Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων
Τομεας Τεχνολογιας Πληροφορικης και Υπολογιστων

Αυτόματη Μεταγραφή Μουσικής Κιθάρας σε Ταμπλατούρα με Χρήση Νευρωνικών Δικτύων

Μελέτη και υλοποίηση

Διπλωματικη Εργασια

του

ΚΩΝΣΤΑΝΤΙΝΟΥ Π. ΒΟΣΙΝΑ

Επιβλέπων: Γιώργος Στάμου
Καθηγητής

Αθήνα, Μάιος 2022

Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων
Τομεας Τεχνολογιας Πληροφορικης και Υπολογιστων

Αυτόματη Μεταγραφή Μουσικής Κιθάρας σε Ταμπλατούρα με Χρήση Νευρωνικών Δικτύων

Μελέτη και υλοποίηση

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΚΩΝΣΤΑΝΤΙΝΟΥ Π. ΒΟΣΙΝΑ

Επιβλέπων: Γιώργος Στάμου
Καθηγητής

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 22α Νοεμβρίου 2022.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Γιώργος Στάμου	Ανδρέας-Γεώργιος Σταφυλοπάτης	Στέφανος Κόλλιας
Καθηγητής	Καθηγητής	Καθηγητής

Αθήνα, Μάιος 2022

Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων
Τομεας Τεχνολογιας Πληροφορικης και Υπολογιστων

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)

.....
Κωνσταντίνος Βόσινας
Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

13 Μαίου 2022

Περίληψη

Η μεταγραφή της μουσικής είναι ένα από τα κυριότερα θέματα του τομέα της Ανάκτησης Μουσικής Πληροφορίας (Music Information Retrieval), με τις εφαρμογές της να είναι περιζήτητες, τόσο από επαγγελματίες μουσικούς όσο και για την εκμάθηση μουσικής. Τα τελευταία χρόνια η Μηχανική Μάθηση και τα Νευρωνικά Δίκτυα έχουν χρησιμοποιηθεί σε μεγάλο βαθμό για την επίλυση τέτοιων προβλημάτων με αυτόματο τρόπο, μέσω της ανάπτυξης διαφόρων μοντέλων για τη μεταγραφή μουσικών κομματιών.

Σκοπός αυτής της διπλωματικής εργασίας είναι η δημιουργία και εξέταση μοντέλων μηχανικής μάθησης για τη μεταγραφή μουσικής κιθάρας, στην πλέον χρησιμοποιούμενη αναπαράσταση της, την ταμπλατούρα. Η μεταγραφή της μουσικής έγκειται σε θεμελιώδες επίπεδο στην αναγνώριση των χαρακτηριστικών κάθε νότας ενός κομματιού, τα κύρια από τα οποία είναι η χρονική στιγμή έναρξης, το τονικό ύψος και η διάρκεια. Τα περισσότερα υπάρχοντα συστήματα μεταγραφής αποσκοπούν στην δημιουργία παρτιτούρας, μια αναπαράσταση που είναι σχεδόν καθολική για όλα τα μουσικά όργανα. Παρόλα αυτά, η κιθάρα ως όργανο παρουσιάζει κάποιες ιδιαιτερότητες, η βασικότερη από τις οποίες είναι πως μία νότα μπορεί να παιχτεί σε πολλά διαφορετικά σημεία, το οποίο οδηγεί στην ανάπτυξη της ταμπλατούρας, ένα συμβολικό τρόπο αναπαράστασης της μουσικής για την κιθάρα. Η κύρια συνεισφορά της εργασίας είναι η ανάπτυξη ενός συστήματος αναπαράστασης ώστε να υπάρξει μια εύκολη μετατροπή υπάρχουσων αρχιτεκτονικών στην μεταγραφή σε παρτιτούρα, καθώς και η ανάπτυξη νέων αρχιτεκτονικών που επιτελούν αυτό το σκοπό.

Η προσέγγιση μας βασίζεται αρχικά στην χρήση μιας νέας αναπαράστασης νοτών προσαρμοσμένη στην κιθάρα. Στη συνέχεια αναπτύχθηκαν με βάση υπάρχουσες αρχιτεκτονικές νέα μοντέλα. Στο πρώτο είδος χρησιμοποιούμε απευθείας τη νέα αναπαράσταση, ενώ στο δεύτερο είδος, προσπαθούμε να χρησιμοποιήσουμε υπάρχουσες αρχιτεκτονικές για μεταγραφή μουσικής και στη συνέχεια να γίνει η μετατροπή σε ταμπλατούρα.

Στα πειράματα χρησιμοποιήθηκε ένα Dataset που είναι φτιαγμένο για μεταγραφή μουσικής κιθάρας, το GuitarSet. Μετά τα πειράματα, συγκρίνουμε τα αποτελέσματα των νευρωνικών τόσο μεταξύ τους, όσο και με την απόδοση ήδη υπάρχουσων δικτύων. Τέλος, παρουσιάζουμε κάποιες μελλοντικές επεκτάσεις που μπορούν να εφαρμοστούν.

Λέξεις Κλειδιά

Ανάκτηση Μουσικής Πληροφορίας, Μεταγραφή Μουσικής Κιθάρας, Ταμπλατούρα, Βαθιά Μάθηση, Συνελικτιά Νευρωνικά Δίκτυα, Αναδρομικά Νευρωνικά Δίκτυα, Κιθάρα, Φασματογράφημμα, Τεχνικές Ομαλοποίησης

Abstract

Music transcription is one of the main tasks in the field of Music Information Retrieval, with its applications being sought after, both by professional musicians and for teaching music. During the recent years, Machine Learning and Neural Networks have been widely used to solve such tasks automatically, by developing models for music transcription.

The main goal of this thesis is the development and examination of machine learning models for guitar transcription, on its most widely used representation, tablature. Music transcription at a fundamental level, boils down to the recognition of the basic characteristics of each note in the track, the main ones being the note onset, pitch and duration. Most existing transcription systems aim to create sheet music, a representation that is almost universal for all musical instruments. However, the guitar shows some unique qualities as an instrument, the most important one being the fact that a note can be played several at several different positions, which leads to the development of tablature, a representation geared towards guitar specifically. The main contributions of this thesis is the development of a note representation which helps transform existing models to be used in tablature transcription, and the development of new architectures which perform this task.

Our approach is based on using a new note representation focused on the guitar. Afterwards, we develop new models, based on existing architectures. The first kind of such models uses the new representation immediately, while the second kind uses the existing architectures to transcribe sheet music, and then translate the results into tablature.

During our experiments, we used a Dataset created for guitar transcription, GuitarSet. After the experiments we compare the results between the different models, and also compare their performance with existing transcription models. Finally, we present some future extensions that could be applied.

Keywords

Music Information Retrieval, Guitar Music Transcription, Tablature, Deep Learning, Convolutional Neural Networks, Recurrent Neural Networks, Guitar, Spectrogram, Regularisation Methods

στους γονείς μου

Ευχαριστίες

Θα ήθελα καταρχήν να ευχαριστήσω τον καθηγητή κ. Γ. Στάμου για την επίβλεψη αυτής της διπλωματικής εργασίας και για την ευκαιρία που μου έδωσε να την εκπονήσω στο εργαστήριο Εργαστήριο Συστημάτων Τεχνητής Νοημοσύνης και Μάθησης. Επίσης ευχαριστώ ιδιαίτερα τον Ε. Δερβάκο για την καθοδήγησή του και την εξαιρετική συνεργασία που είχαμε. Τέλος θα ήθελα να ευχαριστήσω τους γονείς μου για την καθοδήγηση και τη στήριξη που μου παρείχαν καθόλη τη διάρκεια των σπουδών μου.

Αθήνα, Μάιος 2022

Κωνσταντίνος Βόσινας

Περιεχόμενα

Κατάλογος Σχημάτων

Κατάλογος Πινάκων

Κεφάλαιο 1

Εισαγωγή

Η μεταγραφή μουσικής επιτελεί ένα θεμελιώδη σκοπό για όλη τη σταδιοδρομία ενός μουσικού, από την ανάγκη για εκμάθηση ως την ανάγκη για επικοινωνία με άλλους μουσικούς. Η δημοτικότητα της κιθάρας, τόσο από άποψη χρήσης στη δημοφιλή μουσική, όσο και σαν όργανο που διδάσκεται, οδηγεί σε μεγάλη ζήτηση μεταγραφής κομματιών.

Η αυτόματη μεταγραφή μουσικής είναι ένα βασικό πρόβλημα της Ανάκτησης Μουσικής Πληροφορίας (MIR), στο οποίο συνεχώς αναπτύσσονται νέες λύσεις. Το πρόβλημα έγκειται στην εξαγωγή μιας συμβολικής αναπαράστασης της μουσικής από ένα μουσικό σήμα εισόδου. Παραδοσιακά, αυτή η μετατροπή περιλαμβάνει σε μικρό ή μεγάλο βαθμό επεξεργασία σήματος, ενώ τα τελευταία χρόνια, η μηχανική μάθηση και τα νευρωνικά δίκτυα έχουν βοηθήσει στην επίλυση προβλημάτων στα οποία η απλή επεξεργασία σήματος δεν αρκεί.

1.1 Κίνητρο της Εργασίας

Πολλά συστήματα μηχανικής μάθησης έχουν αναπτυχθεί με σκοπό τη μεταγραφή μουσικής, με κάποια παραδείγματα να αναφέρονται στα [1], [2], [3], τα οποία όμως επικεντρώνονται συχνά στη μεταγραφή μουσικής είτε στις κατευθείαν σε νότες, είτε σε μορφή παρτιτούρας. Παρόλα αυτά, αυτή δεν είναι η επιθυμητή αναπαράσταση σε πολλές περιπτώσεις, όπως για την κιθάρα, όπου πολλοί κιθαρίστες, ιδιαίτερα οι αυτοδίδακτοι και μαθητευόμενοι, θεωρούν πιο χρήσιμη την αναπαράσταση σε ταμπλατούρα. Οι λόγοι που αυτή η αναπαράσταση είναι πιο επιθυμητή σε διάφορες περιπτώσεις θα αναλυθεί στη συνέχεια. Είναι χρήσιμο επομένως, να χρησιμοποιήσουμε τεχνικές και μεθόδους που έχει αποδειχθεί πως δίνουν ικανοποιητικά αποτελέσματα σε άλλου είδους μεταγραφές, ώστε να αναπτύξουμε ένα μοντέλο μεταγραφής στοχευμένο για την κιθάρα.

1.2 Συνεισφορά

Στη διπλωματική αυτή μελετάμε τη χρήση υπαρχουσών και νέων τεχνικών και μοντέλων για την αυτόματη μεταγραφή μουσικής κιθάρας σε ταμπλατούρα. Χρησιμοποιούμε μια αναπαράσταση που επιτρέπει την χρήση αυτή των ήδη υπαρχουσών μοντέλων. Χρησιμοποιούμε το σύνολο δεδομένων (Dataset) GuitarSet ([4], [5]), το οποίο είναι φτιαγμένο για αυτόν ακριβώς το σκοπό. Δοκιμάζουμε διάφορες νέες αρχιτεκτονικές και τεχνικές ομαλοποίησης και συγκρίνουμε τα αποτελέσματα με τις υπάρχουσες μελέτες.

1.3 Οργάνωση του τόμου

Στο Κεφάλαιο 2 δίνεται η απαραίτητη θεωρία που σχετίζεται με τη μουσική και την επεξεργασία ήχου. Αρχικά, παρουσιάζονται κάποιες βασικές έννοιες της θεωρίας της μουσικής, κάποια χαρακτηριστικά της κιθάρας ως όργανο. Επιπλέον, παρουσιάζονται οι βασικοί τρόποι μεταγραφής της μουσικής, όπως η παρτιτούρα, η ταμπλατούρα και η αναπαράσταση MIDI (Musical Instrument Digital Interface). Στη συνέχεια, παρουσιάζονται κάποιες βασικές αρχές της επεξεργασίας σήματος που θα χρησιμοποιηθεί, όπως οι διάφοροι μετασχηματισμοί Fourier και το φασματογράφημα.

Στο Κεφάλαιο 3 παρουσιάζονται οι βασικές αρχές των νευρωνικών δικτύων και της μηχανικής μάθησης. Περιγράφονται τα διάφορα θεμελιώδη δίκτυα που θα χρησιμοποιηθούν στις αρχιτεκτονικές, όπως τα Συνελικτικά και τα Αναδρομικά δίκτυα. Επιπλέον, περιγράφονται οι αλγόριθμοι εκπαίδευσης των δικτύων αυτών και το απαραίτητο θεωρητικό υπόβαθρο για την κατανόηση της λειτουργίας τους.

Στο Κεφάλαιο 4 περιγράφεται ο σχεδιασμός των πειραμάτων που διεξήχθησαν κατά τη διάρκεια της υλοποίησης της διπλωματικής. Αναλύουμε αρχικά το σύνολο δεδομένων και την αναπαράσταση που χρησιμοποιείται για την εφαρμογή μας στην κιθάρα. Στη συνέχεια, περιγράφονται οι διάφορες αρχιτεκτονικές δικτύων που δοκιμάστηκαν και η διαδικασία εκπαίδευσής τους. Τέλος, παρουσιάζονται κάποιες τεχνικές βελτιστοποίησης και μέθοδοι ομαλοποίησης.

Στο Κεφάλαιο 5 παρουσιάζεται η αξιολόγηση των μοντέλων μετά την εκπαίδευσή τους. Αρχικά, ορίζονται οι μετρικές που χρησιμοποιήθηκαν για την μέτρηση της απόδοσης των δικτύων. Στη συνέχεια, παρουσιάζονται αυτές οι μετρικές για το κάθε μοντέλο, όπως αυτές προέκυψαν ύστερα από την εκπαίδευσή τους. Τέλος, παρουσιάζεται ένα παράδειγμα πραγματικής μεταγραφής από τα καλύτερα μοντέλα.

Μέρος Ι

Θεωρητικό Μέρος

Κεφάλαιο 2

Μουσική και Επεξεργασία Ήχου

Στο κεφάλαιο αυτό θα εξετάσουμε κάποιες βασικές έννοιες που αφορούν τη μουσική και τον ήχο. Συγκεκριμένα, θα εξετάσουμε κάποιες βασικές αρχές της θεωρίας της μουσικής, με σκοπό να κατανοήσουμε τα προβλήματα προς επίλυση. Επιπλέον, θα αναλυθούν κάποια βασικά στοιχεία επεξεργασίας ήχου που χρειάζονται για την υλοποίηση των μοντέλων και των πειραμάτων.

2.1 Μουσική

2.1.1 Θεωρία της Μουσικής

Η θεωρία της μουσικής αποσκοπεί στη μελέτη της τρόπου με τον οποίο οι μουσικοί εκτελούν και παράγουν μουσική, το οποίο μπορεί να περιλαμβάνει μεταξύ άλλων τονικά συστήματα, μεθόδους σύνθεσης, περιγραφή της μουσικής, μεταξύ άλλων. Τα βασικά χαρακτηριστικά της μουσικής διαφέρουν ανάλογα με τους διαφορετικούς ορισμούς που δίνονται, αλλά μερικά από τα κύρια που αναφέρονται στα [6], [7] είναι το τονικό ύψος, ο ρυθμός, οι δυναμικές και το ηχόχρωμα.

Το τονικό ύψος είναι μια βασική ιδιότητα του ήχου, η οποία διαφοροποιεί τους ¨ψηλούς" και τους ¨χαμηλούς" ήχους. Βασίζεται στην αντίληψη των ανθρώπων της συχνότητας ταλάντωσης ως ήχου. Παρόλο που έχουν άμεση σχέση, η συχνότητα και το τονικό ύψος δεν είναι ταυτόσημα, αφού το τονικό ύψος βασίζεται σε μεγάλο βαθμό στην αντίληψη του ανθρώπου. Υψηλότερες συχνότητες ταλάντωσης αντιστοιχούν σε ¨ψηλότερες¨ νότες.

Στη δυτική μουσική θεωρία, δίνονται ονόματα σε επιλεγμένες συχνότητες, οι οποίες αποτελούν τις μουσικές νότες ή μουσικούς φθόγγους και αποτελούν το αλφάβητο της μουσικής σύνθεσης. Οι νότες είναι οργανωμένες με βάση τη φυσική ή χρωματική κλίμακα, η οποία αποτελείται από 12 νότες, με βάση τη θεμελιώδη συχνότητά τους. Η απόσταση μεταξύ δύο συνεχόμενων νοτών ονομάζεται ημιτόνιο, ενώ η απόσταση μεταξύ μιας νότας και της ίδιας νότας διπλάσιας συχνότητας ονομάζεται οκτάβα. Σύμφωνα με το πιο διαδεδομένο σύστημα κουρδίσματος (12 tone equal temperament), η οκτάβα χωρίζεται σε 12 μέρη, τα οποία είναι ίσα σε λογαριθμική κλίμακα. Τα ονόματα των νοτών σε μια οκτάβα είναι τα ακόλουθα:

Λα (A) - Λα ♯(A♯) - Σι (B) - Ντο (C) - Ντο ♯(C♯) - Ρε (D) - Ρε ♯(D♯) - Μι (E) - Φα (F) - Φα ♯(F♯) - Σολ (G) - Σολ ♯(G♯)

Το σύμβολο ♯ονομάζεται δίεση και υποδηλώνει την νότα ένα ημιτόνιο πάνω από την

αναφερόμενη. Αντίστοιχα το σύμβολο ♭ονομάζεται ύφεση και δηλώνει τη νότα ένα ημιτόνιο χαμηλότερα. Μπορεί δυο διαφορετικοί τρόποι γραφής να αναφέρονται στην ίδια συχνότητα (για παράδειγμα Λα ♯και Σι ♭), στην οποία περίπτωση ονομάζονται εναρμόνιοι φθόγγοι.

Το ύψος της οκτάβας αναπαρίσταται με ένα αριθμό δίπλα στην αντίστοιχη νότα, το οποίο υποδηλώνει την απόσταση από την αρχή των 88 νοτών ενός πιάνου. Για παράδειγμα η νότα A4, είναι η Λα στην 4ή οκτάβα. Με βάση αυτή τη νότα ορίζεται το σύστημα 12-tone equal temperament, συγκεκριμένα η νότα A4 αντιστοιχεί στη θεμελιώδη συχνότητα 440 Hz.

Ο ρυθμός αναφέρεται στην χρονική οργάνωση της μουσικής. Αυτό μπορεί να αφορά το μέτρο του κομματιού, αλλά και τη διάρκεια των νοτών. Για τις εφαρμογές μουσικής μεταγραφής, τα πιο χρήσιμα χαρακτηριστικά είναι η έναρξη (¨onset"), η λήξη (¨offset") και η διάρκεια (¨duration") μιας νότας.

Το ηχόχρωμα (timbre) αφορά την ποιότητα ενός ήχου ενός οργάνου ή φωνής. Είναι το χαρακτηριστικό που ξεχωρίζει τους ήχους από διαφορετικά όργανα, παρόλο που παράγουν τον ίδιο τόνο και την ίδια ένταση. Για παράδειγμα, η ίδια νότα μπορεί να ακουστεί διαφορετική, αν παιχτεί σε ένα βιολί και σε μια κιθάρα. Ο κύριος λόγος που τα διαφορετικά όργανα έχουν διαφορετικό ηχόχρωμα, είναι πως οι αρμονικές τους διαφέρουν, γεγονός που αναλύεται στην αντίστοιχη ενότητα. Ο μουσικός μπορεί επίσης να επηρεάσει το ηχόχρωμα του οργάνου με τον τρόπο παιξίματος, όπως για παράδειγμα το παίξιμο κιθάρας με πένα ή με δάχτυλα.

Τέλος, η ένταση είναι βασικό χαρακτηριστικό του ήχου. Αναφέρεται στο πόσο δυνατά γίνεται αντιληπτός ένας ήχος και μετριέται στην κλίμακα Decibel (dB).

2.1.2 Η κιθάρα

Η κιθάρα είναι ένα από τα πιο διάσημα μουσικά όργανα, με εφαρμογές σε πολλά είδη μουσικής. Αποτελείται συνήθως από έξι χορδές (αν και υπάρχουν κιθάρες με περισσότερες), και 22 ¨τάστα", δηλαδή μεταλλικά ελάσματα που τοποθετούνται στο μπράτσο της κιθάρας και υποδεικνύουν τους τόνους που παράγονται αν πατηθούν οι χορδές σε αυτό το σημείο. Η απόσταση μεταξύ δύο συνεχόμενων τάστων είναι ένα ημιτόνιο. Οι κιθάρες κουρδίζονται παραδοσιακά ως εξής: Μι - Λα - Ρε - Σολ - Σι - Μι.

Ο ήχος της κιθάρας παράγεται είτε ακουστικά, μέσω του ηχείου στο κάτω μέρος της, είτε ηλεκτρικά, χρησιμοποιώντας ¨μαγνήτες" για να μετατραπεί το σήμα από τις χορδές της κιθάρας σε ηλεκτρικό, το οποίο στη συνέχεια ενισχύεται μέσω ενός ενισχυτή. Οι τόνοι της κιθάρας παράγονται μέσω της ταλάντωσης των χορδών μεταξύ δύο σημείων. Το ένα σημείο είναι σταθερό και είναι η γέφυρα της κιθάρας, το σημείο δηλαδή στο οποίο ξεκινούν οι χορδές, και το άλλο μεταβάλλεται καθώς ο οργανοπαίχτης πιέζει συγκεκριμένα τάστα στην ταστιέρα.

Η κιθάρα, όπως και άλλα αναλογικά όργανα, δεν παράγουν έναν μόνο τόνο όταν πάλλεται μία χορδή της. Αντιθέτως, παράγει ένα συνδυασμό συχνοτήτων, που αποτελείται από τη θεμελιώδη συχνότητα και τις αρμονικές. Οι αρμονικές είναι συχνότητες που παράγονται ταυτόχρονα με τη θεμελιώδη συχνότητα και προσδίδουν το ηχόχρωμα στο κάθε όργανο, καθώς έχουν διαφορετική ένταση σε διαφορετικά όργανα.

Με βάση τα δύο σταθερά σημεία, όταν η χορδή πάλλεται, δημιουργείται κατά μήκος

της στάσιμο κύμα, μεταφέροντας την ενέργεια που δίνει ο οργανοπαίχτης χτυπώντας τη. Το στάσιμο κύμα αυτό προκύπτει από την ανάκλαση στα δύο σταθερά σημεία του αρχικού κύματος. Η προσθήκη αυτών των ανακλώμενων κυμάτων δημιουργεί δεσμούς, δηλαδή σημεία τα οποία πάλλονται με μέγιστο πλάτος και κοιλίες, σημεία που δεν ταλαντώνονται.

