo Anne Ting, Sethu Vijayakumar, Stefan Schaal και Luc De Raedt. *Leave-One-Out Cross-Validation*. 2010.

[24] Diederik P. Kingma και Jimmy Ba. *Adam: A Method for Stochastic Optimization*. *CoRR*, αβσ/1412.6980, 2014.

[25] Cem Subakan, Mirco Ravanelli, Samuele Cornell, Mirko Bronzi και Jianyuan Zhong. *Attention Is All You Need In Speech Separation*. *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, σελίδες 21–25, 2021.

Συντομογραφίες - Αρκτικόλεξα - Ακρωνύμια

ΚΜΕ	Κεντρική Μονάδα Επεξεργασίας
MSE	Mean Squared Error
IR	Intermediate representation
NLP	Natural Language Processing
LRU	Least Recently Used
MLP	Multi-Layered Perceptron
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
CNN	Convolutional Neural Network
DNN	Deep Neural Network
RL	Reinforcement Learning

Απόδοση ξενόγλωσσων όρων

Απόδοση	Ξενόγλωσσος όρος
συστάδα	cluster
στρώμα συσσώρευσης μεγίστου	max-pooling layer
στρώμα συσσώρευσης μέσου	average-pooling layer
στρώμα συσσώρευσης	pooling layer
ευστοχία	hit
πυκνό νευρωνικό δίκτυο	dense neural network
αποστάσεις επαναχρησιμοποίησης	reuse distances
αρχείο αναφοράς	benchmark
ενσωμάτωση	embedding
ποσοστό αστοχίας	miss ratio
αρχείο ίχνους	trace file
περίπτωση χρήσης	use case
μετασχηματιστής	transformer
βελτιστοποιητής	optimizer
μέγεθος παρτίδας	batch size
γεωμετρικός μέσος όρος	geometric mean
χάρτης κατακερματισμού	hashmap
θηκόγραμμα	boxplot
μουστάκι	whisker
κλιμακωτής	scaler
κλιμάκωση	scaling
βήμα	stride
πυρήνας	kernel
προσοχή	attention
στρώμα εγκατάληψης	drop-out layer
υπερβολική προσαρμογή	overfitting
συνελικτικό νευρωνικό δίκτυο	convolutional neural network
ανάκτηση πληροφορίας	information retrieval
αντιμεταθετικότητα	commutativity
απόγονος	descedant
απορρόφηση	absorption
βάση δεδομένων	database
γνώρισμα	attribute
διαπροσωπεία	interface

διαφορά	difference
δικτυακός κατάλογος	portal catalog
δικτυωτή δομή	lattice
δομικές επερωτήσεις	structural queries
δομικές σχέσεις	structural relationships
δομικό σχήμα	schema
εγκυρότητα	validity
ένωση	union
αδερφός	sibling
αμεταβλητότητα	idempotency