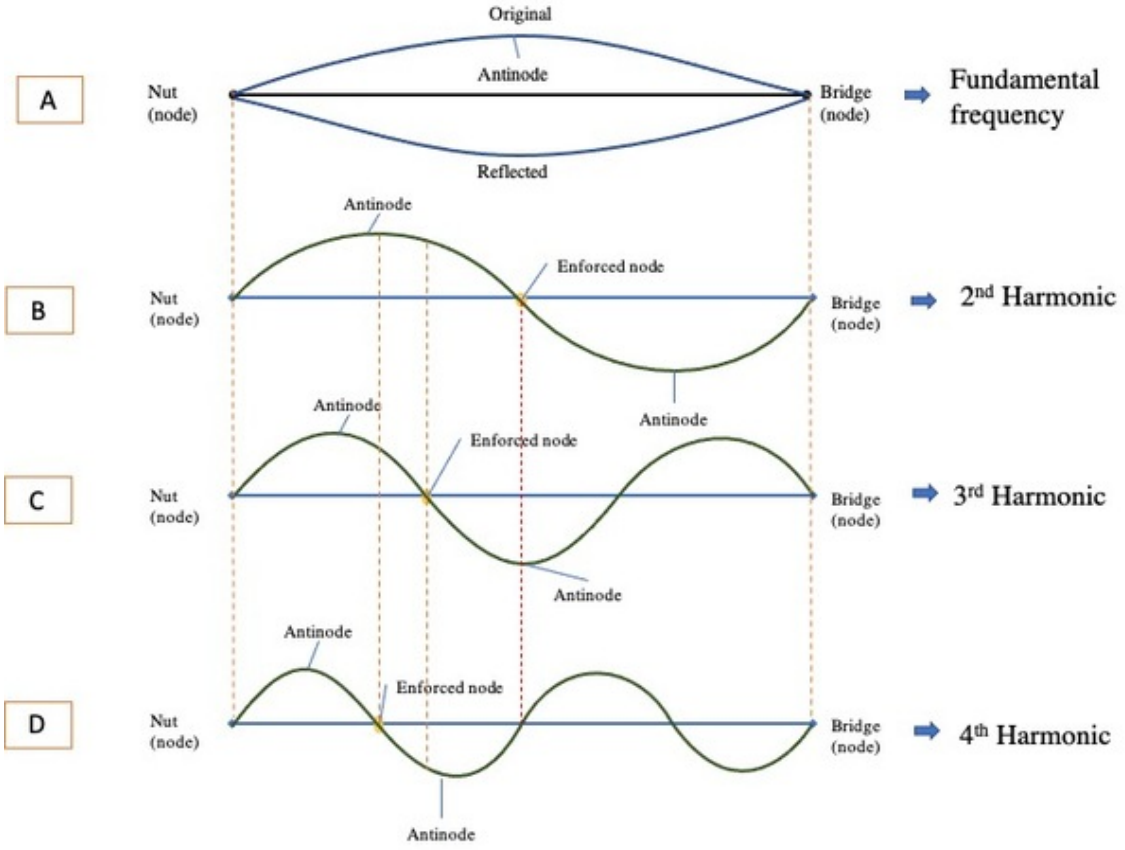
Τα μήκη κύματος των στάσιμων κυμάτων λαμβάνουν τιμές που ικανοποιούν τη σχέση 2.1.

$$\lambda = \frac{2L}{n}, \quad n = 1, 2, \ldots \tag{2.1}$$

Οι συχνότητες των κυμάτων αυτών εξαρτώνται από την ταχύτητα μετάδοσης του κύματος v καθώς και το μήκος της χορδής L, και δίνονται από τη σχέση 2.2.

$$f = \frac{v}{\lambda} = \frac{nv}{2L}, \quad n = 1, 2, \ldots \tag{2.2}$$

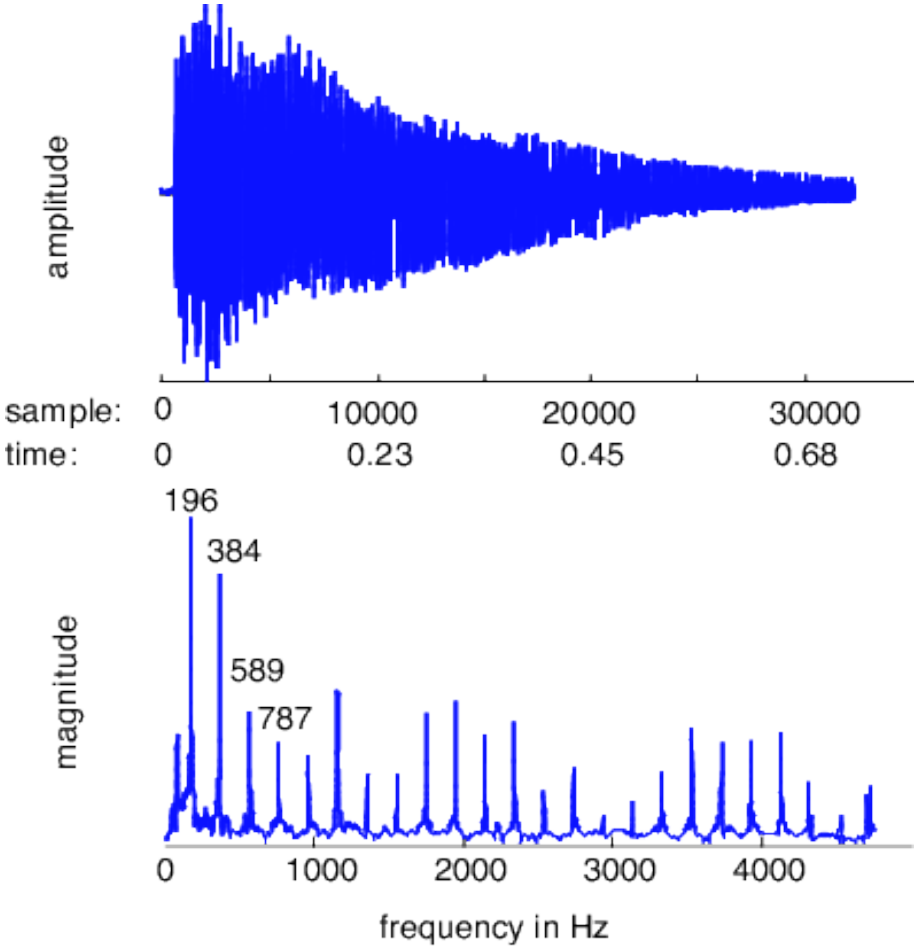
Η συχνότητα που προκύπτει για $n = 1$ αποτελεί τη θεμελιώδη συχνότητα της χορδής ή την πρώτη αρμονική και έχει μήκος κύματος όσο το διπλάσιο του μήκους της χορδής. Στο σχήμα 2.1 φαίνονται σχηματικά τα παραπάνω κύματα, τα οποία παρατηρούμε πως προκύπτουν προσθέτοντας επιπλέον ισαπέχουσες κοιλίες κατά μήκος της χορδής.



Σχήμα 2.1: *Κυματομορφές των αρμονικών σε μια χορδή.*

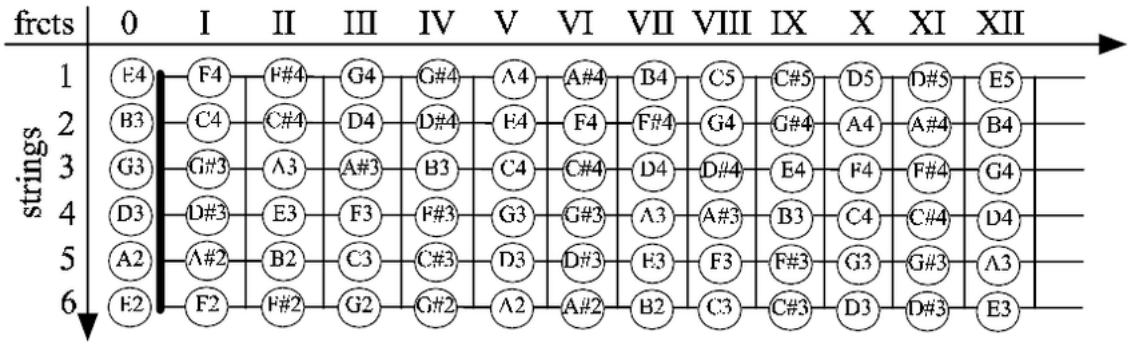
Η ένταση των διάφορων αρμονικών εξαρτάται από πολλούς παράγοντες και διαφέρει από όργανο σε όργανο, προσδίδοντας σε αυτά τη χροιά τους. Παράγοντες όπως το υλικό κατασκευής, το είδος των χορδών, ακόμα και το σημείο όπου ο οργανοπαίχτης τραβά τη

χορδή επηρεάζουν την ένταση των αρμονικών της κιθάρας. Στο σχήμα 2.2 φαίνεται το φασματογράφημα μιας νότας παιγμένης στην κιθάρα, στο οποίο παρατηρούμε τις ξεκάθαρες κορυφές οι οποίες αποτελούν τις συχνότητες των αρμονικών.



Σχήμα 2.2: *Φασματογράφημα νότας χορδής κιθάρας.*

Οι νότες που παράγονται στην κιθάρα φαίνονται στο σχήμα 2.3. Ένα βασικό χαρακτηριστικό της κιθάρας είναι το γεγονός ότι ο ίδιος τόνος μπορεί να παραχθεί από διαφορετικούς συνδυασμούς χορδής και τάστου. Για παράδειγμα η νότα C3 μπορεί να παραχθεί από το όγδοο τάστο της έκτης χορδής και από το τρίτο τάστο της πέμπτης χορδής. Αυτή η ασάφεια μπορεί να οδηγήσει σε προβλήματα στην μεταγραφή της μουσικής για κιθάρα, αφού μπορεί να χαθεί πληροφορία σχετικά με την ακριβή θέση της νότας.

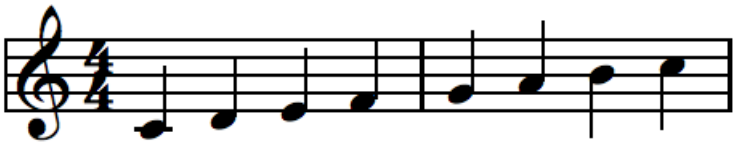


Σχήμα 2.3: *Αντιστοιχία νοτών στα Τάστα της Κιθάρας*

2.1.3 Μεταγραφή Μουσικής

Η μεταγραφή της μουσικής είναι η διαδικασία της εγγραφής μέσω ενός συστήματος συμβόλων ενός ήχου ή ενός μουσικού κομματιού. Υπάρχουν διάφορα είδη μεταγραφής, μερικά από τα οποία περιορίζονται σε συγκεκριμένα όργανα, κάνοντας χρήση των ιδιαιτεροτήτων τους με σκοπό τον αποδοτικότερο τρόπο γραφής. Θα αναλύσουμε αρχικά τον πιο διαδεδομένο σύστημα μεταγραφής, την παρτιτούρα. Στη συνέχεια θα δούμε το σύστημα μεταγραφής που επικεντρώνεται στην κιθάρα, την παρτιτούρα, καθώς και το σύστημα MIDI, το οποίο χρησιμοποιείται εκτενώς για από τους υπολογιστές.

Η παρτιτούρα είναι ο πιο διάσημη μέθοδος μεταγραφής μουσικής, και βρίσκει εφαρμογή σε τεράστιο πλήθος οργάνων και ειδών μουσικής. Ο κύριος σκοπός της παρτιτούρας, είναι να χρησιμοποιηθεί ως οδηγός ή μέσο για την εκτέλεση ενός μουσικού κομματιού. Ένα παράδειγμα παρτιτούρας φαίνεται στο σχήμα 2.4.



Σχήμα 2.4: *Παρτιτούρα της κλίμακας ντο ματζόρε, C4 (261.63 Hz) έως C5 (523.25 Hz)*

Για την κατανόηση της παρτιτούρας, απαιτείται γνώση του μουσικού συμβολισμού. Οι νότες αναπαρίστανται σε ένα πλέγμα, το οποίο αποτελείται από πέντε παράλληλες γραμμές. Η νότα μπορεί να βρίσκεται είτε πάνω σε μια γραμμή, είτε ανάμεσα σε δύο γραμμές (βοηθητικές γραμμές προστίθενται αν το πεντάγραμμο δεν αρκεί). Η θέση της νότας ορίζει το τονικό ύψος της. Άλλα βασικά στοιχεία της παρτιτούρας αποτελούν τα ακόλουθα

- Κλειδί, το οποίο ορίζει ποια νότα αντιστοιχεί στο σημείο το οποίο δείχνει. Για παράδειγμα το κλειδί του Σολ, επισημαίνει την νότα G4, ενώ το κλειδί του Φα, επισημαίνει την νότα F3. Με αυτό τον τρόπο μπορούν να μεταγραφούν όργανα με διαφορετικό εύρος συχνοτήτων.

- Ο οπλισμός (key signature) ορίζει την κλίμακα του κομματιού, αποτελείται από ένα

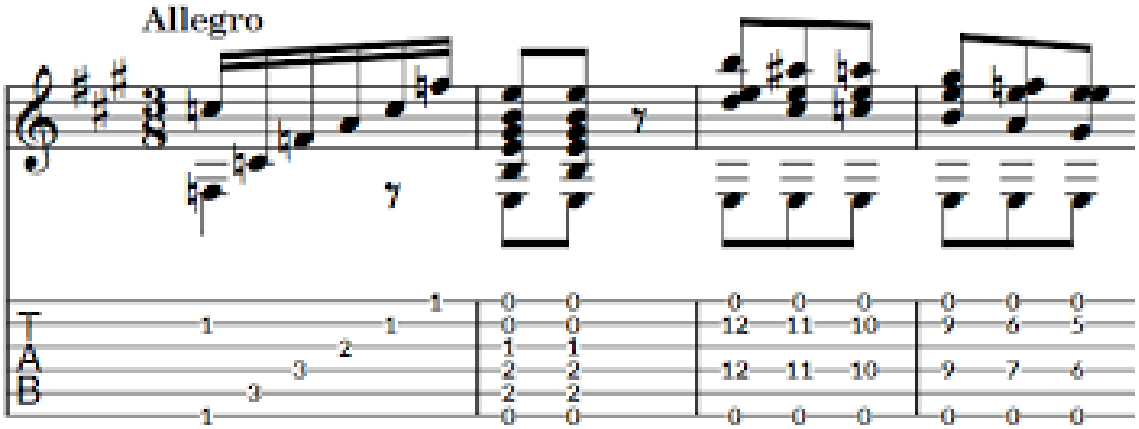
σύνολο διέσεων ή υφέσεων και υποδεικνύει πως όλες οι εμφανίσεις της αντίστοιχης νότας πρέπει να παίζονται με την αντίστοιχη αλλοίωση (χωρίς να χρειαστεί να ξαναγραφτεί).

- Τελευταίο χαρακτηριστικό είναι μετρικός οπλισμός που ορίζει πόσοι παλμοί ανήκουν σε κάθε μέτρο. Η χρονική πληροφορία αναπαρίσταται στην οριζόντια κατεύθυνση, με διαφορετικά σχήματα νοτών να υποδηλώνουν διαφορετικές διάρκειες.

Ο συμβολισμός αυτός, παρόλο που είναι σε πολύ μεγάλο βαθμό κοινός για όλα τα όργανα, δεν είναι απολύτως αντίστοιχος με τη μορφή της κιθάρας. Αυτό οφείλεται κυρίως στο γεγονός πως όταν ανεβαίνουμε μια νότα στο πεντάγραμμο, αυτό δεν αντιστοιχεί σε μετακίνηση ενός τάστου, αφού πολλές νότες μπορούν να παιχτούν σε πολλά σημεία στην κιθάρα. Αυτό το θέμα θα εξεταστεί περαιτέρω, ενώ την ασάφεια λύνουν άλλα είδη συμβολισμού όπως η ταμπλατούρα.

2.1.4 Ταμπλατούρα

Παρόλο που η παρτιτούρα περιγράφει το περιεχόμενο της μουσικής, δεν είναι ιδιαίτερα βοηθητική στην εξήγηση του τρόπου παιξίματος της μουσικής για κάποια όργανα. Η ταμπλατούρα είναι ένα είδος μουσικής σημειογραφίας το οποίο επικεντρώνεται στα έγχορδα όργανα. Διαφέρει από την παρτιτούρα στο γεγονός ότι επικεντρώνεται στην αναπαράσταση στην ταστιέρα του οργάνου των θέσεων στις οποίες πρέπει να τοποθετήσει ο παίχτης τα δάχτυλά του, σε αντίθεση με την παρτιτούρα, η οποία αναπαριστά το τονικό ύψος των νοτών. Στις ταμπλατούρες, το τονικό ύψος εξάγεται έμμεσα. Οι ταμπλατούρες είναι ιδιαίτερα χρήσιμες για την εκμάθηση καθώς και για την γρήγορη κατανόηση ενός κομματιού από τον εκτελεστή.



Σχήμα 2.5: *Παράδειγμα ταμπλατούρας με την αντίστοιχη παρτιτούρα*

Η ταμπλατούρα αποτελείται από παράλληλες οριζόντιες γραμμές, κάθε μία από τις οποίες αντιστοιχεί σε μία χορδή του οργάνου (στην περίπτωση της κιθάρας, υπάρχουν συνήθως έξι χορδές). Το τάστο που πρέπει να παιχτεί σε κάθε χορδή ορίζεται από το αντίστοιχο νούμερο στην οριζόντια γραμμή. Στη περίπτωση που δεν πρέπει να πατηθεί κάποιο τάστο (δηλαδή ανοιχτή χορδή), συμβολίζεται με το 0. Υπάρχει πλήθος άλλων συμβολισμών, μερικοί εκ των οποίων μεταφέρονται άμεσα από τις ταμπλατούρες, καθώς και συμβολισμοί που αποδίδουν

συγκεκριμένες τεχνικές του οργάνου. Για την αναπαράσταση ρυθμού χρησιμοποιείται είτε ξεχωριστός συμβολισμός, είτε εμφανίζεται η ταμπλατούρα σε συνδυασμό με την αντίστοιχη παρτιτούρα, όπως φαίνεται στο σχήμα 2.5.

Στην απλούστερη μορφή τους, οι ταμπλατούρες δεν αναπαριστούν πληροφορία για τον ρυθμό ενός κομματιού. Παρά το υστέρημά τους αυτό, οδηγεί σε ένα μεγάλο πλεονέκτημα, τον εύκολο τρόπο αναπαράστασής τους. Ενώ οι παρτιτούρες απαιτούν γνώση πολύπλοκων προγραμμάτων για τη δημιουργία τους, οι ταμπλατούρες μπορούν να αναπαρασταθούν ως απλό κείμενο ASCII, γεγονός το οποίο οδηγεί στη ευρεία χρήση τους σε σελίδες του διαδικτύου.

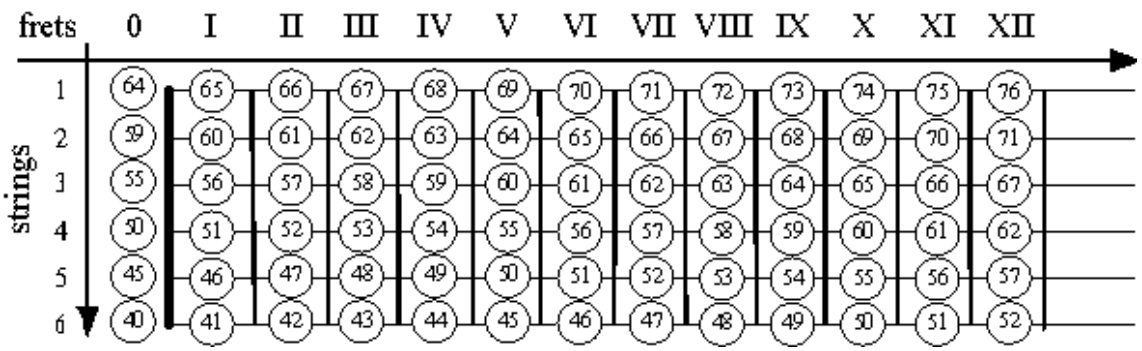
```
e|-----------------------------------2--3--5--|
B|--------------------------------3--5--------|
G|-----------------------------2--4--6---------|
D|-------------------2--4--5--------------------|
A|--2--4--5--------------------------------------|
E|--------------------------------------------|
```

Σχήμα 2.6: *Παράδειγμα ταμπλατούρας σε μορφή ASCII*

2.1.5 Αναπαράσταση MIDI

Τόσο οι ταμπλατούρες, όσο και οι παρτιτούρες χρησιμοποιούνταν από τα χρόνια του μεσαίωνα για την καταγραφή της μουσικής. Παρόλα αυτά, η αναπαράστασή τους δεν είναι άμεσα κατανοητή από τους υπολογιστές. Το πρωτόκολλο MIDI (Musical Instrument Digital Interface) είναι ένα πρωτόκολλο επικοινωνίας μεταξύ ηλεκτρονικών μουσικών οργάνων και του υπολογιστή. Μέσω αυτού δεν μεταδίδεται ηχητικό σήμα, αλλά πληροφορίες για το τονικό ύψος και την ένταση των νοτών, επομένως είναι πολύ χρήσιμο στην μεταγραφή μουσικής και την ηλεκτρονική αναπαράστασή της.

Το πρωτόκολλο MIDI είναι ιδιαίτερο εκτενές, αλλά το τμήμα που μας ενδιαφέρει είναι η αναπαράσταση των νοτών σε αυτό. Οι νότες αναπαρίστανται με διαφορά ημιτονίου, με έναν ακέραιο αριθμό 8 bit στο διάστημα 0 έως 127. Η αντιστοίχηση ξεκινάει από τη νότα C0 συχνότητας 8.18 Hz, έως τη νότα G9 στα 12543 Hz. Η αντιστοίχηση των νοτών αυτών στη κιθάρα μπορεί να γίνει εύκολα και φαίνεται στο σχήμα 2.7.



Σχήμα 2.7: *Οι τιμές MIDI στην ταστιέρα της κιθάρας*

Οι εικόνες 2.2 και 2.7 παρουσιάζουν και το πρόβλημα που αναφέρθηκε σχετικά με τη

μεταγραφή σε ταμπλατούρα, πως δηλαδή ο ίδιος τόνος μπορεί να παιχτεί σε διαφορετικά τάστα στην κιθάρα. Συγκεκριμένα για μια κιθάρα με 6 χορδές και 24 τάστα, στον πίνακα 2.1 φαίνονται οι διαφορετικές αυτές θέσεις.

Διάστημα	E	A	D	G	B	e
E3 - G ♯3	+					
A3 - C ♯4	+	+				
D4 - F ♯4	+	+	+			
G4 - A ♯4	+	+	+	+		
B4 - D ♯5	+	+	+	+	+	
E5 - E 5	+	+	+	+	+	+
F 5 - A 5		+	+	+	+	+
A ♯5 - D 6			+	+	+	+
D ♯6 - G 6				+	+	+
G ♯6 - B 6					+	+
C ♯7 - E 7						+

Πίνακας 2.1: *Πιθανές θέσεις νοτών σε κιθάρα με 6 χορδές και 24 τάστα.*

2.2 Επεξεργασία σήματος

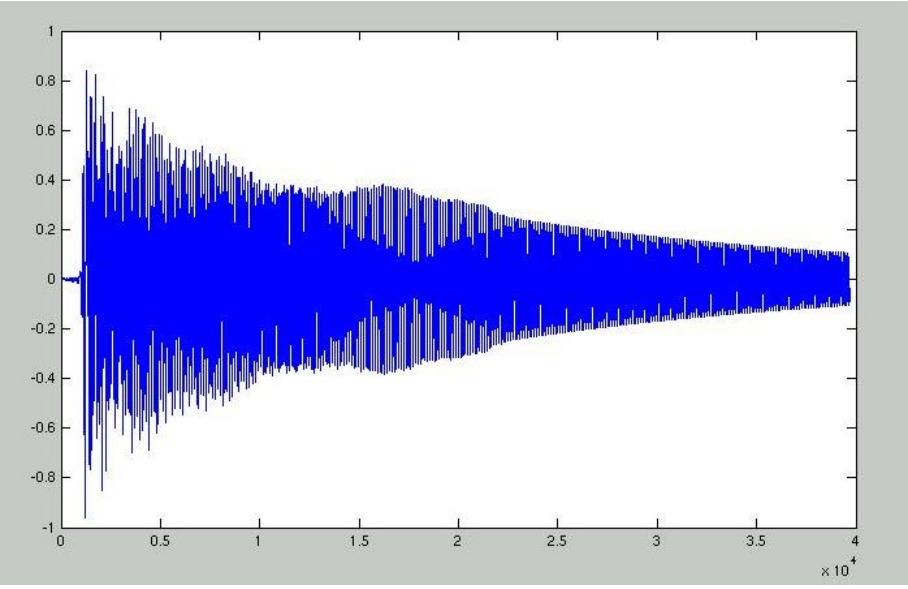
2.2.1 Αναπαράσταση του ήχου

Ο ήχος προκύπτει σαν μεταβολές της πίεσης του αέρα στα τύμπανα του αυτιού. Αυτό ακριβώς καταγράφει και ένα μικρόφωνο ή οποιαδήποτε άλλη αναλογική συσκευή ηχογράφησης. Παρόλα αυτά, οι υπολογιστές χρειάζονται μια ψηφιοποιημένη μορφή αυτής της αναπαράστασης ώστε να εκτελέσουμε την επιθυμητή επεξεργασία σήματος. Αυτό επιτυγχάνεται μέσω της δειγματοληψίας και της κβαντοποίησης της αναλογικής εξόδου του μικροφώνου. Δύο βασικά χαρακτηριστικά των ψηφιακών σημάτων είναι η συχνότητα δειγματοληψίας και το bit-depth.

Η συχνότητα δειγματοληψίας ορίζει το πόσο συχνά λαμβάνονται τα δείγματα από την είσοδο, με πιο συχνή συχνότητα δειγματοληψίας να είναι τα 44.1 kHz (44.100 δείγματα ανά δευτερόλεπτα).

Το bit-depth ορίζει το πλήθος των bit που χρησιμοποιούνται για την αναπαράσταση των τιμών του ψηφιακού σήματος. Όσο περισσότερα bits χρησιμοποιούνται, τόσο πιο ακριβής είναι η αναπαράσταση. Αυξάνοντας το πλήθος των bits αυξάνεται και το εύρος δυναμικών, το οποίο όπως αναφέρθηκε μετριέται σε decibels (dB). Τα πιο διαδεδομένα bit-depth είναι 16-bit και 24-bit.

Παρόλο που είναι απλή, η αναπαράσταση των σημάτων στο πεδίο του χρόνου δεν είναι πάντα επιθυμητή και αποδοτική υπολογιστικά. Μια ιδιαίτερα χρήσιμη αναπαράσταση είναι η αναπαράσταση στο πεδίο της συχνότητας. Με χρήση κατάλληλων μετασχηματισμών, μπορούμε να λάβουμε το φασματικό περιεχόμενο ενός ηχητικού σήματος, δηλαδή τις συχνότητες οι οποίες περιέχονται σε αυτό. Η αναπαράσταση αυτή είναι χρήσιμη επειδή έχει χαρακτηριστικά εικόνας, επομένως μπορεί να δοθεί ως είσοδος σε ένα συνελικτικό νευρωνικό δίκτυο και να λάβουμε πολύ καλά αποτελέσματα.



Σχήμα 2.8: *Κυματομορφή χορδής κιθάρας.*

2.2.2 Μετασχηματισμός Fourier (DFT - FFT)

Μέσω του διακριτού μετασχηματισμού Fourier (Discrete Fourier Transform - DFT), μπορούμε να μετατρέψουμε μια ακολουθία δειγμάτων στον χρόνο, σε μία ακολουθία δειγμάτων στο πεδίο της συχνότητας. Μπορούμε να αναπαραστήσουμε ένα συνεχές σήμα $x(t)$ ως διακριτό σήμα N δειγμάτων ως $x[n] = x(n), n = 0, \ldots, N-1$.

Ο διακριτός μετασχηματισμός Fourier δίνεται από τη σχέση 2.3

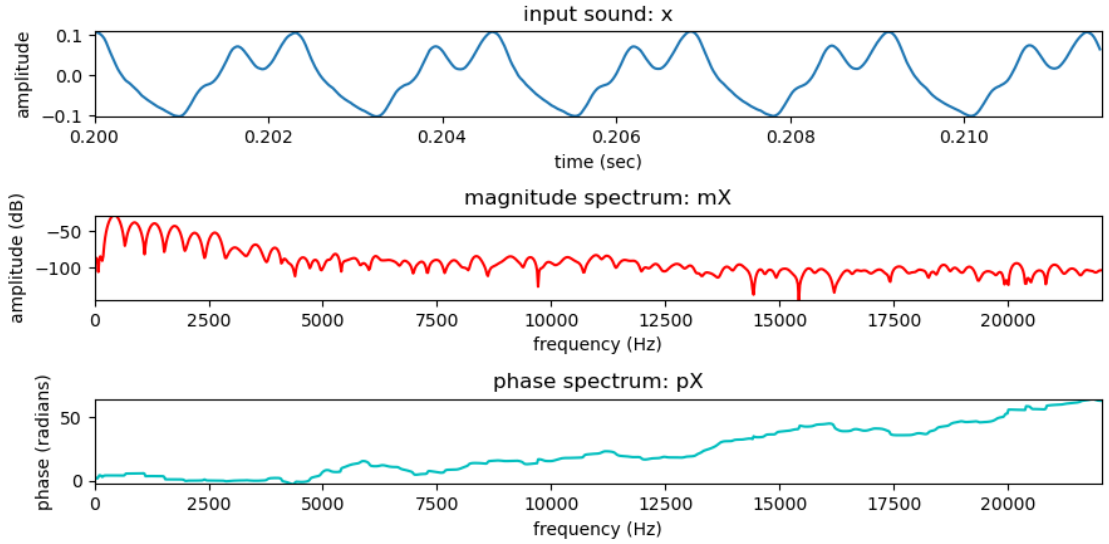
$$\begin{aligned} X[k] &= \sum_{n=0}^{N-1} x[n] e^{-j2\pi kn/N} \\ &= \sum_{n=0}^{N-1} x[n] \left[\cos\left(\frac{2\pi}{N} kn\right) - j \sin\left(\frac{2\pi}{N} kn\right) \right] \end{aligned} \tag{2.3}$$

Στη σχέση 2.3, το αποτέλεσμα $X[k]$ είναι το φασματικό περιεχόμενο του σήματος εισόδου, δηλαδή οι συχνότητες οι οποίες περιέχονται σε αυτό. Η συνάρτηση αυτή είναι μιγαδική, οπότε μπορεί να αναπαρασταθεί σε πολικές συντεταγμένες μέτρου και φάσης (γενικότερα ενδιαφερόμαστε για το μέτρο, που μας δείχνει τις συχνότητες του σήματος). Ένα παράδειγμα μετασχηματισμού DFT για μια νότα σε ένα όργανο φαίνεται στο σχήμα 2.9.

Ο μετασχηματισμός DFT είναι αναστρέψιμος, και μπορούμε να λάβουμε το αρχικό σήμα από το φάσμα μέσω του αντίστροφου μετασχηματισμού DFT (Inverse DFT - IDFT) που παρουσιάζεται στη σχέση 2.4.

$$x[n] = \frac{1}{N} \sum_{n=0}^{N-1} X[k] e^{-j2\pi kn/N} \tag{2.4}$$

Ο υπολογισμός του DFT είναι συχνά πολύ υπολογιστικά απαιτητικός, με τις απλές υλοποιήσεις να έχουν πολυπλοκότητα $O(N^2)$, το οποίο δεν είναι πρακτικό για μεγάλο πλήθος εφαρμογών. Τη λύση σε αυτό το πρόβλημα δίνει ο αλγόριθμος Fast Fourier Transform (FFT), ο οποίος υπό περιορισμούς μπορεί να υπολογίσει το DFT με πολυπλοκότητα $O(N \log n)$ πε-



Σχήμα 2.9: *Μετασχηματισμός DFT σε μια νότα φλάουτου.*

τυχαίνοντας σημαντική βελτίωση από τον απλό DFT.

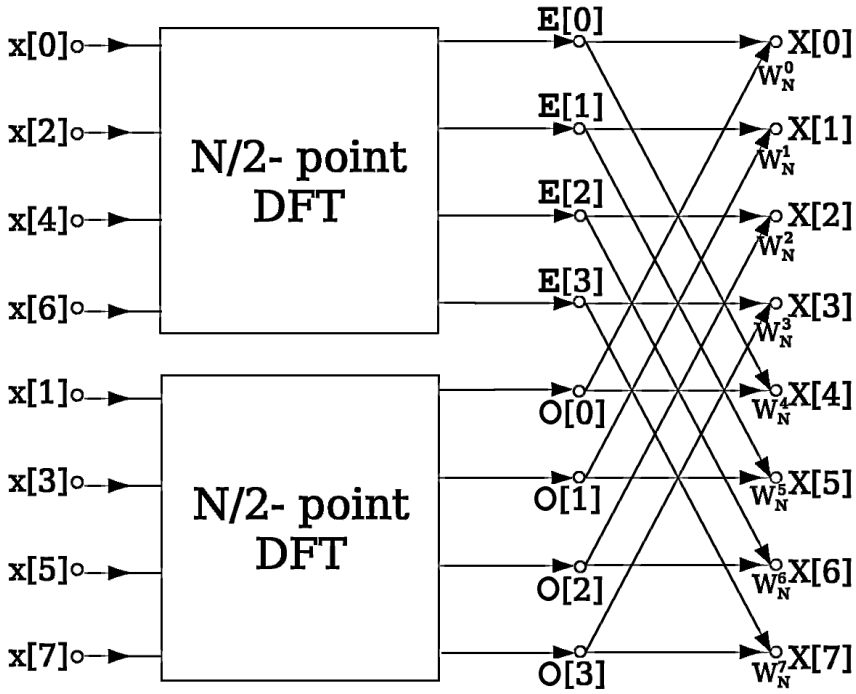
Υπάρχουν διάφορες υλοποιήσεις του Fast Fourier Transform, αλλά η πιο διαδεδομένη είναι ο αλγόριθμος Cooley–Tukey FFT. Για την υλοποίηση αυτού του αλγορίθμου, γίνεται η υπόθεση πως τα δείγματα του σήματος εισόδου N είναι δυνάμεις του 2. Η βασική ιδέα του αλγορίθμου βασίζεται στη τεχνική ¨διαίρει και βασίλευε¨, όπου λαμβάνοντας υπόψιν συμμετρίες του σήματος, σπάμε τον αρχικό υπολογισμό DFT N δειγμάτων, στον υπολογισμό δύο ξεχωριστών DFT $N/2$ δειγμάτων. Η υλοποίηση φαίνεται σχηματικά στο σχήμα 2.10. Εφαρμόζοντας αναδρομικά τη διαίρεση αυτή, επιτυγχάνεται η πολυπλοκότητα $O(N \log n)$.

2.2.3 Γραμμικό Φασματογράφημα - Short-time Fourier Transform

Μέχρι στιγμής ασχοληθήκαμε με τρόπους υπολογισμού φασματικού περιεχομένου σημάτων τα οποία δεν αλλάζουν στο χρόνο. Τα πραγματικά σήματα όμως δεν είναι τέτοια, και δε μπορούν να αναπαρασταθούν με ένα μόνο φάσμα συχνοτήτων. Για την αναπαράσταση του φάσματος πραγματικών σημάτων χρησιμοποιείται ο μετασχηματισμός Short-time Fourier Transform (STFT). Ο μετασχηματισμός αυτός διαιρεί το σήμα σε μικρότερα τμήματα και υπολογίζει το μετασχηματισμό Fourier στο καθένα από αυτά. Ο μετασχηματισμός STFT ενός διακριτού σήματος $x[n]$ υπολογίζεται με βάση τη σχέση 2.5.

$$X_l[k] = \sum_{n=-N/2}^{N/2-1} w_[n]x[n + lH]e^{-j2\pi kn/N} \tag{2.5}$$

Στη σχέση 2.5, k είναι οι ¨κάδοι συχνοτήτων¨ (frequency bins), δηλαδή οι διακριτές συχνότητες οι οποίες υπολογίζονται κατά τον μετασχηματισμό, Η το μέγεθος του άλματος (hop size) μεταξύ των διαδοχικών μετασχηματισμών Fourier. Τα frequency bins μπορούν να αντιστοιχιστούν σε πραγματικές συχνότητες με βάση το ρυθμό δειγματοληψίας f_s με βάση τη σχέση 2.6.



Σχήμα 2.10: *Παράδειγμα χρήσης συμμετριών για απλοποίηση υπολογισμών στον αλγόριθμο Cooley-Tukey FFT.*

$$f(k) = k\frac{f_s}{N} \tag{2.6}$$

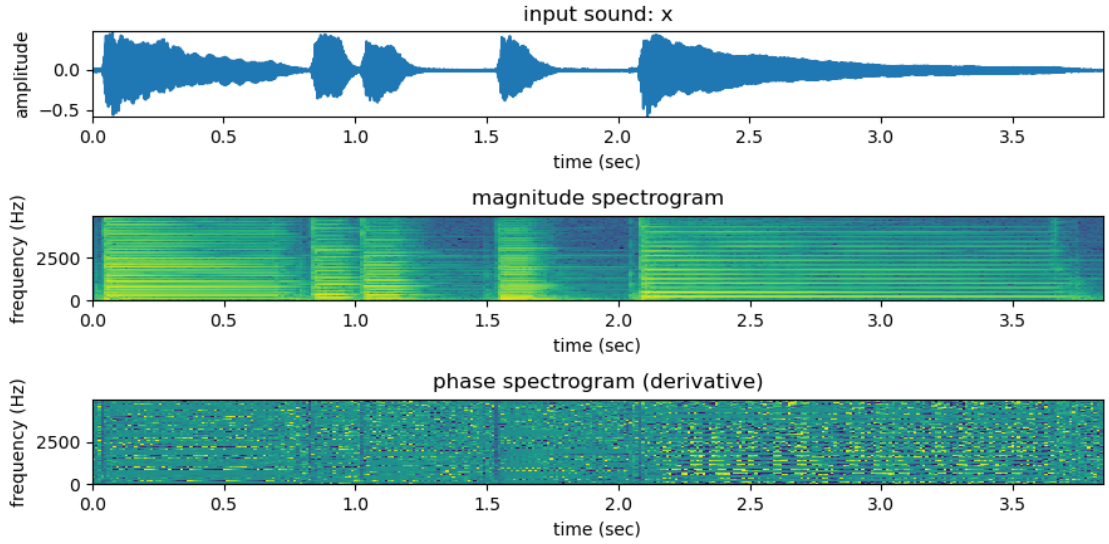
Ο όρος $w[n]$ αφορά μια συνάρτηση παραθύρου (window function). Οι συναρτήσεις παραθύρου είναι συναρτήσεις οι οποίες έχουν μηδενικές τιμές σε όλο το εύρος τους εκτός από ένα διάστημα, συνήθως κεντραρισμένο στην αρχή των αξόνων και συμμετρικό, και χρησιμοποιείται για να αποσπάσουμε ομαλά τα τμήματα της εισόδου που θα γίνει ο μετασχηματισμός Fourier. Συχνά χρησιμοποιούμενες συναρτήσεις παραθύρου είναι οι blackman, hamming, hanning. Κάθε

Από τον μετασχηματισμό αυτό προκύπτει το Φασματογράφημα (Spectrogram) ενός σήματος, το οποίο αποτελεί τη γραφική αναπαράσταση του τετραγώνου του πλάτους του STFT. Μέσω αυτού, μπορούμε να δούμε πως οι συχνότητες του σήματος αλλάζουν κατά τη διάρκεια του χρόνου. Ένα παράδειγμα φασματογραφηματος φαίνεται στην εικόνα 2.11.

2.2.4 Φασματογράφημα Λογαριθμικής Κλίμακας

Η κλίμακα του φασματογραφήματος δε χρειάζεται απαραίτητα να είναι γραμμική. Διάφορες έρευνες ([8], [9]) έχουν δείξει πως οι άνθρωποι αντιλαμβάνονται τις διαφορετικές συχνότητες όχι με γραμμικό, αλλά με λογαριθμικό τρόπο. Για παράδειγμα, μπορούμε να αντιληφθούμε εύκολα διαφορές μεταξύ 500 Hz και 600 Hz, αλλά δύσκολα αντιλαμβανόμαστε τη διαφορά μεταξύ 10.000 Hz και 10.100 Hz. Λόγω αυτού, είναι χρήσιμο να χρησιμοποιούμε λογαριθμικές κλίμακες για την αναπαράσταση του φασματογραφηματος.

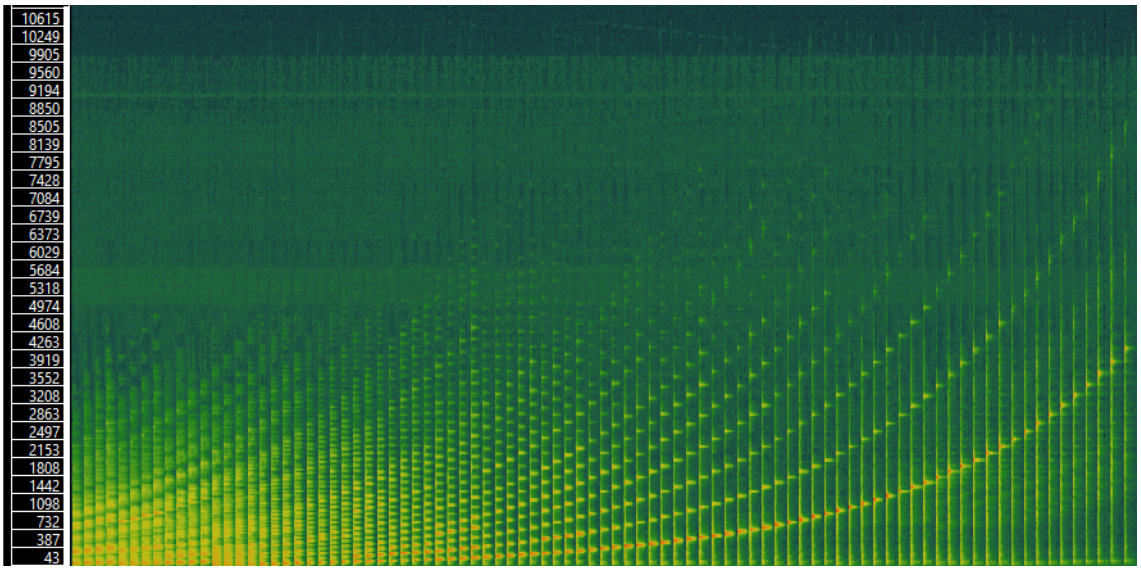
Η βασική ιδέα της χρήσης λογαριθμικής κλίμακας στη μουσική, αφορά στον τρόπο που



Σχήμα 2.11: *Γραμμικό Φασματογράφημα σύντομης μελωδίας πιάνου.*

ορίζονται οι νότες στο 12-tone equal temperament. Συγκεκριμένα, με βάση τη νότα A4, η οποία αποτελεί τη MIDI νότα $p = 69$ στα 440Hz, οι κεντρικές συχνότητες των υπόλοιπων νοτών ορίζονται από τη σχέση 2.7.

$$F_{\text{pitch}}(p) = 2^{(p-69)/12} * 440 \tag{2.7}$$



Σχήμα 2.12: *Γραμμικό Φασματογράφημα χρωματικής κλίμακας παιγμένης στο πιάνο.*

Στην εικόνα 2.12 φαίνεται το γραμμικό φασματογράφημα συνεχόμενων νοτών της χρωματικής κλίμακας παιγμένες σε πιάνο (δηλαδή νότες με διαφορά ημιτονίου). Στο φασματογράφημα αυτό φαίνεται ξεκάθαρα η εκθετική εξάρτηση των θεμελιωδών συχνοτήτων των νοτών. Για να αποκτήσουμε αυτή την αναπαράσταση στον άξονα της συχνότητας, αναθέτουμε σε κάθε τιμή $X_l[k]$ τον τόνο που είναι πλησιέστερα στη πραγματική συχνότητα f(k) της σχέσης 2.6 που αντιστοιχεί κάποιου bin. Συγκεκριμένα, για κάθε τόνο $p \in [0, 127]$ ορίζουμε

το σύνολο $P(p)$ με βάση τη σχέση 2.8.

$$P(p) := \{k : F_{\text{pitch}}(p - 0.5) \leq f(k) \leq F_{\text{pitch}}(p + 0.5)\} \tag{2.8}$$

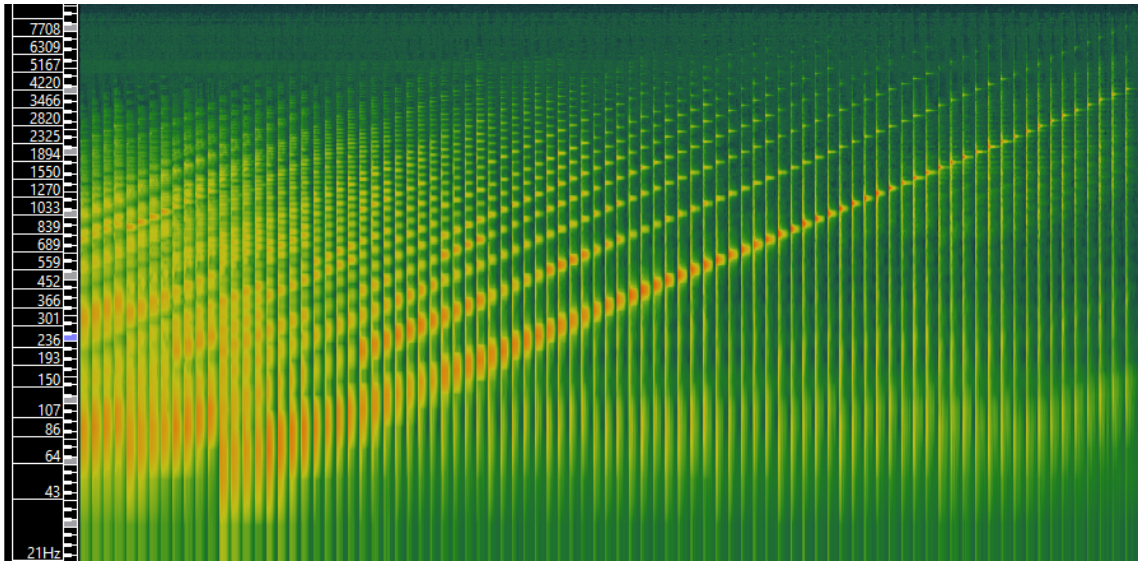
Το εύρος συχνοτήτων αυτό είναι λογαριθμικό, ενώ μπορούμε να ορίσουμε το εύρος ζώνης $BW(p)$ για κάθε τόνο με βάση τη σχέση 2.9. Το εύρος ζώνης μεγαλώνει όσο αυξάνονται οι συχνότητες (συγκεκριμένα διπλασιάζεται καθώς ανεβαίνουμε μία οκτάβα).

$$BW(p) = F_{\text{pitch}}(p + 0.5) - F_{\text{pitch}}(p - 0.5) \tag{2.9}$$

Με βάση αυτό το σύνολο bins μπορούμε να ορίσουμε το Log-Frequency Spectrogram με βάση τη σχέση 2.10.

$$\text{spec}(l, p) = \sum_{k \in P(p)} |X_l(k)|^2 \tag{2.10}$$

Το αντίστοιχο φασματογράφημα που προκύπτει για το αρχείο ήχου με νότες με διαφορά ημιτονίου φαίνεται στο σχήμα 2.13, στο οποίο βλέπουμε τη γραμμική σχέση μεταξύ των τόνων. Με τη βοήθεια των λογαριθμικών φασματογραφημάτων είναι πιο εύκολη η κατανόηση των αρμονικών χαρακτηριστικών των ηχητικών σημάτων, αφού η εικόνα είναι πιο ξεκάθαρη και δεν υπάρχει μεγάλος συνωστισμός στις χαμηλές συχνότητες.

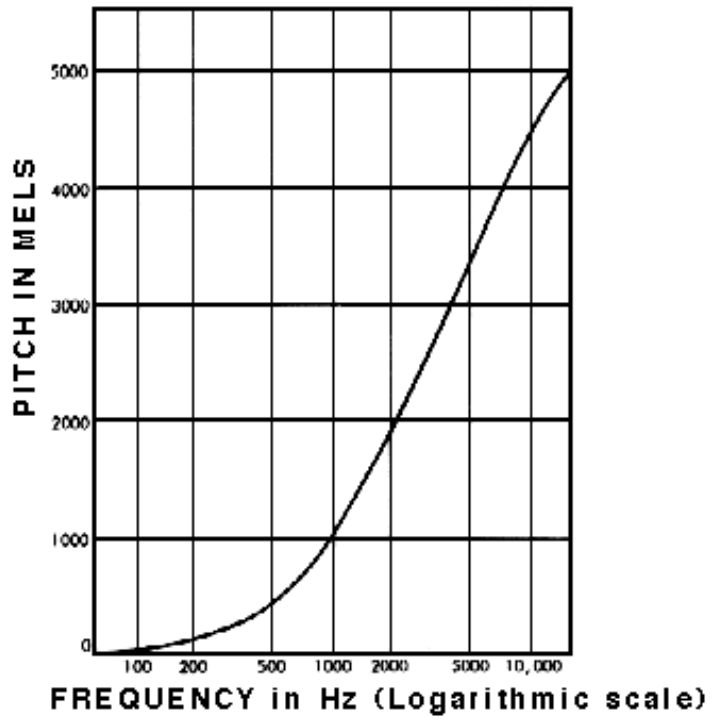


Σχήμα 2.13: *Λογαριθμικό Φασματογράφημα χρωματικής κλίμακας παιγμένης στο πιάνο.*

2.2.5 Φασματογράφημα Mel

Όπως αναφέρθηκε στην προηγούμενη ενότητα, οι άνθρωποι δεν αντιλαμβάνονται τα τονικά ύψη των νοτών με γραμμικό τρόπο. Παρόλο που η λογαριθμική προσέγγιση είναι καλύτερη, δεν είναι και η μοναδική. Στη δημοσίευση [9], οι Stevens, Volkmann, και Newmann προτείνουν μια κλίμακα που βασίζεται στη διαίσθηση των ανθρώπων, στην οποία ίσες αποστάσεις ακούγονται ίσες στους ακροατές, η οποία ονομάζεται κλίμακα Mel. Η αντιστοιχία

με τις πραγματικές συχνότητες γίνεται ορίζοντας την διαισθητική συχνότητα 1000 mels στη συχνότητα των 1000 Hz. Η κλίμακα που προτάθηκε στο [9] φαίνεται στο σχήμα 2.14.

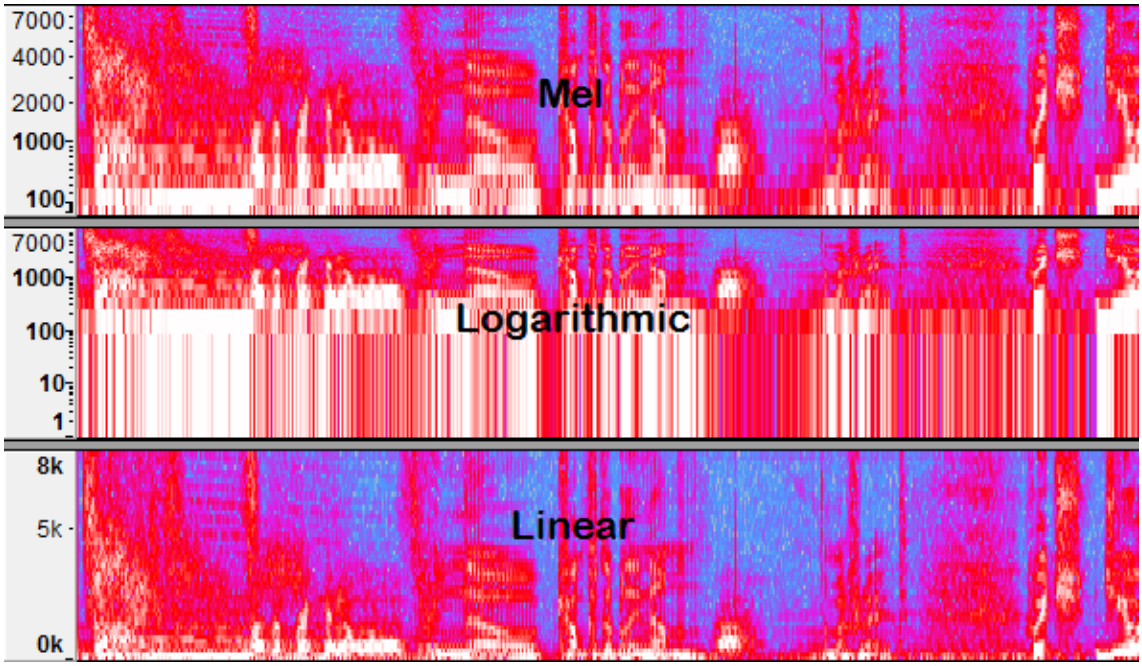


Σχήμα 2.14: *Η κλίμακα mel.*

Δεν υπάρχει μόνο μία κλίμακα mel, αλλά ένας γενικά αποδεκτός τρόπος μετατροπής από mel σε Hz γίνεται με τη σχέση 2.11.

$$m = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \tag{2.11}$$

Στην εικόνα 2.15 φαίνεται η σύγκριση μεταξύ των τριών φασματογραφημάτων (Mel, Log, Linear) για το ίδιο ηχητικό σήμα. Παρατηρούμε πως υπάρχει καλύτερη ανάλυση στις χαμηλές συχνότητες, χωρίς αυτές να λαμβάνουν πολύ μεγάλο κομμάτι του φασματογράμματος όπως στην περίπτωση του λογαριθμικού φασματογράμματος. Έχει παρατηρηθεί ότι η χρήση των φασματογραφημάτων Mel οδηγεί σε καλύτερη απόδοση των συνελικτικών δικτύων [8], οπότε αυτά θα χρησιμοποιηθούν στις υλοποιήσεις.



Σχήμα 2.15: *Σύγκριση φασματογραφημάτων Mel, Log, Linear για την ίδια ηχητική είσοδο.*

Κεφάλαιο 3

Μηχανική Μάθηση - Νευρωνικά Δίκτυα

Στο κεφάλαιο αυτό θα εξετάσουμε κάποιες βασικές έννοιες της μηχανικής μάθησης, καθώς και των νευρωνικών δικτύων. Η υλοποίηση των μοντέλων απαιτεί τη χρήση συγκεκριμένων δομών δικτύων, οπότε είναι χρήσιμο να γνωρίζουμε τις βασικές αρχές λειτουργίας τους.

3.1 Ιστορική Αναδρομή

Η ανάπτυξη των νευρωνικών δικτύων βασίστηκε στην μοντελοποίηση του τρόπου με τον οποίο ο ανθρώπινος εγκέφαλος εκτελεί υπολογισμούς. Το 1943 ο νευροφυσιολόγος Warren McCulloch μαζί με τον μαθηματικό Walter Pitts μοντελοποίησαν το με ένα απλό ηλεκτρικό κύκλωμα την παραπάνω ιδέα. Το 1949, στο βιβλίο του ¨The Organization of Behaviour¨, ο Donald Hebb έδειξε πως οι διαδρομές μεταξύ των νευρώνων ενισχύονται κάθε φορά που αυτοί χρησιμοποιούνται, το οποίο αποτελεί θεμέλιο της ανθρώπινης μάθησης.

Με την αύξηση της υπολογιστικής δύναμης την δεκαετία του 1950, ξεκίνησαν τα πρώτα βήματα στην υλοποίηση νευρωνικών δικτύων. Το 1958, προτάθηκε από τον ψυχολόγο Frank Rosenblatt η ιδέα του Perceptron. Το 1959, οι ερευνητές του Stanford Bernard Widrow και Marcian Hoff, ανέπτυξαν τα πρώτα νευρωνικά δίκτυα που εφαρμόσθηκαν στην επίλυση πραγματικών προβλημάτων. Τα συστήματα αυτά ονομάστηκαν ADALINE και MADALINE και χρησιμοποιούνται ακόμα και σήμερα για την απομάκρυνση θορύβου από τις τηλεφωνικές γραμμές.

Παρόλο το θεωρητικό υπόβαθρο, η τεχνολογία της εποχής δεν επέτρεψε σε περαιτέρω ανάπτυξη νευρωνικών δικτύων, κυρίως λόγω του μεγάλου υπολογιστικού όγκου. Αυτό είχε ως αποτέλεσμα τη δεκαετία του 1960 και του 1970, η πρόοδος στον τομέα των νευρωνικών δικτύων να μείνει στάσιμη. Σημαντικό αντίκτυπο είχε και η δημοσίευση ¨Perceptrons" από τον Marvin Minsky, η οποία ανέδειξε διάφορα προβλήματα στην πρακτική υλοποίηση ενός νευρωνικού δικτύου εκείνη την εποχή, οδηγώντας σε ακόμα μικρότερο ενδιαφέρον σε αυτά. Παρόλα αυτά, το πρώτο πολυεπίπεδο δίκτυο αναπτύχθηκε το 1975.

Την κατάσταση αυτή διατάραξε η ανάπτυξη του δικτύου ¨Hopfield Net" το 1982 από τον John Hopfield. Παράλληλα με αυτό, η Ιαπωνία ανακοίνωση την ίδια χρονιά πως θα προχωρήσει στην ανάπτυξη νέων νευρωνικών δικτύων, προσδίδοντας αναζωογόνηση στον κλάδο. Ένα από τα σημαντικότερα βήματα στην ιστορία των νευρωνικών δικτύων, ήταν η εφαρμογή των αλγορίθμων backpropagation και gradient descent στην διαδικασία εκμάθησης το

1985, τους οποίους θα εξετάσουμε στη συνέχεια. Με αυτό τον τρόπο, τα νευρωνικά δίκτυα εφαρμόζονται σε τεράστιο πλήθος εφαρμογών μέχρι και σήμερα.

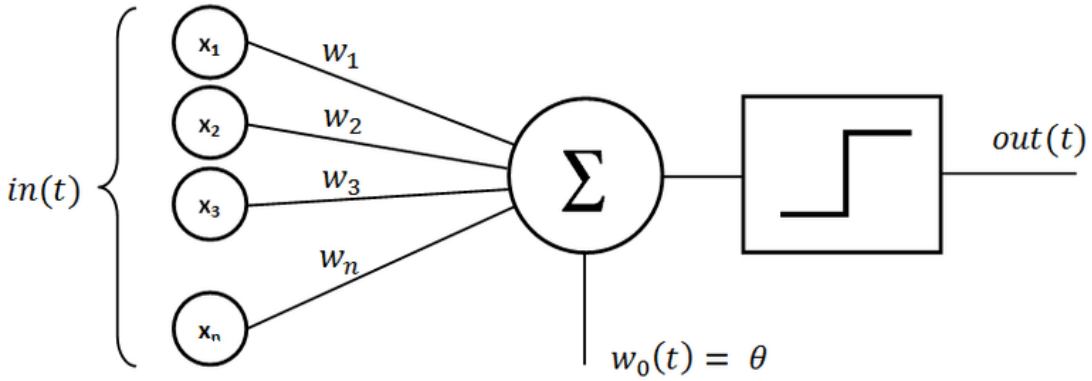
3.2 Feedforward Neural Networks

Οι δύο μεγάλες κατηγορίες νευρωνικών δικτύων είναι τα δίκτυα πρόσθιας τροφοδότησης (feedforward) και τα αναδρομικά δίκτυα. Θα ξεκινήσουμε εξετάζοντας την πρώτη κατηγορία, αφού ήταν και η πρώτη που αναπτύχθηκε ιστορικά. Στα feedforward δίκτυα, η πληροφορία μεταβιβάζεται προς μία κατεύθυνση και δεν υπάρχουν κύκλοι στις συνδέσεις των νευρώνων.

3.2.1 Δίκτυο Single-layer Perceptron

Τα τεχνητά νευρωνικά δίκτυα είναι υπολογιστικά μοντέλα, τα οποία βασίζονται στη λειτουργία των νευρώνων στον ανθρώπινο εγκέφαλο. Αποτελούνται από διασυνδεδεμένους νευρώνες, οι οποίοι ¨μαθαίνουν" μέσω των δεδομένων εισόδου και εξόδου, χωρίς να απαιτείται περαιτέρω προγραμματισμός.

Το πιο απλό παράδειγμα νευρωνικού δικτύου είναι ο απλός νευρώνας, του οποίου η αρχιτεκτονική φαίνεται στο σχήμα 3.1.



Σχήμα 3.1: *Αρχιτεκτονική δικτύου Perceptron*

Αποτελείται από τις εισόδους, ένα n-διάστατο διάνυσμα x, το οποίο στη συνέχεια πολλαπλασιάζεται (εσωτερικό γινόμενο) με ένα n-διάστατο διάνυσμα w, το διάνυσμα των βαρών, ενώ προσθέτουμε στο άθροισμα αυτών την πόλωση (bias) b. Η τελική έξοδος του νευρώνα προκύπτει περνώντας το προηγούμενο αποτέλεσμα μέσα από μια συνάρτηση ενεργοποίησης φ. Συνολικά, η έξοδος του νευρωνικού δίνεται από την σχέση 3.1

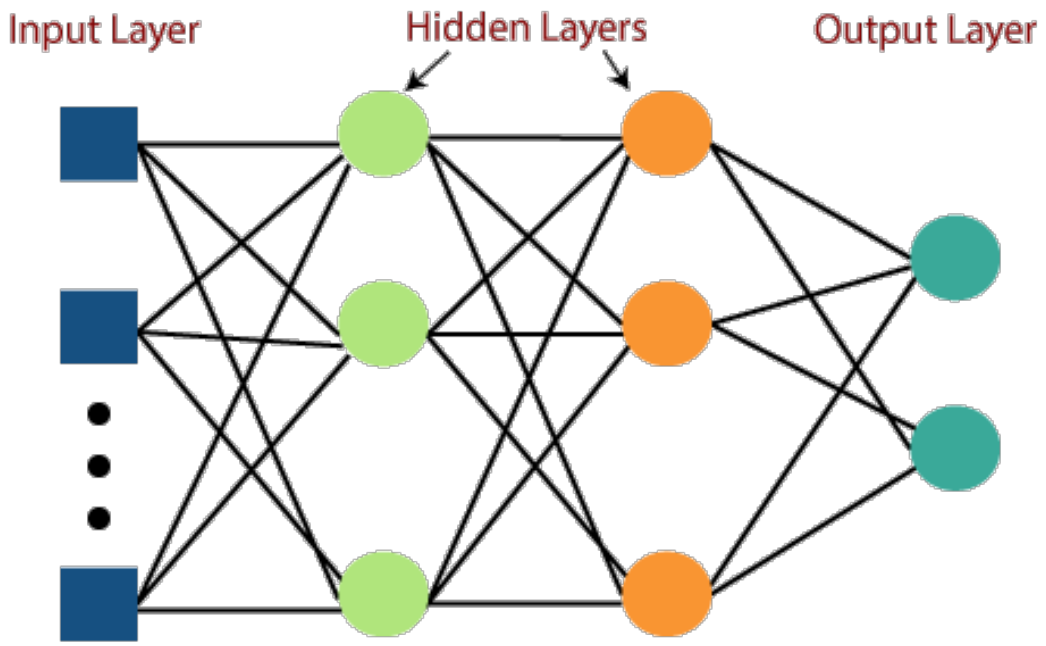
$$y = \varphi\left(\sum_{i=1}^{n} x_i * w_i + b\right) \tag{3.1}$$

3.2.2 Δίκτυο Multi-layer Perceptron

Παρά την απλή αρχιτεκτονική του, το Single-layer Perceptron έχει περιορισμένες εφαρμογές, αφού έχει τη δυνατότητα να αναγνωρίσει μόνο γραμμικά διαχωρίσιμα προβλήματα.

Όπως ενέδειξε ο Marvin Minsky στη δημοσίευση "Perceptrons", το Single-layer Perceptron δεν μπορεί να αναγνωρίσει τη συνάρτηση "XOR". Τη λύση σε τέτοια προβλήματα δίνουν τα πολυεπίπεδα Perceptrons (MLPs).

Ένα MLP αποτελείται από τουλάχιστον τρία επίπεδα κόμβων: ένα επίπεδο εισόδου, ένα ή περισσότερα κρυφά επίπεδα και ένα επίπεδο εξόδου. Τα ενδιάμεσα επίπεδα είναι αυτά που επιτελούν τις μη γραμμικές απεικονίσεις των δεδομένων.



Σχήμα 3.2: *Αρχιτεκτονική δικτύου MLP*

Η ροή της πληροφορίας από το επίπεδο i στο επίπεδο i+1 περιγράφεται από τη σχέση 3.2.

$$\boldsymbol{y}^{(i+1)} = \phi^{(i)}\left(\boldsymbol{W} * \boldsymbol{y}^{(i)} + \boldsymbol{b}\right) \tag{3.2}$$

Στην σχέση 3.2, $\boldsymbol{y}^{(i)}$ είναι η έξοδος του i-οστού επιπέδου, $\boldsymbol{y}^{(i+1)}$ είναι η έξοδος του i+1 επιπέδου, $\boldsymbol{W}$ ο πίνακας των βαρών και $\phi^{(i)}$ η συνάρτηση ενεργοποίησης.

3.2.3 Αλγόριθμος (Backpropagation)

Ο αλγόριθμος backpropagation αποτελεί ένα θεμελιώδη τρόπο με τον οποίο εκτελείται η επιβλεπόμενη μάθηση στα perceptron. Βασίζεται στον κανόνα της αλυσίδας, προσαρμόζει τα βάρη στις συνδέσεις των νευρώνων, ξεκινώντας από την έξοδο, με κατεύθυνση προς το επίπεδο εισόδου. Όπως αναφέρθηκε, σε ένα επίπεδο j του νευρωνικού δικτύου, το σήμα που εμφανίζεται στην έξοδο του νευρώνα κατά την επανάληψη n ορίζεται από τη σχέση 3.3

$$y_j(n) = \phi_j\left(v_j(n)\right) = \phi_j\left(\sum_{i=0}^{m} w_{ji}(n) y_i(n)\right) \tag{3.3}$$

Ο αλγόριθμος backpropagation βασίζεται στην εφαρμογή διόρθωσης $\Delta w_{ji}(n)$ στο βάρος της σύνδεσης w_{ji}, ανάλογο με τη μερική παράγωγο του σφάλματος στον νευρώνα $\partial C(n)/\partial w_{ji}(n)$,

το οποίο αποτελεί και το συντελεστή ευαισθησίας. Εφαρμόζοντας τον κανόνα της αλυσίδας, αυτό μπορεί να εκφραστεί ως

$$\frac{\partial C(n)}{\partial w_{ji}(n)} = \frac{\partial C_j(n)}{\partial e_j(n)} \frac{\partial e_j(n)}{\partial y_i(n)} \frac{\partial y_i(n)}{\partial v_j(n)} \frac{\partial v_j(n)}{\partial w_{ji}(n)} \tag{3.4}$$

Το σήμα σφάλματος $e_j(n)$ που παράγεται στην έξοδο του νευρώνα j εκφράζει τη διαφορά της εξόδου του νευρώνα $y_j(n)$ από την επιθυμητή τιμή $d_j(n)$, επομένως

$$e_j(n) = d_j(n) - y_j(n) \qquad \frac{\partial e_j(n)}{\partial y_i(n)} = -1 \tag{3.5}$$

Η στιγμιαία ενέργεια σφάλματος στον νευρώνα j του δικτύου εκφράζεται από τη σχέση

$$C_j(n) = \frac{1}{2} e_j^2(n) \qquad \frac{\partial C_j(n)}{\partial e_j(n)} = e_j(n) \tag{3.6}$$

Διαφορίζοντας τη σχέση για την έξοδο του νευρώνα προκύπτει

$$\frac{\partial y_i(n)}{\partial v_j(n)} = \varphi_j' \left(v_j(n) \right) \tag{3.7}$$

Τέλος, διαφορίζοντας την αρχική σχέση για την έξοδο της συνάρτησης ενεργοποίησης προκύπτει

$$\frac{\partial v_j(n)}{\partial w_{ji}(n)} = y_i(n) \tag{3.8}$$

Συνολικά από τις σχέσεις 3.3 - 3.8, μπορούμε να εκφράσουμε το συντελεστή ευαισθησίας ως

$$\frac{\partial C(n)}{\partial w_{ji}(n)} = -e_j(n) \varphi_j' \left(v_j(n) \right) y_i(n) \tag{3.9}$$

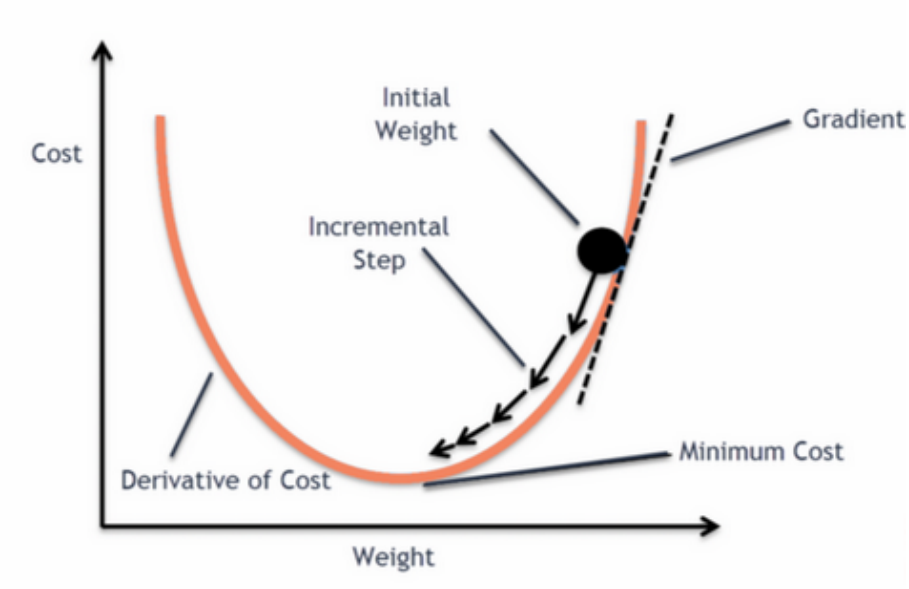
Με βάση τον συντελεστή αυτόν, εφαρμόζουμε τη διόρθωση ΔW_{ji} στα βάρη των συνδέσεων με βάση την παράμετρο ρυθμού μάθησης η. Με αυτό τον τρόπο, εκτελούμε βαθμωτή κατάβαση (gradient descent) στο χώρο των βαρών, αναζητώντας την τιμή των βαρών που ελαχιστοποιεί τη συνολική ενέργεια σφάλματος.

$$\Delta W_{ji} = -\eta \frac{\partial C(n)}{\partial w_{ji}(n)} = \eta \delta_j(n) y_i(n) = \eta e_j(n) \varphi_j' \left(v_j(n) \right) \tag{3.10}$$

Ο όρος $\delta_j(n)$ εκφράζει την τοπική κλίση του νευρώνα εξόδου και ισούται με το γινόμενο του σήματος σφάλματος $e_j(n)$ και την παράγωγο συνάρτησης ενεργοποίησης του νευρώνα $\varphi_j' \left(v_j(n) \right)$. Επομένως, το σφάλμα στην έξοδο του νευρώνα παίζει κρίσιμο ρόλο στην προσαρμογή των βαρών. Διακρίνουμε 2 περιπτώσεις για τους τύπους νευρώνων του δικτύου.

Περίπτωση 1ή: Κόμβος Εξόδου : Σε αυτή την περίπτωση διαθέτουμε την επιθυμητή απόκριση, επομένως το σφάλμα $e_j(n)$ υπολογίζεται από τη σχέση 3.5.

Περίπτωση 2ή: Κρυφός Κόμβος : Σε αυτή την περίπτωση δεν υπάρχει διαθέσιμη η επιθυμητή απόκριση, οπότε το σφάλμα στον κάθε νευρώνα υπολογίζεται αναδρομικά, με βάση τους νευρώνες που συνδέονται άμεσα με αυτόν. Για να το πετύχουμε αυτό, θεωρούμε πως το συνολικό σφάλμα προκύπτει από το άθροισμα των τετραγώνων των νευρώνων εξόδου $e_k(n)$. Η τοπική κλίση του νευρώνα j εκφράζεται από τη σχέση 3.11



Σχήμα 3.3: *Γραφική Αναπαράσταση Βαθμωτής Κατάβασης*

$$\delta_j(n) = \phi'_j\left(v_j(n)\right) \sum_k \delta_k(n) w_{kj}(n) \tag{3.11}$$

Ο πρώτος όρος $\phi'_j\left(v_j(n)\right)$ εξαρτάται από τη συνάρτηση ενεργοποίησης του νευρώνα j. Ο δεύτερος παράγοντας $\sum_k \delta_k(n) w_{kj}(n)$, εξαρτάται από το δ_k το οποίο αφορά τους κόμβους του ακριβώς δεξιότερου επιπέδου του δικτύου που συνδέονται με τον νευρώνα j, και το w_{kj}, το οποίο αφορά τις συνδέσεις μεταξύ του νευρώνα j και των νευρώνων κ. Συνολικά ο αλγόριθμος backpropagation περιγράφεται από τα ακόλουθα βήματα.

Βήμα 1 - Αρχικοποίηση. Αρχικοποίηση των βαρών των συνδέσεων με ομοιόμορφη κατανομή μηδενικού μέσου όρου.

Βήμα 2 - Παραδείγματα εκπαίδευσης. Παρουσιάζονται στο δίκτυο μια εποχή παραδειγμάτων εκπαίδευσης, με σκοπό στη συνέχεια να αλλάξει το βάρος των συνδέσεων με βάση αυτά.

Βήμα 3 - Υπολογισμός προς τα εμπρός. Έστω ένα δείγμα εκπαίδευσης $(\mathbf{x}(n), \mathbf{d}(n))$, όπου $\mathbf{x}(n)$ η είσοδος στο δίκτυο και $\mathbf{d}(n)$ η επιθυμητή απόκριση. Υπολογίζουμε τα σήματα εξόδου για κάθε νευρώνα j και το σήμα σφάλματος.

$$v_j^l(n) = \sum_i w_{ji}^l(n) y_i^{l-1}(n) \qquad e_j(n) = d_j(n) - y_j^L(n) \tag{3.12}$$

Βήμα 4 - Υπολογισμός προς τα πίσω. Υπολογισμός των τοπικών κλίσεων με βάση τους τύπους που προέκυψαν στην προηγούμενη ανάλυση.

$$\delta_j^l(n) = \begin{cases} e_j^L(n)\phi'_j\left(v_j(n)^L\right), & \text{νευρώνας επίπεδου εξόδου.} \\ \phi'_j\left(v_j(n)^l\right) \sum_k \delta_k^{l+1}(n) w_{kj}^{l+1}(n), & \text{νευρώνας κρυφού επιπέδου.} \end{cases} \tag{3.13}$$

Στη συνέχεια ακολουθεί η αλλαγή των βαρών του δικτύου με βάση τον κανόνα κατάβασης

κλίσης και τη παράμετρο ρυθμού μάθησης η.

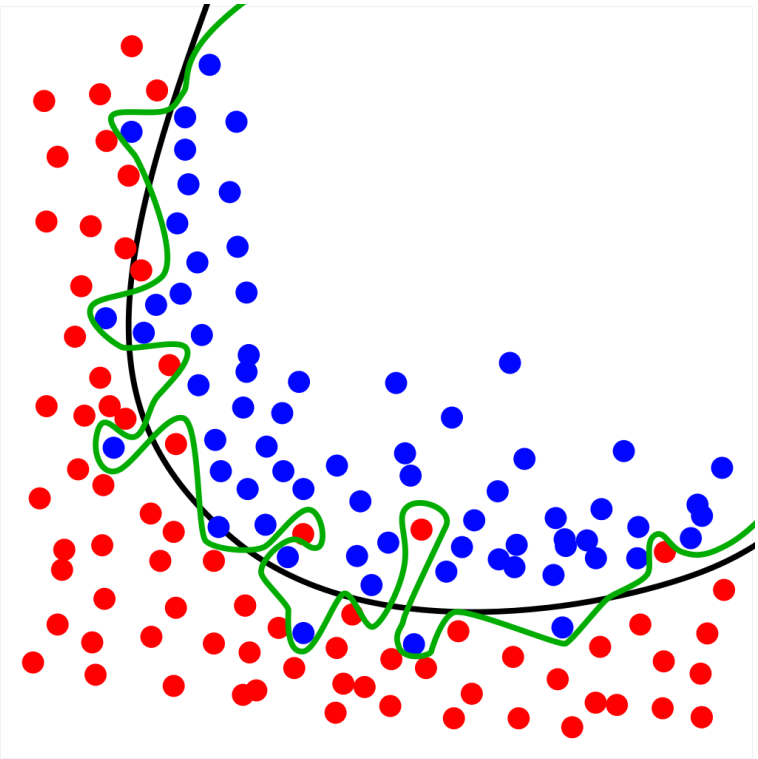
$$w_{ji}^{l}(n+1) = w_{ji}^{l}(n) + \eta\delta_{j}^{l}(n)y_{i}^{l-1}(n) \tag{3.14}$$

Βήμα 5 - Επανάληψη. Επαναλαμβάνονται τα 3 τελευταία βήματα έως ότου να ικανοποιηθεί ένα κριτήριο τερματισμού.

3.2.4 Overfitting

Κατά τη διαδικασία της επιβλεπόμενης μάθησης, είναι πλήρως διαδεδομένο να γίνεται διαχωρισμός του dataset σε τρία διαφορετικά dataset, τα οποία επιτελούν το καθένα ξεχωριστό σκοπό. Το training dataset είναι το βασικότερο και χρησιμοποιείται αποκλειστικά για την εκπαίδευση του δικτύου και την προσαρμογή των βαρών. Το test dataset χρησιμοποιείται για την αξιολόγηση του δικτύου μετά την ολοκλήρωση της εκπαίδευσης, Είναι πολύ σημαντικό στοιχεία του test dataset να μην γίνουν φανερά στο δίκτυο κατά τη διαδικασία της εκπαίδευσης, καθώς αυτό θα επηρεάσει την αξιοπιστία της απόδοσης του δικτύου. Παρόλα αυτά, χρειαζόμαστε ένα τρόπο να παρακολουθούμε την επίδοση του δικτύου κατά τη διαδικασία της μάθησης, ώστε να λάβουμε αποφάσεις σχετικά με τη διαδικασία της εκπαίδευσης δυναμικά. Για αυτό το λόγο, γίνεται χρήση του validation dataset, το οποίο χρησιμοποιείται για την αξιολόγηση του δικτύου μόνο κατά τη διάρκεια της εκπαίδευσης. Παρόλο που με αυτό τον τρόπο χάνουμε ένα κομμάτι χρήσιμης πληροφορίας από τα αρχικά δεδομένα, μας επιτρέπει να έχουμε μια εποπτεία της επίδοσης κατά την εκπαίδευση και να εφαρμόσουμε τεχνικές όπως το Early Stopping.

Κατά τη διάρκεια της εκπαίδευσης παρακολουθούμε τη μεταβολή στα losses για τα Train και Validation dataset. Μετά από κάποιο σημείο σε όλες τις περιπτώσεις παρατηρούμε πως το Train Loss μειώνεται, αλλά το Validation Loss παύει να μειώνεται, ενώ μπορεί και να αυξάνεται. Αυτό σηματοδοτεί πως το δίκτυο μαθαίνει σε τόσο λεπτομερή βαθμό τα δεδομένα του training dataset, που δεν έχει την ευελιξία να αναγνωρίσει δεδομένα που δεν έχει αντικρύσει. Αυτό το φαινόμενο ονομάζεται Overfitting και παρουσιάζεται σχηματικά στο σχήμα 3.4



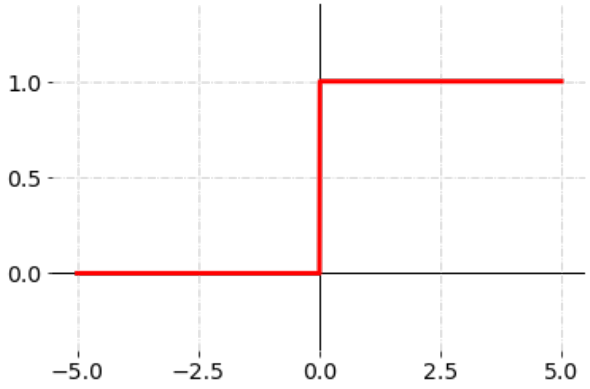
Σχήμα 3.4: *Φαινόμενο Overfitting*

3.2.5 Συναρτήσεις Ενεργοποίησης

Όπως αναφέρθηκε στην προηγούμενη ενότητα, η έξοδος του νευρωνικού προκύπτει περνώντας το άθροισμα της εισόδου πολλαπλασιασμένη με τα βάρη, μέσω από μια συνάρτηση ενεργοποίησης. Η επιλογή της κατάλληλης συνάρτησης ενεργοποίησης έχει κρίσιμη σημασία στην επίδοση του δικτύου. Οι παρακάτω συναρτήσεις ενεργοποίησης που αναφέρονται στο [10] αποτελούν τις συνηθέστερες.

- Βηματική συνάρτηση

$$\varphi(x) = \begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if} x \geq 0 \end{cases} \tag{3.15}$$

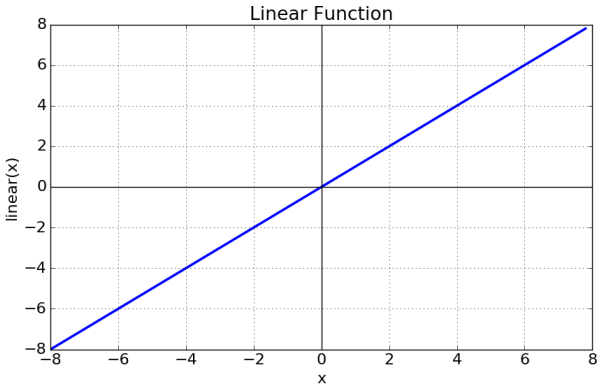


Σχήμα 3.5: *Βηματική συνάρτηση ενεργοποίησης*

Η βηματική συνάρτηση ή συνάρτηση Heavyside είναι η απλούστερη συνάρτηση ενεργοποίησης και αυτή που πρότεινε ο Frank Rosenblatt για το Perceptron. Ο κάθε νευρώνας ενεργοποιείται αν η έξοδος του ξεπεράσει το κατώφλι που ορίζεται (σε αυτή την περίπτωση 0). Η συνάρτηση αυτή είναι ασυνεχής και χρησιμοποιείται για δυαδική ταξινόμηση, αλλά δε μπορεί να χρησιμοποιηθεί όταν υπάρχουν περισσότερες από μία κλάσεις.

- Γραμμική συνάρτηση

$$\varphi(x) = a * x \tag{3.16}$$



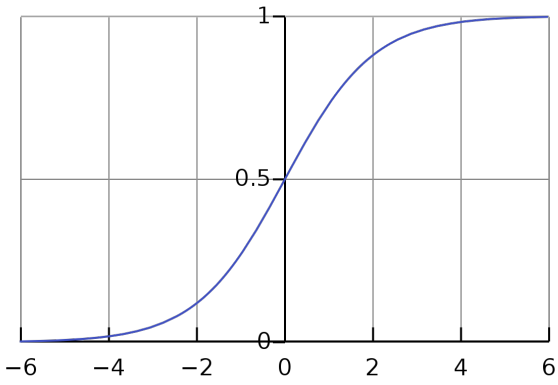
Σχήμα 3.6: *Γραμμική συνάρτηση ενεργοποίησης*

Σε αυτή τη συνάρτηση η έξοδος είναι ανάλογη του αθροίσματος του γινομένου της εισόδου με τα βάρη. Σε αντίθεση με την βηματική, η γραμμική συνάρτηση μπορεί να χρησιμοποιηθεί για multiclass classification. Παρόλα αυτά, δε χρησιμοποιείται ιδιαίτερα, αφού προκαλεί προβλήματα στον αλγόριθμό back-propagation και περιορίζει τις δυνατότητες του δικτύου, αφού η έξοδος θα είναι πάντα συνάρτηση του πρώτου επιπέδου.

Οι κυρίως χρησιμοποιούμενες συναρτήσεις ενεργοποίησης είναι οι μη γραμμικές. Ακολουθούν οι κύριες από αυτές.

- Σιγμοειδής συνάρτηση

$$\varphi(x) = \frac{1}{1 + e^{-x}} \tag{3.17}$$



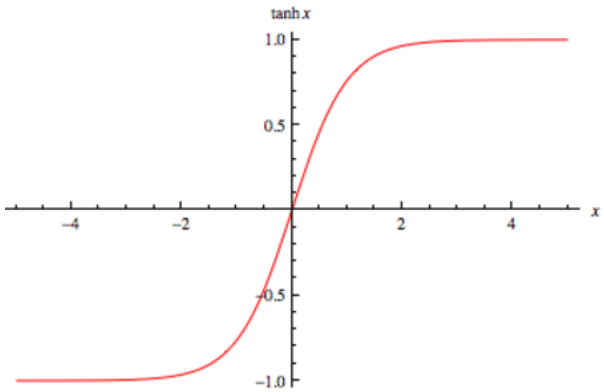
Σχήμα 3.7: *Σιγμοειδής συνάρτηση ενεργοποίησης*

Η σιγμοειδής συνάρτηση είναι μια από τις πιο ευρέως χρησιμοποιούμενες συναρτήσεις ενεργοποίησης, ειδικά σε εφαρμογές που περιλαμβάνουν πρόβλεψη πιθανοτήτων, καθώς λαμβάνει τιμές στο διάστημα (0,1). Επιπλέον, είναι διαφορίσιμη και μονότονη, το οποίο αποτρέπει μεγάλες διακυμάνσεις στις εξόδους των νευρώνων.

Η σιγμοειδής δεν είναι πάντα η σωστή επιλογή, καθώς δεν είναι συμμετρική γύρω από το 0, και επικεντρώνεται στις μεσαίες τιμές, ενώ οι πολύ μεγάλες ή πολύ μικρές τιμές δεν επηρεάζουν αισθητά την κλίση (vanishing gradient).

- Υπερβολική Εφαπτομένη

$$\varphi(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{3.18}$$

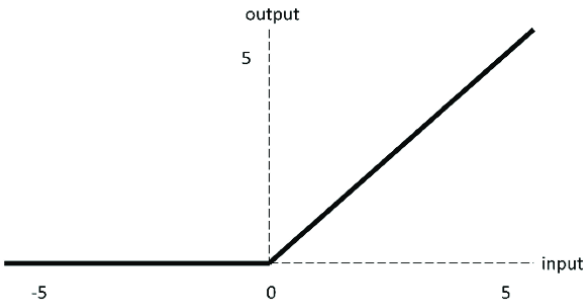


Σχήμα 3.8: *Υπερβολική Εφαπτομένη συνάρτηση ενεργοποίησης*

Η συνάρτηση υπερβολικής εφαπτομένης μοιάζει με τη σιγμοειδή αλλά είναι κεντραρισμένη γύρω από την αρχή των αξόνων, ενώ είναι πιο απότομη από την σιγμοειδή. Τόσο η συνάρτηση υπερβολικής εφαπτομένης, όσο και η σιγμοειδής χρησιμοποιούνται σε feedforward δίκτυα.

- `ReLU (Rectified linear unit)`

$$\phi(x) = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{if} x \geq 0 \end{cases} \tag{3.19}$$

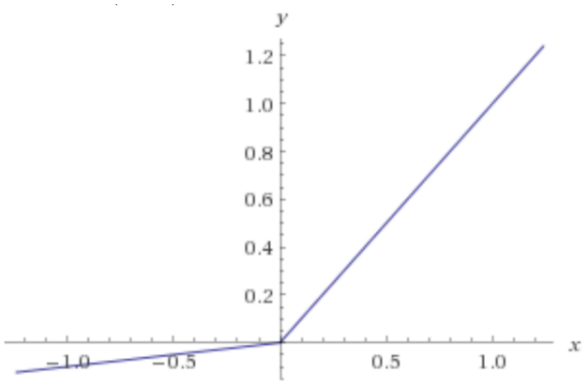


Σχήμα 3.9: *ReLU (Rectified linear unit)*

Το πλεονέκτημα της ReLU είναι ότι δεν ενεργοποιούνται ποτέ όλοι οι νευρώνες ταυτόχρονα, το οποίο την κάνει πιο αποδοτική. Σαν αρνητικό της προσάπτεται ότι η κλίση είναι μηδενική, οπότε δεν ανανεώνονται τα βάρη στη διαδικασία του backpropagation. Οι αρνητικές τιμές εισόδου δεν αντιστοιχίζονται λοιπόν σωστά.

Για την επίλυση του προηγούμενου προβλήματος προτείνεται η συνάρτηση Leaky ReLU, η οποία δίνει μια μικρή κλίση στις αρνητικές τιμές και ορίζεται με τον ακόλουθο τρόπο:

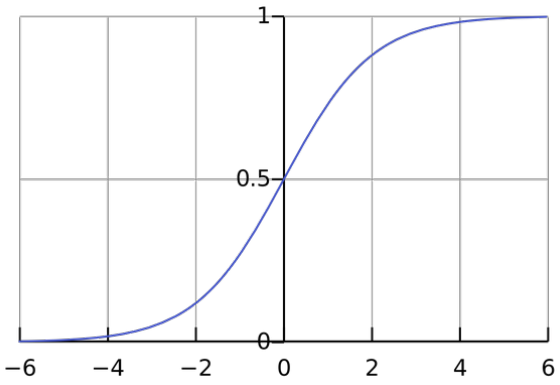
$$\phi(x) = \begin{cases} a * x & \text{if } x < 0 \\ x & \text{if} x \geq 0 \end{cases} \tag{3.20}$$



Σχήμα 3.10: *Leaky ReLU*

- `Softmax`

$$\phi(x) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{3.21}$$



Σχήμα 3.11: *Softmax*

Η Softmax αποτελεί συνδυασμό πολλαπλών σιγμοειδών συναρτήσεων και χρησιμοποιείται για multiclass classification. Η συνάρτηση επιστρέφει για κάθε στοιχείο εισόδου, την πιθανότητα της κάθε κλάσης.

3.2.6 Συνελικτικά Δίκτυα (Convolutional Neural Networks)

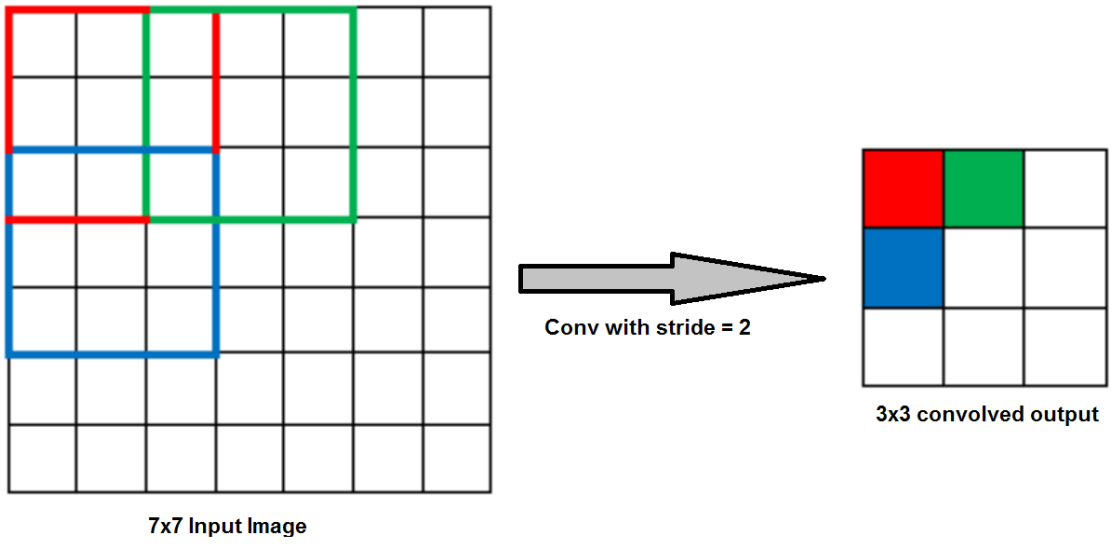
Τα συνελικτικά νευρωνικά δίκτυα αποτελούν μια ειδική κατηγορία feedforward δικτύων, τα οποία είναι ευρέως χρησιμοποιούμενα στην όραση υπολογιστών. Η επιτυχία τους σε εφαρμογές που αφορούν εικόνες αφορά στην ικανότητά τους να αναγνωρίζουν τοπικά χαρακτηριστικά και είναι αποδοτικά από άποψη υπολογιστικής ισχύος. Η λογική τους βασίζεται στη δομή των βιολογικών νευρωνικών δικτύων, συγκεκριμένα στον τρόπο με τον οποίο οι νευρώνες στον οπτικό φλοιό απαντούν σε ερεθίσματα μόνο σε κάποια περιοχή του πεδίου της όρασης. Το σύνολο αυτών των περιοχών αποτελεί το συνολικό οπτικό πεδίο. Τα συνελικτικά νευρωνικά δίκτυα αξιοποιούν το γεγονός πως η είσοδος είναι μια εικόνα (ή στην περίπτωσή μας ένα φασματογράφημα), και προσπαθούν να αναγνωρίσουν συγκεκριμένες περιοχές της εισόδου. Αποτελούνται κυρίως από δύο είδη επιπέδων, τα συνελικτικά και τα pooling επίπεδα, τα οποία θα εξετάσουμε ξεχωριστά.

Το πρώτο βασικό επίπεδο είναι το συνελικτικό. Η επεξεργασία που γίνεται σε αυτό το επίπεδο βασίζεται σε φίλτρα ή πυρήνες (kernels) τα οποία εκτελούν ένα συνελικτικό γινόμενο με την είσοδο. Συγκεκριμένα, μετακινούμε το φίλτρο πάνω στην είσοδο και η έξοδος είναι το άθροισμα της συνέλιξης μεταξύ της εισόδου και του φίλτρου. Μπορούμε να ορίσουμε το βήμα με το οποίο μετακινείται το φίλτρο πάνω στην είσοδο (stride). Συνοπτικά, η μεταφορά της εισόδου στην έξοδο παρουσιάζεται στο σχήμα 3.12.

Για καλύτερη εφαρμογή του φίλτρου μπορούμε να εφαρμόσουμε γέμισμα (padding) στις άκρες της εικόνας. Αυτό βοηθάει στο να αξιοποιούνται περισσότερο οι τιμές στα άκρα της εισόδου, που σε άλλη περίπτωση θα χρησιμοποιούνταν λιγότερο από τις κεντρικές.

Το δεύτερο βασικό επίπεδο είναι το pooling επίπεδο. Όπως και με το συνελικτικό επίπεδο, ο σκοπός του είναι να μειώσει τη διάσταση της εισόδου, μειώνοντας ταυτόχρονα και την απαιτούμενη υπολογιστική ισχύ. Πετυχαίνουν, τέλος την εξαγωγή κυρίαρχων χαρακτηριστικών τα οποία δεν εξαρτώνται από τη θέση και την περιστροφή.

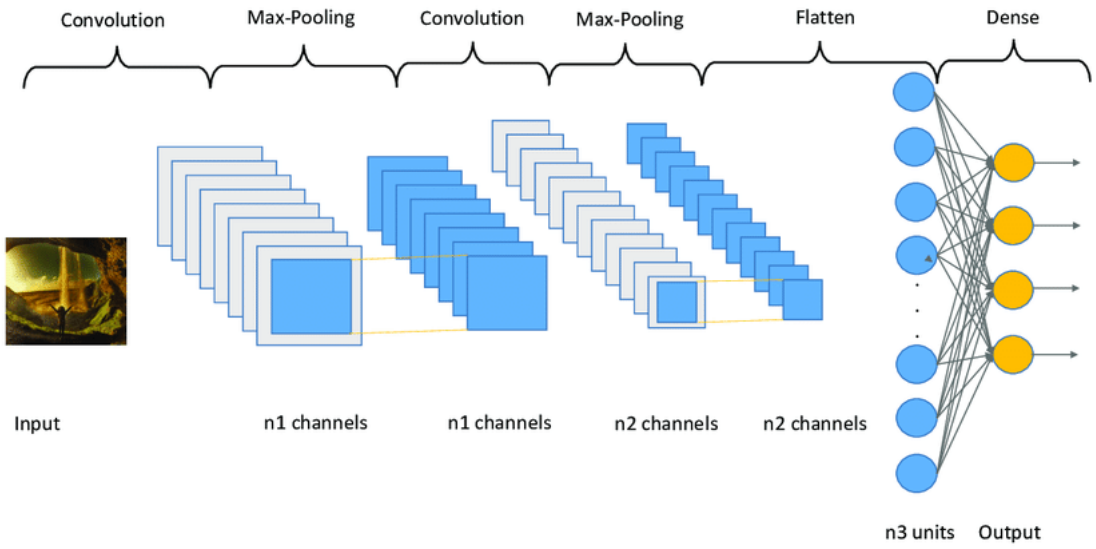
Το pooling επιτυγχάνεται μετακινώντας ένα πυρήνα πάνω στην είσοδο και διατηρώντας



Σχήμα 3.12: *Μετακίνησή φίλτρου 3x3 πάνω σε είσοδο συνεκτικού δικτύου*

μία τιμή για κάθε τμήμα, η οποία εξαρτάται από το είδος του pooling.Υπάρχουν δύο κύρια είδη Pooling: max pooling και average pooling. Στην περίπτωση του Max Pooling, διατηρείται η μέγιστη τιμή του τμήματος της εισόδου που καλύπτεται από τον πυρήνα, ενώ στην περίπτωση του Average Pooling διατηρείται η μέση τιμή αυτού του τμήματος. Έχει παρατηρηθεί ότι τα επίπεδα Max Pooling αποδίδουν αρκετά καλύτερα, αφού έχουν το σημαντικό πλεονέκτημα της μείωσης θορύβου στην είσοδο, αφού απορρίπτουν ταυτόχρονα πολλές θορυβώδεις περιοχές.

Τα συνελικτικά δίκτυα αποτελούνται από πλήθος ζευγαριών συνελικτικών και pooling επιπέδων, τα οποία συνήθως καταλήγουν σε ένα πλήρως συνδεδεμένο δίκτυο, το οποίο μπορεί να εξάγει τα χαρακτηριστικά της εισόδου ως μονοδιάστατο διάνυσμα. Όσο πιο βαθύ είναι το δίκτυο, τόσο πιο υψηλού επιπέδου χαρακτηριστικά εξάγονται για την είσοδο.



Σχήμα 3.13: *Αρχιτεκτονική απλού Συνελικτικού Δικτύου*

3.3 Αναδρομικά Δίκτυα (Recurrent Neural Networks)

Τα νευρωνικά δίκτυα που αναλύσαμε μέχρι στιγμής είναι στατικά, δηλαδή δεν εισέρχεται ο χρόνος στη διαδικασία της μάθησης. Αυτό δεν αποτελεί πρόβλημα για διάφορα είδη εφαρμογών, όπως η ταξινόμηση εικόνων, αλλά στην περίπτωση του ήχου και της μουσικής, τα δεδομένα εξαρτώνται σε μεγάλο βαθμό από το χρόνο, επομένως τα στατικά δίκτυα δεν αρκούν. Για αυτό το λόγο, θα ασχοληθούμε με μια τελευταία κατηγορία νευρωνικών δικτύων, τα οποία έχουν ως βασικό χαρακτηριστικό τη διατήρηση μνήμης για τα δεδομένα εισόδου, τα αναδρομικά νευρωνικά δίκτυα (Recurrent Neural Networks - RNNs).

3.3.1 Βασική Αρχιτεκτονική

Τα αναδρομικά νευρωνικά δίκτυα χρησιμοποιούνται για ακολουθίες δεδομένων. Θεωρούμε ως ακολουθία εισόδου το διάνυσμα

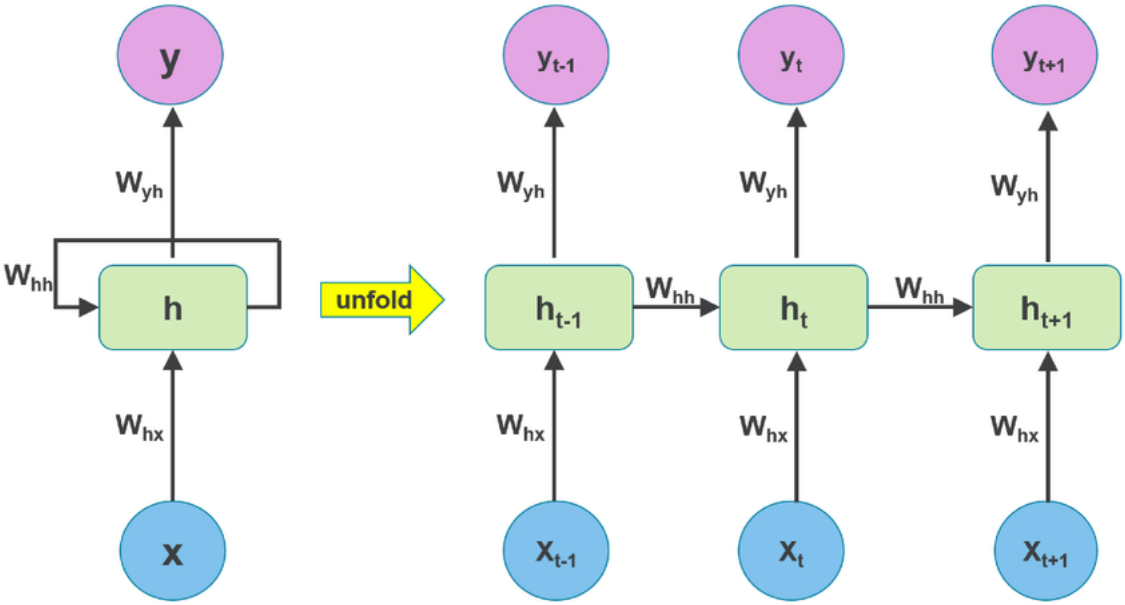
$$x^T = x_1, x_2, \ldots, x_{T-1}, x_T$$

και ως ακολουθία εξόδου το διάνυσμα

$$y^T = y_1, y_2, \ldots, y_{T-1}, y_T$$

με τα γειτονικά ζεύγη δεδομένων να είναι οργανωμένα με βάση το χρόνο. Δεδομένης της εισόδου και της εξόδου, ο σκοπός του δικτύου είναι να αναγνωρίσει την ακολουθία εξόδου με βάση την ακολουθία εισόδου. Σε αντίθεση με τα feedforward δίκτυα στα οποία η ροή της πληροφορίας είναι σειριακή, η πληροφορία στα RNNs μεταφέρεται κυκλικά με ανατροφοδότηση. Συγκεκριμένα τα RNNs αποτελούνται από τα επίπεδα εισόδου και εξόδου, καθώς και ένα σύνολο κρυφών επιπέδων. Για να γίνει πιο κατανοητή η λειτουργία, μπορούμε

να ξεδιπλώσουμε το δίκτυο σε διαφορετικές χρονικές στιγμές, όπως φαίνεται στο σχήμα 3.14.



Σχήμα 3.14: *Αναδρομικό Δίκτυο Αναδιπλωμένο στο Χρόνο*

Με βάση αυτό, η έξοδος τη χρονική στιγμή $t+1$ για ένα δίκτυο με m κρυφά επίπεδα προκύπτει από τη σχέση 3.22

$$h_{t+1} = \phi(W_{hx}x_t + W_{hh}h_t + b_h) \tag{3.22}$$

όπου $W_{hx} \in \mathbb{R}^m$ τα βάρη που αφορούν τις εισόδους, $W_{hh} \in \mathbb{R}^{m \times m}$ τα βάρη που αφορούν τα κρυφά επίπεδα, $b_h \in \mathbb{R}^m$ η αρχική πόλωση των εσωτερικών επιπέδων και φ η συνάρτηση ενεργοποίησης. Η συνολική έξοδος του δικτύου προκύπτει ως

$$y_t = \phi\left(W_{yh} \cdot h_t + b_y\right) \tag{3.23}$$

όπου $W_{yh} \in \mathbb{R}^m$ τα βάρη που σχετίζονται με τις συνδέσεις στην έξοδο και $b_y \in \mathbb{R}^m$ η πόλωση της εξόδου.

3.3.2 Backpropagation through time (BPTT) και Vanishing Gradient

Για την εκπαίδευση των RNNs χρησιμοποιείται μια παραλλαγή του αλγόριθμου back-propagation που αναλύθηκε στην προηγούμενη ενότητα. Αφού παραχθεί το διάνυσμα h_k, υπολογίζουμε το σφάλμα E_k με σκοπό να υπολογίσουμε τη κλίση

$$\frac{\partial E}{\partial w} = \sum_{t=1}^{T} \frac{\partial E_t}{\partial w} \tag{3.24}$$

Η κλίση αυτή μπορεί να εκφραστεί μετά από υπολογισμούς ως

$$\frac{\partial E}{\partial w} = \frac{\partial E}{\partial h_T} \sum_{t=1}^{T} \left(\left(\prod_{i=t+1}^{n} \frac{\partial h_i}{\partial h_{i-1}} \right) \frac{\partial h_t}{\partial w} \right) \tag{3.25}$$

Η σχέση αυτή αναδεικνύει ένα πρόβλημα στα απλά αναδρομικά, το οποίο είναι γνωστό ως vanishing and exploding gradient. Αν η ακολουθία εισόδου είναι μεγάλη (δηλαδή πολλά βήματα), τότε εισέρχονται πολλοί πολλαπλασιασμοί μερικών παραγώγων για τον υπολογισμό των κλίσεων. Επομένως μπορούμε να δούμε πως αν το μέτρο των μερικών παραγώγων είναι μικρότερο από 1, το γινόμενο αυτό τείνει να μηδενιστεί, όσο μεγαλώνει η είσοδος.

$$\left|\frac{\partial h_i}{\partial h_{i-1}}\right| < 1 \longrightarrow \prod_{i=t+1}^{n} \frac{\partial h_i}{\partial h_{i-1}} \to 0$$

Αντίστοιχα, όταν το μέτρο των μερικών παραγώγων είναι μεγαλύτερο του 1, η κλίση τείνει να γίνει πολύ μεγάλη

$$\left|\frac{\partial h_i}{\partial h_{i-1}}\right| > 1 \longrightarrow \prod_{i=t+1}^{n} \frac{\partial h_i}{\partial h_{i-1}} \to \infty$$

Λόγω αυτού του προβλήματος, η ανανέωση των βαρών είναι αδύνατη, αφού η διόρθωση που υφίστανται από τον αλγόριθμο backpropagation είναι ανάλογη της κλίσης. Αυτό το βασικό πρόβλημα των RNNs επιλύεται με τη χρήση ενός είδους RNN, τα Long Term Short Term Memory Networks (LSTMs).

3.3.3 Δίκτυα Long Term Short Term Memory (LSTMs)

Τα LSTMs είναι μια ειδική κατηγορία αναδρομικών δικτύων, τα οποία έχουν τη δυνατότητα να μαθαίνουν μακροπρόθεσμες εξαρτήσεις. Προτάθηκαν πρώτη φορά από τους Hochreiter και Schmidhuber (1997) και χρησιμοποιούνται ευρέως μέχρι και σήμερα σε πλήθος εφαρμογών, ιδιαίτερα σε εφαρμογές φυσικής γλώσσας (Natural Language Processing) καθώς και στην επεξεργασία ήχου.

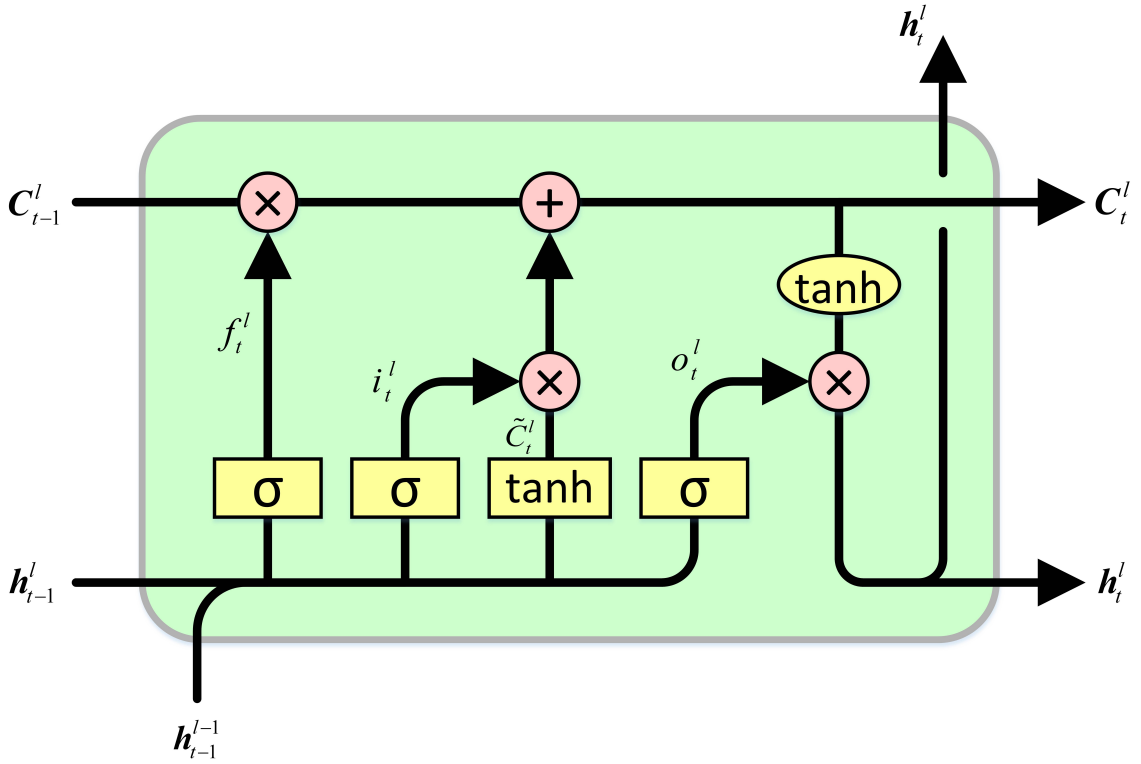
Η δομή των LSTMs είναι παρόμοια με αυτή των αναδρομικών δικτύων, αποτελούνται δηλαδή από επαναλαμβανόμενα κελιά. Σε αντίθεση όμως με τα απλά αναδρομικά δίκτυα, οι μονάδες των οποίων αποτελούνται μόνο από τη συνάρτηση ενεργοποίησης, η δομή τους είναι πιο περίπλοκη. Οι μονάδες αποτελούνται από 4 επίπεδα νευρωνικού δικτύου τα οποία αλληλεπιδρούν μεταξύ τους.

Το κύριο τμήμα της μονάδας είναι η πάνω οριζόντια γραμμή, η κατάσταση του κελιού μνήμης C_t. Διατρέχει ολόκληρη την αλυσίδα και η πληροφορία μεταφέρεται μέσω αυτής απαράλλαχτη. Η απομάκρυνση και προσθήκη πληροφορίας γίνεται μέσω των των πυλών, οι οποίες είναι συνδυασμός σιγμοειδών συναρτήσεων ενεργοποίησης και πολλαπλασιαστών. Τα LSTMs διαθέτουν 3 τέτοιες πύλες, ώστε να ελέγχουν την κατάσταση του κελιού.

Το πρώτο βήμα στη λειτουργία του LSTM είναι η απόφαση της πληροφορίας η οποία θα απορριφθεί. Αυτό επιτυγχάνεται από την forget gate, και αναπαρίσταται στο διάγραμμα ως f_t. Χρησιμοποιώντας ως εισόδους τα h_{t-1} και x_t, δίνει μια έξοδο στο διάστημα [0,1], υποδεικνύοντας το πόσο θα διατηρηθεί η πληροφορία.

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{3.26}$$

Το επόμενο βήμα είναι η απόφαση της πληροφορίας που θα αποθηκευτεί στη κατάσταση του κελιού. Αυτό αποτελείται από δύο τμήματα. Το πρώτο επίπεδο ονομάζεται input gate



Σχήμα 3.15: *Εσωτερική Δομή Μονάδας LSTM*

layer και αναπαρίσταται ως i_t στο διάγραμμα, αποφασίζει ποιές τιμές θα ανανεωθούν. Στη συνέχεια, το επίπεδο της υπερβολικής εφαπτομένης δημιουργεί ένα νέο σύνολο τιμών $\tilde{C}_t$, οι οποίες θα προστεθούν στη κατάσταση του κελιού. Το τελικό αποτέλεσμα προκύπτει από τον πολλαπλασιασμό των δύο αυτών επιπέδων.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \qquad \tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{3.27}$$

Με βάση αυτά τα 3 επίπεδα, ανανεώνεται η παλιά τιμή της κατάστασης κελιού

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{3.28}$$

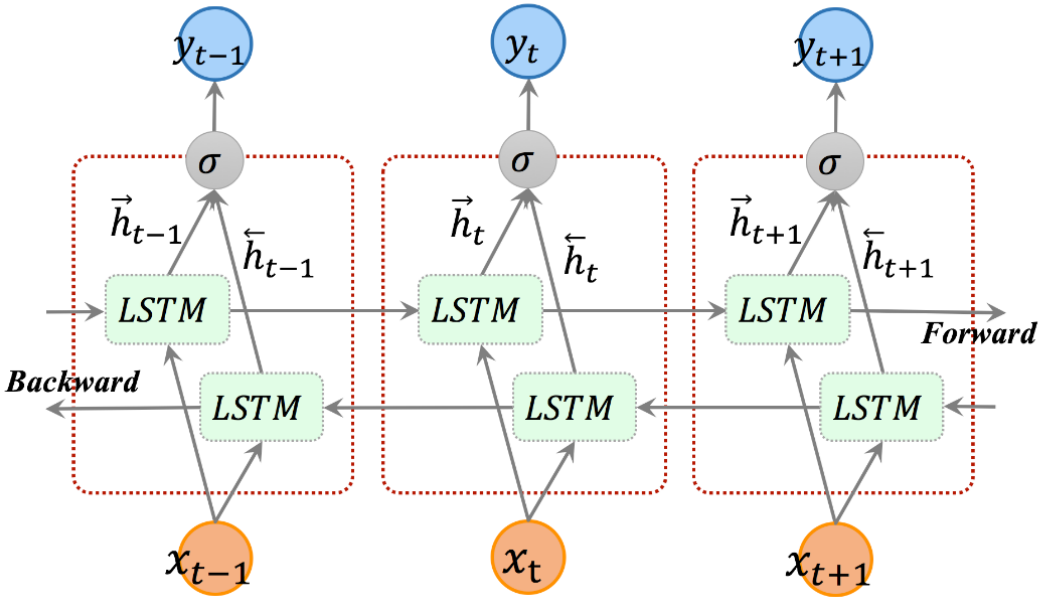
Τέλος, πρέπει να αποφασιστεί τι θα παραχθεί ως έξοδος. Αυτό γίνεται βάση φιλτραρίσματος της κατάστασης κελιού. Συγκεκριμένα, αποφασίζεται μέσω μιας σιγμοειδούς συνάρτησης ενεργοποίησης ποια τμήματα θα ενεργοποιηθούν o_t. Στη συνέχεια η κατάσταση του κελιού περνάει από μια συνάρτηση υπερβολικής εφαπτομένης, ώστε να μεταφερθούν οι τιμές στο διάστημα [-1,1], και πολλαπλασιάζεται με την έξοδο o_t, ώστε να δοθεί η τελική έξοδος της μονάδας.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \qquad h_t = o_t * \tanh(C_t) \tag{3.29}$$

3.3.4 LSTM Διπλής Κατεύθυνσης

Μέχρι στιγμής έχουν περιγραφεί αρχιτεκτονικές αναδρομικών δικτύων μιας κατεύθυνσης, δηλαδή οι ακολουθίες εισόδου διαβάζονται μόνο από μία κατεύθυνση. Πολλές φορές

είναι χρήσιμο να επεξεργαστεί η ακολουθία και από τις δύο κατευθύνσεις. Αυτό το ρόλο καλύπτουν τα LSTM διπλής κατεύθυνσης (bi-directional LSTMs).



Σχήμα 3.16: *LSTM Διπλής Κατεύθυνσης*

Η αρχή λειτουργίας τους είναι παρόμοια με αυτή των κανονικών LSTMs. Στην αναδιπλωμένη μορφή, κάθε επίπεδο περιέχει ένα "μπροστά" (forward) και ένα "πίσω" (backward) LSTM επίπεδο, η έξοδος των οποίων συνδέεται στην ίδια έξοδο. Η forward έξοδος $\overrightarrow{h_t}$ υπολογίζεται μέσω των σχέσεων του LSTM, για τα δείγματα της ακολουθίας $[T-n, T-1]$, ενώ η (backward) έξοδος $\overleftarrow{h_t}$ υπολογίζεται από τις ίδιες σχέσεις αλλά για την αντεστραμμένη ακολουθία εισόδου $[T-n, T-1]$. Η συνολική έξοδος y_t προκύπτει συνδυάζοντας τις δύο ακολουθίες.

$$y_t = \sigma(\overrightarrow{h_t}, \overleftarrow{h_t}) \tag{3.30}$$

Εδώ σ μπορεί να είναι μια συνάρτηση συνένωσης, άθροισης ή πολλαπλασιασμού.

Μέρος II

Πρακτικό Μέρος

Κεφάλαιο 4

Σχεδιασμός Πειραμάτων

Στο κεφάλαιο αυτό παρουσιάζονται τα βήματα που έγιναν για το σχεδιασμό και την υλοποίηση των πειραμάτων.

4.1 Περιβάλλον Υλοποίησης

Η υλοποίηση των μοντέλων και των πειραμάτων έγιναν με χρήση της γλώσσας προγραμματισμού Python. Για την εκπαίδευση των νευρωνικών δικτύων χρησιμοποιήθηκε η βιβλιοθήκη pytorch, καθώς παρέχει πλήθος εργαλείων που βοηθούν στην βαθιά μάθηση, ενώ παράλληλα επιτρέπει μια αρκετά χαμηλού επιπέδου επαφή με τη διαδικασία της μάθησης. Το γεγονός αυτό βοηθά στην εύκολη εξειδίκευση και προσαρμογή των μοντέλων. Χρησιμοποιήθηκε επίσης εκτενώς η βιβλιοθήκη librosa για την επεξεργασία ήχου.

Το περιβάλλον που έγινε η εκπαίδευση ήταν το Google Colab, καθώς αυτό προσφέρει ένα εύκολο στη χρήση περιβάλλον αλληλεπίδρασης, και το σημαντικότερο, προσφέρει υπολογιστικούς πόρους της Google (GPU). Οι πόροι αυτοί βοήθησαν σε πολύ μεγάλο βαθμό ώστε να γίνει σε λογικά χρονικά πλαίσια η εκπαίδευση των νευρωνικών δικτύων, διαδικασία που είναι γενικά ιδιαίτερα χρονοβόρα.

4.2 Σύνολα Δεδομένων (Dataset)

4.2.1 GuitarSet

Το Dataset ([5]) που χρησιμοποιήθηκε για την εκπαίδευση και την αξιολόγηση των μοντέλων είναι το GuitarSet. Το dataset αυτό αποτελείται από κομμάτια κιθάρας με τις αντίστοιχες σημειώσεις για κάθε νότα που παίζεται. Όπως αναφέρεται στην αντίστοιχη δημοσίευση [4], παρόλο που η κιθάρα και το πιάνο είναι εξίσου δημοφιλή, τα dataset που αφορούν επιβλεπόμενη μάθηση μουσικής αφορούν αποκλειστικά το πιάνο (MAPS, MAESTRO). Επομένως, το GuitarSet είναι ιδιαίτερα χρήσιμο για τη δημιουργία μοντέλων που δεν αφορούν αποκλειστικά το πιάνο.

Το GuitarSet αποτελείται από 360 κομμάτια μήκους σχεδόν 30 δευτερολέπτων. Τα κομμάτια αυτά δημιουργήθηκαν από έξι διαφορετικούς παίχτες, δεδομένων 30 διαφορετικών οδηγών (lead sheets) που αποτελούνται από διάφορα είδη μουσικής (Rock, Singer-Songwriter, Bossa Nova, Jazz, Funk), ακολουθίες και ρυθμούς. Κάθε κομμάτι έχει δύο

παραλλαγές, μία είναι η συνοδεία, η οποία περιέχει κυρίως συγχορδίες, και η άλλη είναι το solo, που περιέχει κυρίως μελωδίες.

Οι ηχογραφήσεις έγιναν μέσω μαγνητών και μικροφώνου, με τους μαγνήτες να προσδίδουν την καλύτερη πιστότητα. Χρησιμοποιήθηκε ειδικός εξαφωνικός μαγνήτης που καταγράφει το σήμα της κάθε χορδής ξεχωριστά, το οποίο χρησιμοποιείται στη σημείωση των δεδομένων. Η σημείωση δίνεται σε μορφή αρχείου JAMS, τα οποία περιέχουν πληροφορίες για το τονικό ύψος και τη χροιά της νότας που παίζεται, μεταξύ άλλων.

Η κύρια κατηγορία σημειώσεων που αφορά την εφαρμογή είναι τα midi note annotations. Σε αυτά καταγράφεται για κάθε χορδή ξεχωριστά πληροφορία για τις νότες που παίζονται, σε αναπαράσταση midi, καθώς και τα onsets και offsets της νότας, τα οποία είναι τα βασικά χαρακτηριστικά που χρειαζόμαστε για τη μάθηση.

4.2.2 Προ-επεξεργασία Δεδομένων

Το πρώτο βήμα στην προ-επεξεργασία των δεδομένων, είναι το resampling των ηχητικών κομματιών, ώστε να έχουν το σωστό sampling rate που χρησιμοποιούμε στα μοντέλα, δηλαδή 16 kHz. Τα αρχικά κομμάτια του dataset έχουν sampling rate 44.1 kHz, οπότε τα μετατρέπουμε στην κατάλληλη μορφή με χρήση του λογισμικού ffmpeg ([11]).

Το δεύτερο βασικό βήμα, είναι να αποσπάσουμε από το αρχείο JAM, την πληροφορία για τα onset και τα offset των νοτών. Ο τρόπος που αυτά είναι αποθηκευμένα στο αρχείο είναι ταξινομημένα με βάση το onset της νότας, ξεχωριστά για κάθε χορδή. Επομένως, μέσω ενός script jams2tsv.py αρχικά τοποθετούμε όλες τις νότες σε μία κοινή λίστα και την ταξινομούμε με βάση τα onsets.

Με τα παραπάνω βήματα έχουμε τα δεδομένα εισόδου και τα αντίστοιχα labels για την μάθηση, αλλά αυτά αντιπροσωπεύουν νότες midi. Όπως αναφέρθηκε στην ενότητα 2.1.3, το τονικό ύψος δεν περιέχει όλη την πληροφορία για τη θέση στην κιθάρα, οπότε πρέπει να γίνει μια κατάλληλη αναπαράσταση που να περιέχει και αυτή την πληροφορία.

4.2.3 Αναπαράσταση τάστων

Για την αναπαράσταση της πληροφορίας συγκεκριμένων τάστων, θεωρήθηκε αρχικά πως η κιθάρα έχει το κούρδισμα ¨Standard¨ (E2 - A2 - D3 - G3 - B3 - E4) και 22 τάστα, δηλαδή τα πιο διαδεδομένα χαρακτηριστικά. Με βάση αυτό, ορίζεται ο αριθμός stringfret, ο οποίος αντιστοιχίζει τη νότα στο τάστο f της χορδής s σε μια μοναδική τιμή ως. Για να λάβουμε την τιμή midi md από μια τιμή stringfret sf, χρησιμοποιούμε τον τύπο 4.1, όπου ως base ορίζεται ένας πίνακας 6 θέσεων που περιέχει τις νότες midi των ανοιχτών χορδών της κιθάρας [40, 45, 50, 55, 59, 64].

$$md = \text{base}[sf//23] + sf\%23 \tag{4.1}$$

Οι αριθμοί αυτοί που περιέχονται σε κάθε χορδή φαίνονται στον πίνακα 4.1. Αυτή η αντιστοιχία είναι πολύ κοντά στην αναπαράσταση midi, και βοηθάει στην τροποποίησή υπαρχόντων μοντέλων ώστε να έχουμε καλύτερη απόδοση. Με βάση αυτή τη σχέση, ακολουθείται η ίδια διαδικασία επεξεργασίας των δεδομένων, ώστε τώρα τα αρχεία labels να έχουν

τα onsets και τα offsets νοτών, αλλά σε μορφή stringfret.

Χορδή	Διάστημα stringfret
E	0 - 22
A	23 - 45
D	46 - 68
G	69 - 91
B	92 - 114
e	115 - 137

Πίνακας 4.1: *Διαστήματα stringfret σε κάθε χορδή της κιθάρας*

4.3 Αρχιτεκτονική Δικτύων

Αρχικά, θα παρουσιάσουμε τα θεμελιώδη τμήματα που αποτελούν όλα τα μοντέλα. Οι αρχιτεκτονικές βασίστηκαν σε μεγάλο βαθμό στην αρχιτεκτονική που παρουσιάζεται στο [1].

Φασματογράφημα Mel

Για την αναπαράσταση των δεδομένων εισόδου, χρησιμοποιήθηκε Φασματογράφημα Mel, για τους λόγους που αναφέρθηκαν στην ενότητα 2.2.5. Οι παράμετροι του φασματογραφήματος είναι

- 229 frequency bins λογαριθμικής κλίμακας
- hop length 512 samples
- παράθυρο FFT 2048 samples
- ρυθμός δειγματοληψίας 16 kHz

Η έξοδος μεταβιβάζεται ολόκληρη στις εισόδους του νευρωνικού, σαν μια εικόνα.

Ακουστικό Μοντέλο (Convolutional Stack)

Θεμέλιο όλων των αρχιτεκτονικών, αποτελεί το ακουστικό μοντέλο που περιγράφεται στη δημοσίευση [1]. Το πλήθος των συνδέσεων εξαρτάται από το πλήθος των επιθυμητών χαρακτηριστικών εισόδου και εξόδου (input features και output features), καθώς και το πλήθος των frames. Αυτά μπορεί να αλλάζουν για διαφορετικές υλοποιήσεις. Συγκεκριμένα, για το μοντέλο που περιγράφεται στο [1], το οποίο αφορά τη μεταγραφή μουσικής πιάνου, έχουμε 88 χαρακτηριστικά εξόδου (όσα τα πλήκτρα του πιάνου). Αντίστοιχα, στην αναπαράσταση των τάστων της κιθάρας που περιγράφεται στην ενότητα 4.2.3, έχουμε 139 χαρακτηριστικά εξόδου.

- Είσοδος: batch size * 1 κανάλι * frames * input Features
- Επίπεδο 1
  - Συνελικτικό Επίπεδο : Κανάλια εισόδου 1, κανάλια εξόδου 48, kernel 3x3, padding 1
  - Batch Normalization 2D επίπεδο

  - ReLU Συνάρτηση ενεργοποίησης

- Επίπεδο 2
  - Συνελικτικό Επίπεδο : Κανάλια εισόδου 48, κανάλια εξόδου 48, kernel 3x3, padding 1
  - Batch Normalization 2D επίπεδο
  - ReLU Συνάρτηση ενεργοποίησης
- Επίπεδο 3
  - MaxPooling2D επίπεδο, kernel 1x2
  - Dropout επίπεδο με πιθανότητα $p = 0.25$
  - Συνελικτικό Επίπεδο : Κανάλια εισόδου 48, κανάλια εξόδου 96, kernel 3x3, padding 1
  - Batch Normalization 2D επίπεδο
  - ReLU Συνάρτηση ενεργοποίησης
- Επίπεδο 4
  - MaxPooling2D επίπεδο, kernel 1x2
  - Dropout επίπεδο με πιθανότητα $p = 0.5$
- Μετατροπή διανύσματος σε : batch size * frames * (input Features // 8 * 192)
- Fully Connected επίπεδο, Είσοδος (input Features // 8 * output Features // 4), έξοδος : 768
- Dropout επίπεδο με πιθανότητα $p = 0.5$

Τέλος, χρησιμοποιήθηκαν τα ακόλουθα blocks εκτενώς ως κλάσεις.

ConvBlock

- ConvStack επίπεδο, είσοδος (input Features)
- Fully Connected Επίπεδο, είσοδος 768, έξοδος (outputFeatures)
- Σιγμοειδής Συνάρτηση Ενεργοποίησης

LSTMBlock 1

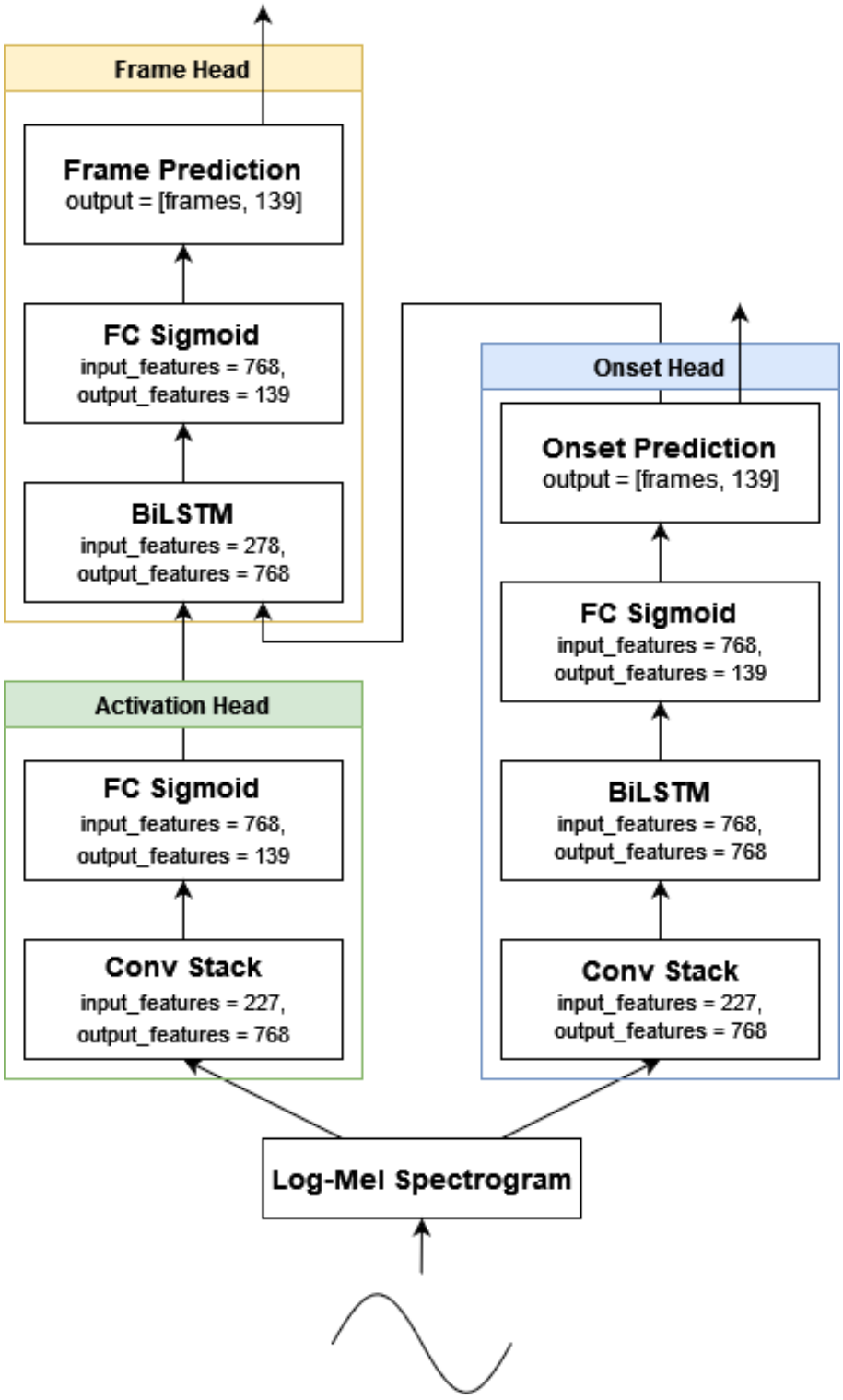
- ConvStack επίπεδο, είσοδος (input Features)
- BiLSTM επίπεδο, είσοδος 768, έξοδος 768
- Fully Connected Επίπεδο, είσοδος 768, έξοδος (outputFeatures)
- Σιγμοειδής Συνάρτηση Ενεργοποίησης

LSTMBlock 2

- BiLSTM επίπεδο, είσοδος input Features * 2, έδοξος 768
- Fully Connected Επίπεδο, είσοδος 768, έξοδος (outputFeatures)
- Σιγμοειδής Συνάρτηση Ενεργοποίησης

4.3.1 Μοντέλο OAF Org

Το μοντέλο αυτό βασίστηκε στο μοντέλο που παρουσιάστηκε στη δημοσίευση [1]. Αποτελεί μετατροπή του μοντέλου Onsets And Frames, το οποίο όμως αντί να αφορά το πιάνο, έχει αλλαχθεί ώστε πλέον να έχει τα features της κιθάρας. Συγκεκριμένα, αντί για τα 88 output features που είχε το μοντέλο αρχικά (πλήκτρα του πιάνου), τώρα έχει 139 (πλήθος stringfret). Το μοντέλο OAF Org παρουσιάζεται στο σχήμα 4.1. Ο σκοπός της υλοποίησης του μοντέλου αυτού είναι να υπάρξει ένα μέτρο αναφοράς για την απόδοση των υπόλοιπων.



Σχήμα 4.1: *Μοντέλο OAF Org.*

Το μοντέλο αυτό περιέχει 2 κεφαλές εξόδου, τις onset head και frame head, και μία ενδιάμεση κεφαλή, την activation head. Η κεφαλή onset αναγνωρίζει τα onsets των νοτών, δηλαδή έχει την τιμή 1 στο αντίστοιχο frame, εάν η νότα αυτή ξεκινάει σε εκείνο το frame. Αντίστοιχα, η κεφαλή frame, αναγνωρίζει τα frames στα οποία η νότα είναι ¨ενεργή¨, δηλαδή

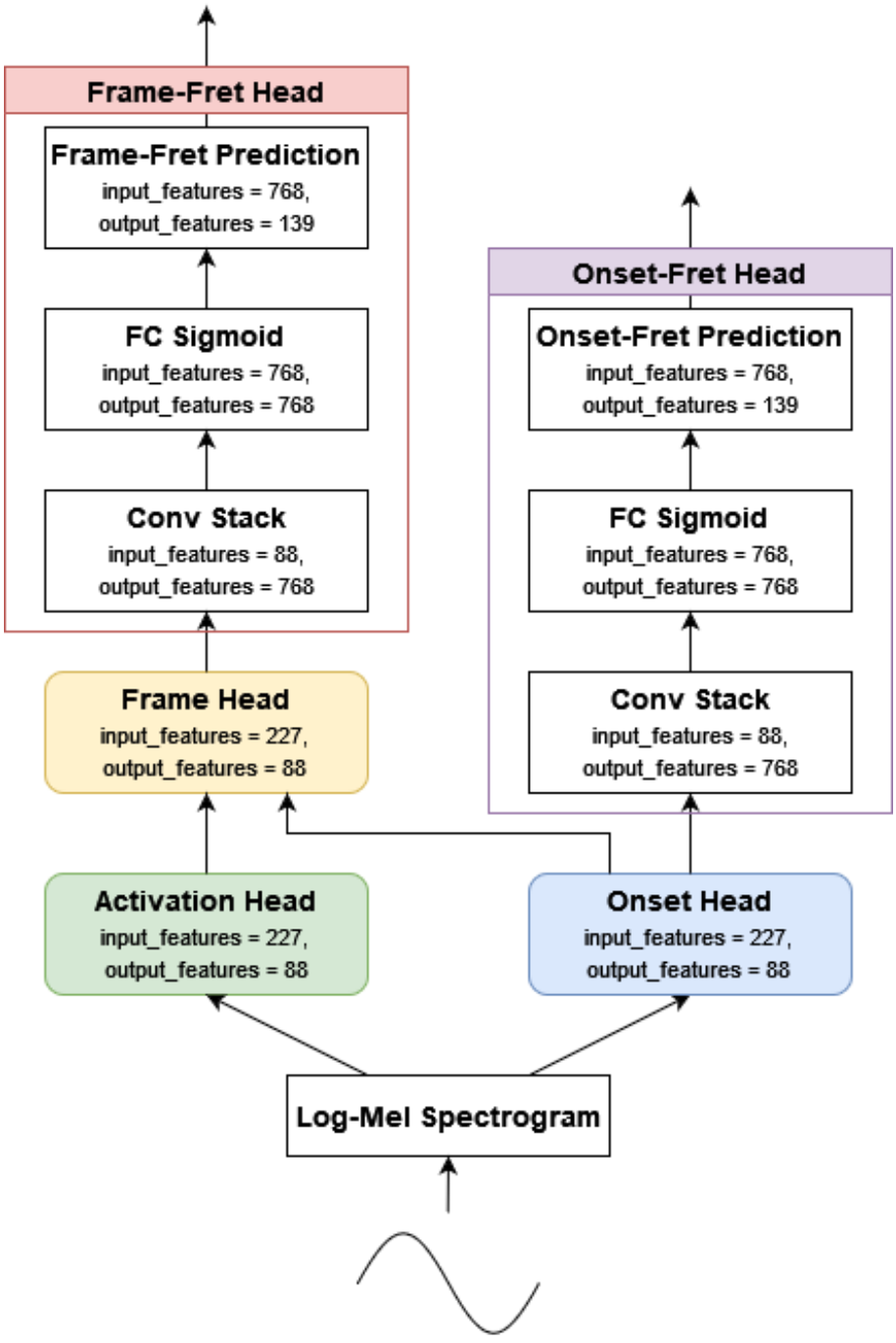
έχει την τιμή 1 στο αντίστοιχο frame αν η νότα είναι ενεργή σε αυτό το frame.

Ο διαχωρισμός αυτός βασίζεται στο γεγονός ότι κάποια frames είναι πιο σημαντικά από άλλα. Συγκεκριμένα, η χρονική στιγμή που ξεκινάει μια νότα, μπορεί να θεωρηθεί ως η πιο σημαντική, καθώς περιέχει το πλήθος της πληροφορίας, ενώ όλα τα υπόλοιπα frames που είναι ενεργή απλά προσδίδουν πληροφορία για τη διάρκεια. Επομένως, χρησιμοποιείται ειδική κεφαλή αναγνώρισης αυτών των frames. Η έξοδος του onset head προκύπτει από μια σιγμοειδή συνάρτηση, η οποία ορίζει τις πιθανότητες κάθε ένα από τα 139 stringfret.

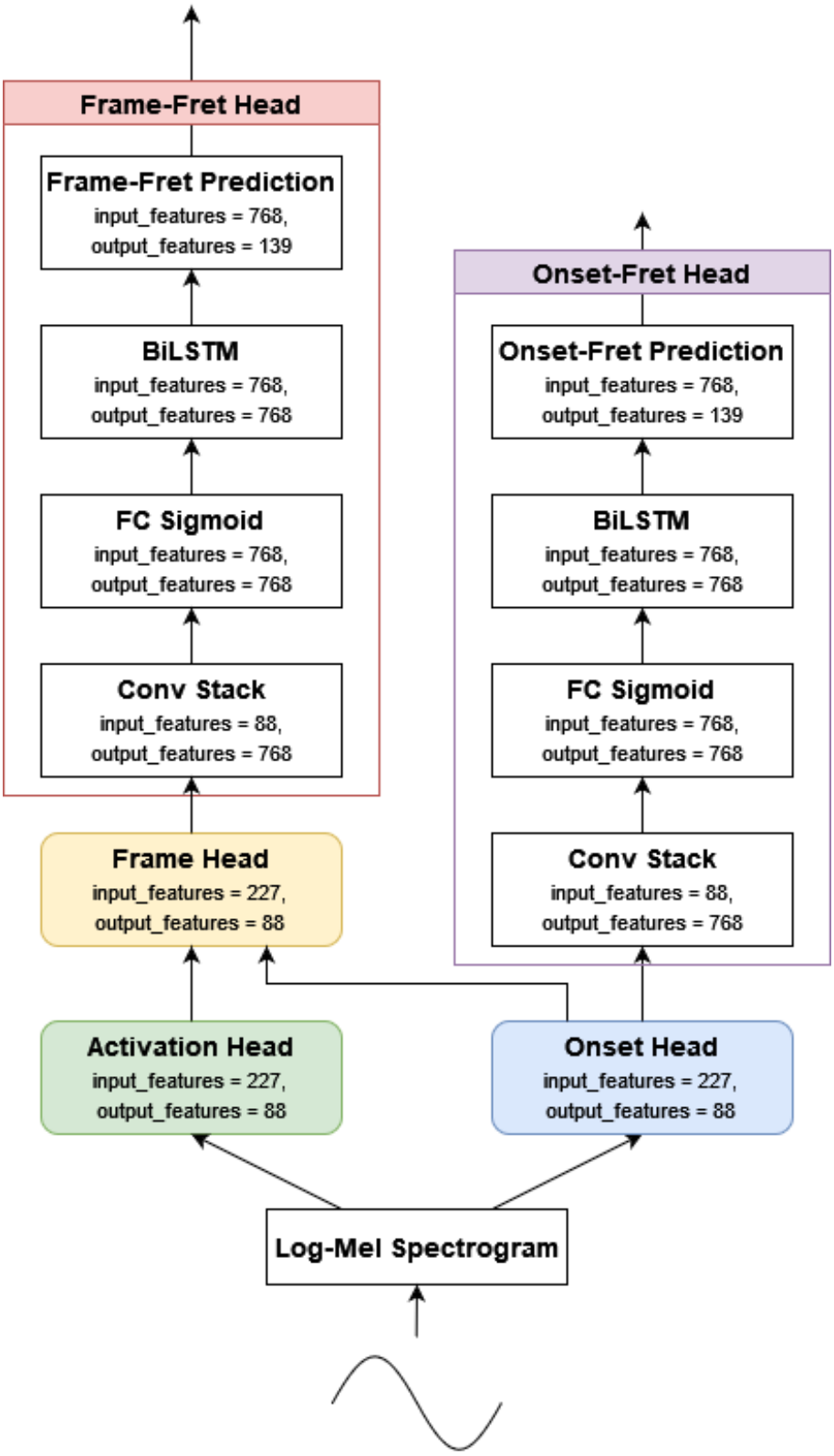
Το frame activation head αποτελείται από ένα ξεχωριστό ακουστικό μοντέλο ConvStack, η έξοδος του οποίου συνενώνεται με την έξοδο του onset head, με το αποτέλεσμα να διέρχεται από ένα νέο LSTM διπλής κατεύθυνσης. Με αυτό τον τρόπο, χρησιμοποιούμε την πληροφορία σχετικά με την αναγνώριση των onsets για τον υπολογισμό των διαρκειών.

4.3.2 Τροποποιημένα Μοντέλα

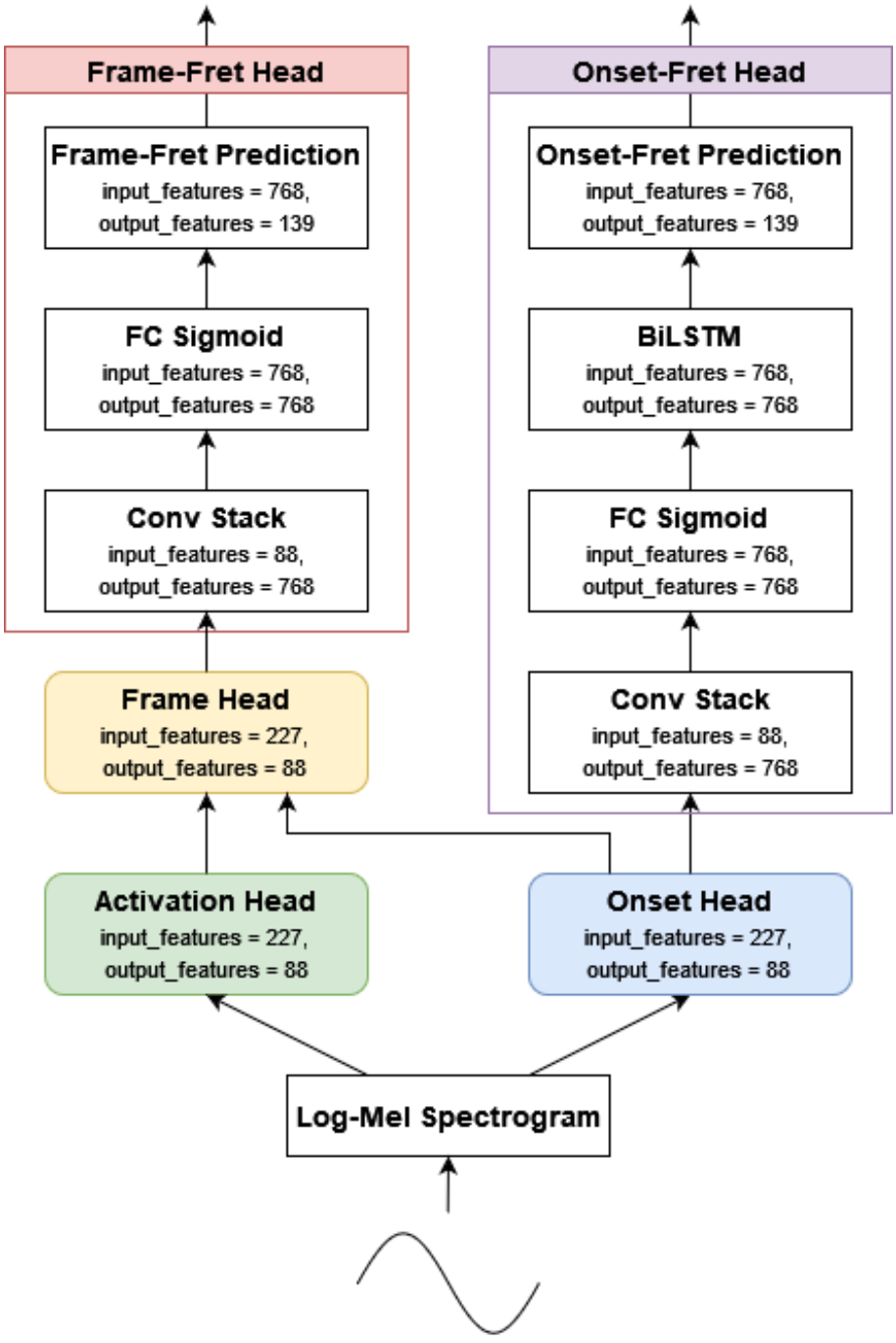
Για τη βελτίωση της επίδοσης του απλού μοντέλου, αναπτύχθηκαν επιπλέον αρχιτεκτονικές. Η βασική ιδέα είναι να χρησιμοποιήσουμε την υπάρχουσα αρχιτεκτονική που παρουσιάζεται στις δημοσιεύσεις [1], [2] και [3] και στη συνέχεια να γίνει μετατροπή της αναπαράστασης του πιάνου σε αναπαράσταση κιθάρας. Προς αυτό, προσθέτουμε στις εξόδους του μοντέλου επιπλέον heads, τα οποία εκτελούν αυτή τη μετατροπή. Στα σχήματα 4.2, 4.3, 4.4 παρουσιάζονται οι αρχιτεκτονικές που δοκιμάστηκαν.



Σχήμα 4.2: *Μουτέλο OAF Dual Conv.*



Σχήμα 4.3: *Μοντέλο OAF Dual Recc.*



Σχήμα 4.4: *Μοντέλο OAF Dual Mixed.*

4.4 Εκπαίδευση των Δικτύων

4.4.1 Προ-επεξεργασία δεδομένων

Η εκπαίδευση των αναδρομικών νευρωνικών δικτύων για μεγάλες ακολουθίες είναι χρονοβόρα διαδικασία, οπότε μεταβιβάζουμε ως είσοδο στο νευρωνικό κομμάτια 10 δευτερολέπτων από το αρχικό ηχητικό κομμάτι. Στην ενότητα 5.2.4, παρουσιάζουμε πως ελέγχθηκε και διαφορετικό μήκος κομματιού, χωρίς να προκαλέσει αισθητή διαφορά.

Για την προ-επεξεργασία του Dataset, χρησιμοποιήθηκαν οι κλάσεις Dataset και DataLoader του pytorch. Με τη βοήθεια της πρώτης κλάσης ορίζονται τα στοιχεία του dataset και ο τρόπος πρόσβασης σε αυτά, ενώ με τη δεύτερη κλάση παρέχεται ένας εύκολος τρόπος προσπέλασης τους.

Στην κλάση του Dataset ορίζουμε διάφορες μεθόδους που χρησιμοποιούμε για την υλοποίηση του. Η πρώτη και βασική μέθοδος είναι η load(audio_path, annotation_path), η οποία φορτώνει τα δεδομένα ήχου και τις σημειώσεις. Αρχικά, διαβάζεται από το audio_path με τη συνάρτηση read() της βιβλιοθήκης soundfile ο ήχος εισόδου και αποθηκεύεται στο αντίστοιχο πίνακα audio, ενώ καταγράφεται και ο ρυθμός δειγματοληψίας sr. Φροντίζουμε οι τιμές αυτές να κυμαίνονται μεταξύ [-1, 1], κάνοντας κοινωνικοποίησή στα δεδομένα.

Το επόμενο που πρέπει να γίνει, είναι να αναπαραστήσουμε τις σημειώσεις σε μορφή κατάλληλη για την είσοδο στο νευρωνικό. Συγκεκριμένα, θέλουμε δύο πίνακες onset_label[n_steps, n_features] και frame_label[n_steps, n_features]. Το μέγεθος των πινάκων εξαρτάται από τα χαρακτηριστικά εισόδου και το μήκος του κομματιού. Ορίζουμε ως υποδιαίρεση του χρόνου τα 32ms, επομένως, τα μεγέθη των πινάκων είναι

$$\begin{aligned} n\_steps &= (audio\_length - 1)//HOP\_LENGTH + 1 \\ n\_features &= MAX\_FEATURE - MIN\_FEATURE + 1 \end{aligned} \tag{4.2}$$

όπου HOP_LENGTH ορίζεται με βάση το ρυθμό δειγματοληψίας ώστε να αντιπροσωπεύει απόσταση μεταξύ frames 32 ms ως

$$HOP\_LENGTH = SAMPLE\_RATE * 32//1000 \tag{4.3}$$

και τα MAX_FEATURE και MIN_FEATURE εξαρτόνται από τον τύπο της εισόδου. Στην περίπτωση που χρησιμοποιούμε νότες MIDI, λαμβάνουν τιμές MAX_FEATURE = 88 και MIN_FEATURE = 22, δηλαδή όσες οι νότες του πιάνου, ενώ στην περίπτωση των stringfrets λαμβάνουν τιμές από 0 έως 138.

Οι πίνακες αυτοί θέλουμε να έχουν την ακόλουθη μορφή:

- Πίνακας onset_label: τιμή 1 στο στοιχείο onset_label[t, f] αν η νότα f ξεκινά τη στιγμή t, 0 αλλιώς.

- Πίνακας frame_label: τιμή 1 στο στοιχείο frame_label[t, f] αν η νότα f είναι ενεργή τη στιγμή t, 0 αλλιώς.

Πίνακας 4.2: *Παράδειγμα αναπαράστασης στοιχείων εισόδου.*

Πίνακας 4.3: *onset_label*

frames/features	0	1	2	3	4	5
5	0	0	0	0	0	0
4	0	0	0	0	0	0
3	0	0	0	1	0	0
2	0	0	0	0	0	0
1	0	1	0	0	0	0
0	0	0	0	0	0	0

Πίνακας 4.4: *frame_label*

frames/features	0	1	2	3	4	5
5	0	0	0	0	0	0
4	0	0	0	0	0	0
3	0	0	0	1	1	0
2	0	0	0	0	0	0
1	0	1	1	1	1	0
0	0	0	0	0	0	0

Στον πίνακα 4.2 φαίνεται ένα απλοποιημένο παράδειγμα για την καλύτερη κατανόηση της αναπαράστασης της εισόδου. Με βάση αυτούς τους δύο πίνακες, συμπεραίνουμε πως η νότα 1 ξεκινά το frame 1 και διαρκεί άλλα 3 frames, έως το frame 4, καθώς και η νότα 3 ξεκινάει το frame 3 και διαρκεί άλλο 1 frame, έως το frame 4. Για τη μετατροπή των δεδομένων εισόδου από την αναπαράσταση των χαρακτηριστικών που παρουσιάστηκε στην επιθυμητή αναπαράσταση χρησιμοποιείται ο αλγόριθμος 4.1.

Αλγοριθμος 4.1: *Μετατροπή δεδομένων εισόδου σε μορφή πινάκων onset_label, frame_label*

Είσοδος :annotations, n_steps, n_features, MIN_FEATURE
Έξοδος :onset_label, frame_label

Έστω πίνακες $onset\_label[n\_steps, n\_features]$ και $frame\_label[n\_steps, n\_features]$
for *onset, offset, note* in *annotation* **do**
  $left \leftarrow \text{round}(onset * SAMPLE\_RATE/HOP\_LENGTH)$
  $onset\_right \leftarrow \min(n\_steps, left + HOPS\_IN\_ONSET)$
  $frame\_right \leftarrow \text{round}(offset * SAMPLE\_RATE/HOP\_LENGTH)$
  $frame\_right \leftarrow \min(n\_steps, frame\_right)$
  $offset\_right \leftarrow \min(n\_steps, frame\_right + HOPS\_IN\_ONSET)$

  $f \leftarrow \text{round}(note) - MIN\_FEATURE$
  $onset\_label\,[left \ldots onset\_right, f] \leftarrow 1$
  $frame\_label\,[frame\_right \ldots offset\_right, f] \leftarrow 1$
end for
return *onset_label, frame_label*

Με τα παραπάνω ολοκληρώνεται η υλοποίηση της μεθόδου load(), η οποία καλείται μόνο κατά την αρχικοποίηση του dataset. Το τελευταίο που πρέπει να γίνει για την προετοιμασία του dataset είναι η υλοποίηση της μεθόδου που ορίζει τον τρόπο λήψης δεδομένων από τον dataloader. Όπως αναφέρθηκε, σαν είσοδο στο δίκτυο δίνουμε κομμάτια 10 δευτερολέπτων, επομένως κάθε φορά που λαμβάνεται ένα τυχαίο τμήμα 10 δευτερολέπτων. Με αυτό τον τρόπο, κατά την εκπαίδευση, το δίκτυο θα δει ολόκληρο το κομμάτι σε διαφορετικές εποχές, δρώντας με αυτό τον τρόπο σαν ένα είδος cross-validation.

Ο διαχωρισμός του Dataset έγινε με τον ακόλουθο τρόπο

- Training Dataset 80%
- Validation Dataset 10%

- Test Dataset 10%

4.4.2 Βρόχος εκπαίδευσης

Στην ενότητα αυτή θα περιγράψουμε τα βήματα που έγιναν για την εκπαίδευση των δικτύων. Αρχικά, υπενθυμίζουμε πως η έξοδος των δύο κεφαλών του νευρωνικού είναι πίνακες της μορφής $out[n\_features, n\_steps]$, με στοιχεία κανονικοποιημένα στο διάστημα [0,1] λόγω της Softmax, τα οποία αντιπροσωπεύουν πιθανότητες. Η συνάρτηση κόστους (loss function) που χρησιμοποιούμε στην εκπαίδευση είναι η δυαδική Cross-Entropy Loss.

Η συνάρτηση binary Cross-Entropy Loss ορίζεται για ένα διάνυσμα πιθανοτήτων εξόδου $\mathbf{p}$ και ένα δείγμα $\mathbf{y}$ ως

$$CE(y, p) = -\frac{1}{N}\left[\sum_{j=1}^{N}\left[t_j \log(p_j) + (1 - t_j)\log(1 - p_j)\right]\right] \tag{4.4}$$

Επομένως, η συνολική συνάρτηση σφάλματος προκύπτει από το άθροισμα των επιμέρους σφαλμάτων στις δύο κεφαλές εξόδου. Οι συναρτήσεις σφάλματος φαίνονται στις σχέσεις 4.5 - 4.7, όπου CE η συνάρτηση cross-entropy loss, f_min και f_max το ελάχιστο και μέγιστο χαρακτηριστικό, T το πλήθος των frames, $\mathbf{P}$ οι πιθανότητες εξόδου των νευρωνικών και $\mathbf{Y}$ οι τιμές των δειγμάτων.

$$L_{total} = L_{onset} + L_{frame} \tag{4.5}$$

$$L_{onset} = \sum_{p=f\_min}^{f\_max}\sum_{t=0}^{T} CE(\mathbf{Y}_{onset}(t,p), \mathbf{P}_{onset}(t,p)) \tag{4.6}$$

$$L_{frame} = \sum_{p=f\_min}^{f\_max}\sum_{t=0}^{T} CE(\mathbf{Y}_{frame}(t,p), \mathbf{P}_{frame}(t,p)) \tag{4.7}$$

Για τη φόρτωση των δεδομένων και τον διαχωρισμό σε train/validation/test sets χρησιμοποιείται η βιβλιοθήκη του pytoch DataLoader. Η εκπαίδευση των νευρωνικών, και ιδιαίτερα των αναδρομικών δικτύων απαιτεί πολύ χρόνο, οπότε μεταβιβάζουμε τα δεδομένα εισόδου σε batches μεγέθους 8. Ο αρχικός ρυθμός μάθησης (learning rate) ορίζεται 0.0006. Για την επιτάχυνση της εκπαίδευσης χρησιμοποιήθηκαν επιπλέον βελτιστοποιήσεις

- Optimizer : Αποτελεί τεχνική βελτιστοποίησης του αλγορίθμου κατάβασης κλίσης, βοηθώντας στη σύγκλιση. Επιλέχθηκε ο ADAM optimizer, ο οποίος αποτελεί μία από τις πιο κοινές επιλογές σε προβλήματα regression.

- Scheduler : Μείωση του ρυθμού μάθησης κάθε συγκεκριμένο αριθμό εποχών. Αυτό βοηθάει στο να έχουμε αρχικά σχετικά γρήγορη βελτίωση του μοντέλου, ενώ στη συνέχεια που οι αλλαγές είναι μικρότερες, να είναι αισθητές, μέσω ενός μικρότερου ρυθμού μάθησης. Για το σκοπό αυτό χρησιμοποιήθηκε η κλάση torch.optim.lr_scheduler.StepLR του pytorch.

- Gradient Clipping : Όπως αναφέρθηκε στην ενότητα 3.3.1, ένα σημαντικό πρόβλημα στα αναδρομικά νευρωνικά δίκτυα είναι το exploding gradient. Για να το αποφύγουμε αυτό, μπορούμε κάθε φορά που η παράγωγος βγαίνει εκτός ορίων, να την επανακλιμακώσουμε στα επιθυμητά επίπεδα.

Όσον αφορά το βρόχο της εκπαίδευσης σε κάθε εποχή διαβάζουμε το αρχείο ήχου, το μετατρέπουμε σε μορφή φασματογραφήματος mel, και μεταβιβάζουμε την είσοδο στο μοντέλο. Στη συνέχεια, συγκρίνουμε τις εξόδους του μοντέλου με τα δείγματα εισόδου με βάση τις σχέσεις 4.5 - 4.7, και με βάση τα σφάλματα αυτά ανανεώνουμε τα βάρη.

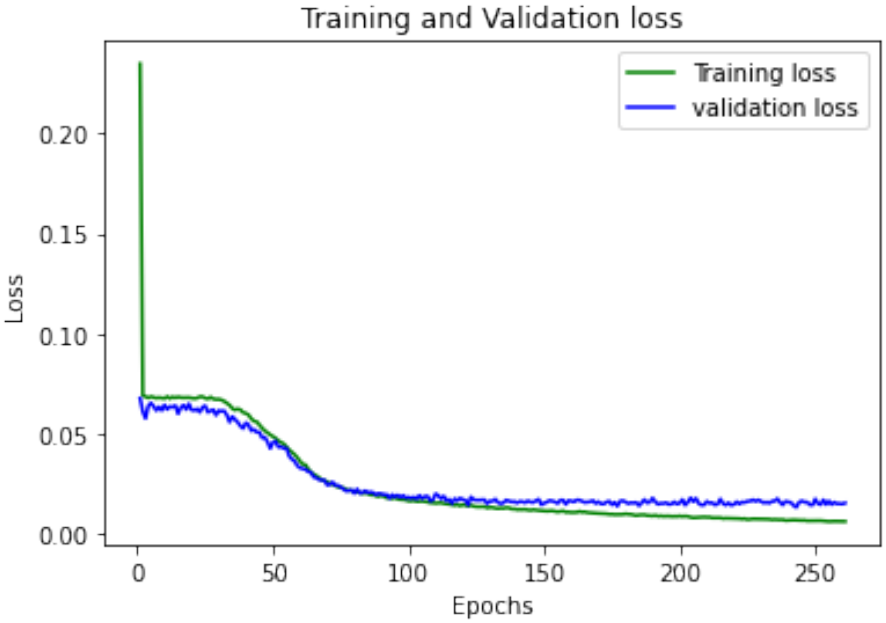
4.4.3 Early Stopping

Όσον αφορά τις εποχές, αρχικά έγινε προσπάθεια να γίνει εκπαίδευση στις 10.000 εποχές, όπως στη δημοσίευση [1], αλλά παρατηρήθηκε σύντομα πως εμφανίζεται αρκετά νωρίς το φαινόμενο overfitting. Ο βασικός λόγος που γίνεται αυτό είναι το πολύ μικρότερο μέγεθος του dataset το οποίο χρησιμοποιούμε. Χρησιμοποιήθηκαν αρκετοί τρόποι αντιμετώπισης του φαινομένου αυτού, οι οποίοι θα αναλυθούν στη συνέχεια, αλλά ο βασικός ήταν το Early Stopping.

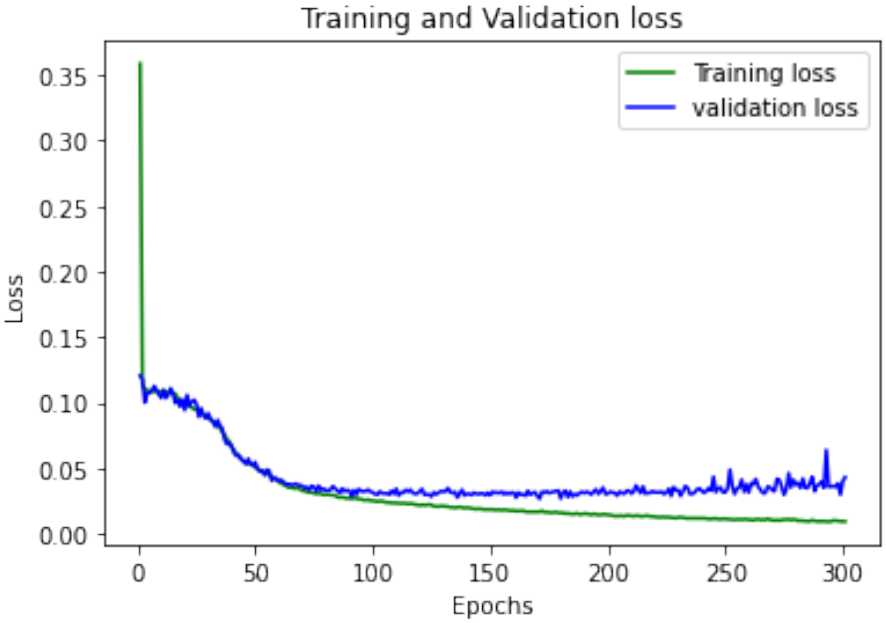
Για την εφαρμογή του Early Stopping, παρακολουθούμε το Validation Loss και όταν παρατηρηθεί ένα διάστημα στο οποίο η διαφορά μεταξύ διαδοχικών εποχών δε μειώνεται πάνω από ένα συγκεκριμένο όριο, σταματάμε τη διαδικασία της εκπαίδευσης. Το όριο αυτό εξαρτάται από το μοντέλο, καθώς πολλά συγκλίνουν πιο γρήγορα ή πιο αργά. Σε κάθε περίπτωση χρησιμοποιήθηκε μια callback κλάση Early Stopping που επιτελεί αυτή τη λειτουργία.

4.4.4 Γραφικές Παραστάσεις Εκπαίδευσης

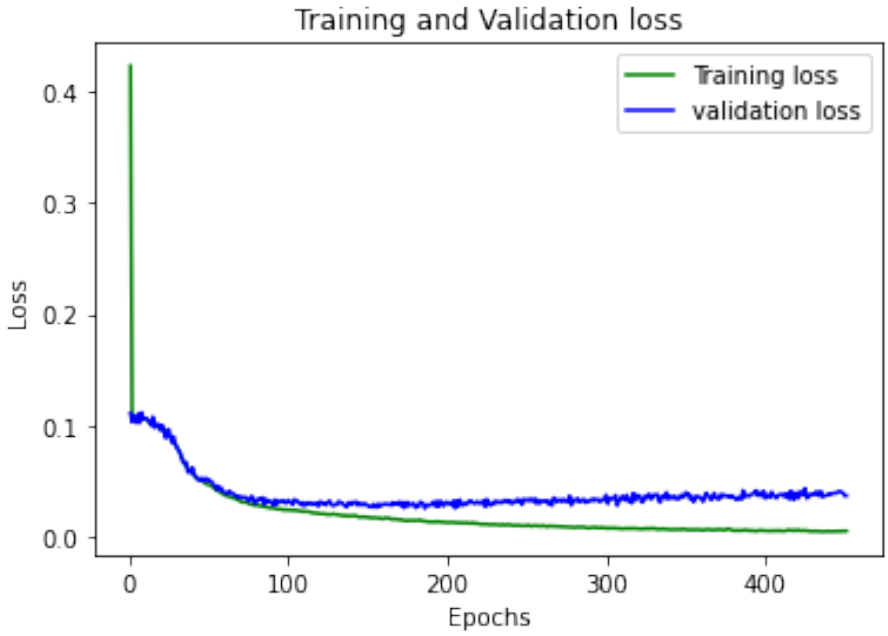
Σε αυτή την ενότητα παρουσιάζονται σε γραφικές παραστάσεις οι πρόοδοι της εκπαίδευσης. Στον οριζόντιο άξονα απεικονίζονται οι εποχές, ενώ στον κατακόρυφο απεικονίζονται τα losses (training/validation). Παρατηρούμε πως σε όλες τις περιπτώσεις εμφανίζεται το φαινόμενο overfitting.



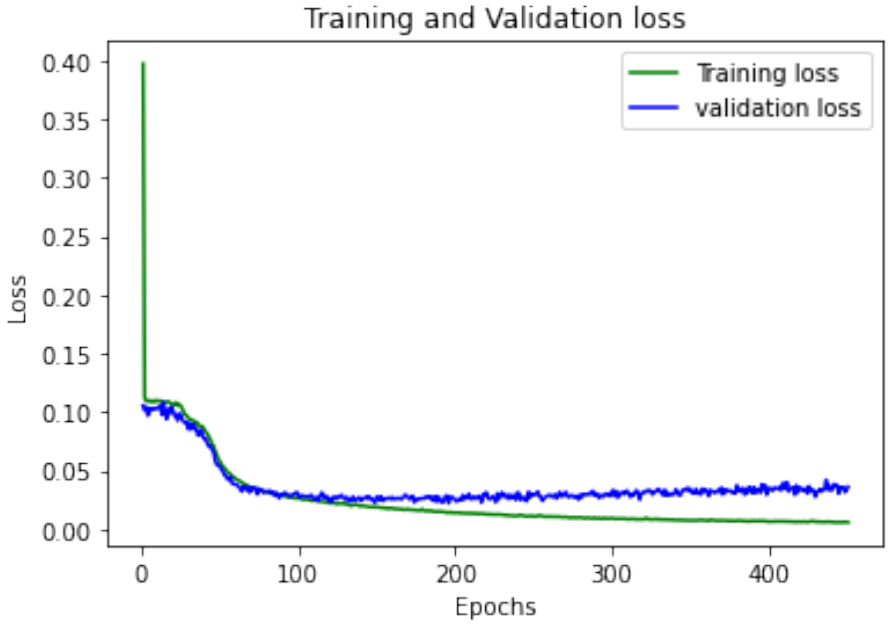
Σχήμα 4.5: *Γραφική Παράσταση Μοντέλου OAF Org.*



Σχήμα 4.6: *Γραφική Παράσταση Μοντέλου OAF Conv.*



Σχήμα 4.7: *Γραφική Παράσταση Μοντέλου OAF Recc.*



Σχήμα 4.8: *Γραφική Παράσταση Μοντέλου OAF Mixed.*

4.5 Τεχνικές Βελτιστοποίησης και Αποφυγής Overfitting

Όπως φαίνεται από τις παραπάνω γραφικές παραστάσεις, το φαινόμενο του overfitting εμφανίζεται αρκετά σύντομα στην διαδικασία της εκπαίδευσης. Για την αντιμετώπιση αυτού εφαρμόστηκαν κάποιες τεχνικές, οι οποίες επηρεάζουν τόσο το ίδιο το μοντέλο, όσο και το dataset.

4.5.1 Data Augmentation

Η πρώτη προσπάθεια αντιμετώπισης του overfitting έγινε μέσω της εφαρμογής του Data Augmentation. Ο βασικός λόγος που έγινε αυτό είναι πως το Dataset μας είναι αρκετά μικρό, οπότε θα βοηθούσε αν μπορούσαμε να αντλήσουμε επιπλέον πληροφορία από αυτό.

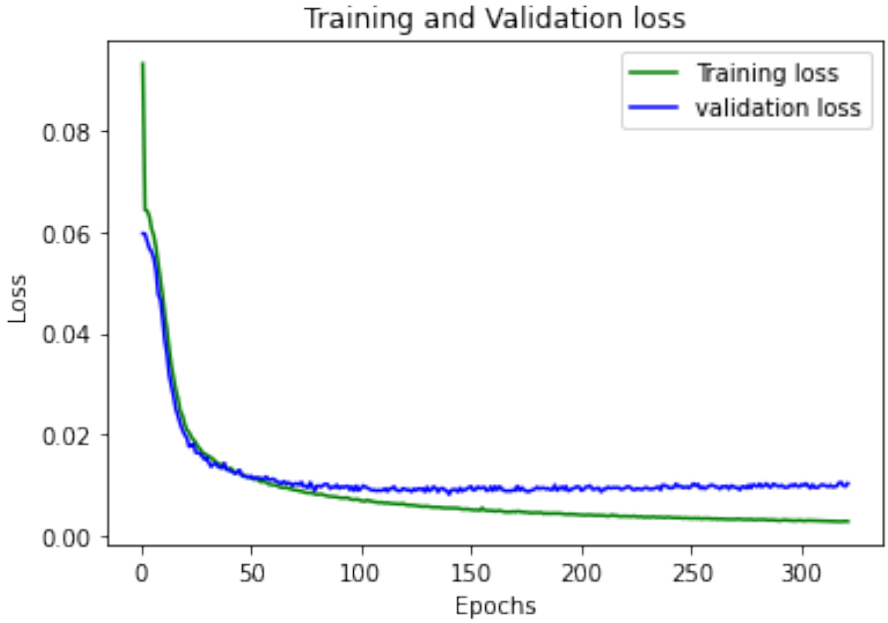
Το βασικό υπόβαθρο που οδηγεί στην επιπλέον πληροφορία είναι το γεγονός πως εκτός κάποιων εξαιρέσεων, ένα κομμάτι παιγμένο στην κιθάρα μπορεί να παιχθεί μετακινώντας όλες τις νότες μια συγκεκριμένη απόσταση πάνω ή κάτω στην ταστιέρα, και να προκύψει ένα ουσιαστικά νέο σύνολο δειγμάτων από το ίδιο κομμάτι. Αυτό ονομάζεται στη μουσική μετατόπιση ή αλλαγή κλειδιού και είναι μια αρκετά ευρέως χρησιμοποιούμενη τεχνική, η οποία χρησιμοποιείται συνήθως για την προσαρμογή της μουσικής στο εύρος ενός τραγουδιστή.

Στην κιθάρα, το πρόβλημα είναι λίγο πιο περίπλοκο, καθώς πρέπει να λάβουμε υπόψιν τις ανοιχτές χορδές. Για το λόγο πως δεν μπορούμε να μετακινήσουμε μια νότα που παιζόταν σε ανοιχτή χορδή πιο ¨κάτω¨, χωρίς να αλλάξουμε τον τρόπο που παίζεται το κομμάτι, πρέπει να φροντίσουμε όλες οι μεταφορές να ισχύουν μόνο όταν δεν μεταφέρουν νότες σε άλλη χορδή. Επιπλέον, έχουμε κάνει την θεώρηση πως η κιθάρα έχει 22 τάστα, οπότε μια μεταγραφή που θα περιελάμβανε τάστα πάνω από το 22ό δεν είναι έγκυρη.

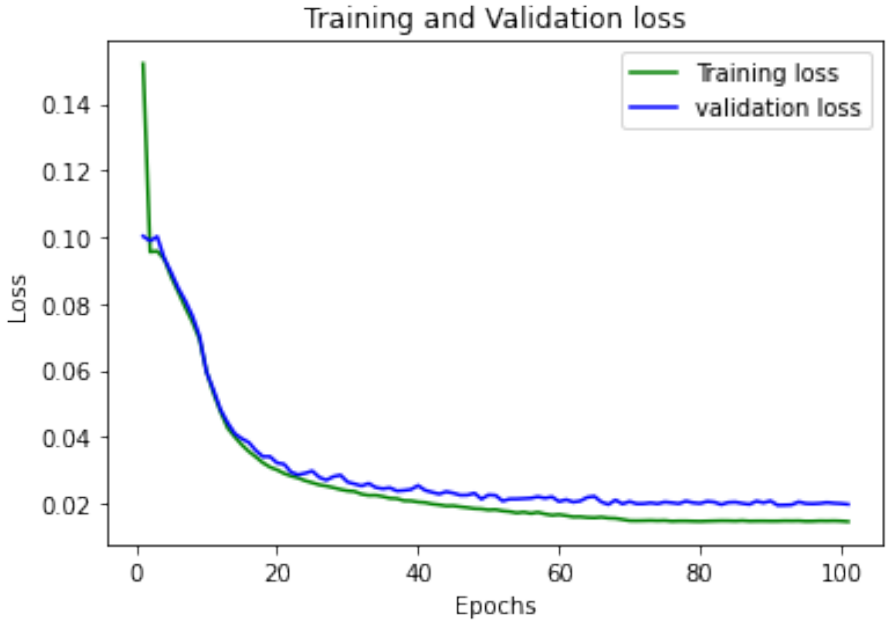
Με το σκεπτικό αυτό, δημιουργήθηκαν νέα δείγματα για το dataset. Διατρέχουμε τις σημειώσεις του κομματιού και βρίσκουμε όλες τις δυνατές μεταφορές που μπορούν να προκύψουν ώστε καμία νότα να μην παραβιάζει τους κανόνες που αναφέρθηκαν σε καμία μεταφορά. Σαν πρώτο βήμα, μετατρέπουμε όλες τις σημειώσεις, τόσο για τις νότες MIDI, όσο και για τα stringfrets για όλα τα έγκυρα διαστήματα. Στη συνέχεια, γίνεται η μετατροπή του κομματιού ήχου. Δοκιμάστηκαν διάφορες βιβλιοθήκες που επιτελούν αυτό τον σκοπό, και βρέθηκε ότι αυτή που δίνει τον πιο πιστό ήχο μετά τη μετατροπή ήταν η pitch_shift της βιβλιοθήκης pyrubberband.

Με την παραπάνω διαδικασία, προστέθηκαν στα αρχικά 360 δείγματα, 1699 νέα δείγματα, δηλαδή αυξήθηκε το μέγεθος του dataset κατά 571.9%. Ακολουθήθηκε η ίδια διαδικασία εκπαίδευσης και στη συνέχεια παρουσιάζονται οι γραφικές παραστάσεις εκπαίδευσης.

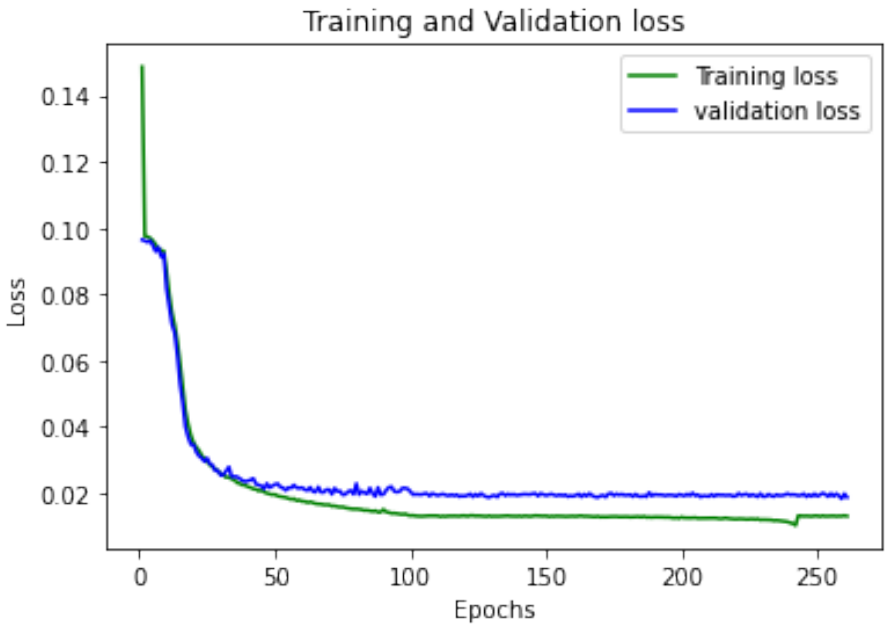
Παρατηρούμε ότι σε όλες τις περιπτώσεις ο αριθμός των εποχών στον οποίο ξεκινάει το overfitting δεν αλλάζει αισθητά. Παρόλα αυτά, πρέπει να αναλογιστούμε πως πλέον υπάρχουν σχεδόν έξι φορές περισσότερα δεδομένα, με αποτέλεσμα σε κάθε εποχή το δίκτυο να δέχεται πολύ περισσότερα δεδομένα. Επομένως συνολικά, η εκπαίδευση βελτιώνεται αισθητά, και δεν οδηγείται το δίκτυο σε κορεσμό το ίδιο γρήγορα.



Σχήμα 4.9: *Γραφική Παράσταση Μοντέλου OAF Org με Augmented Dataset.*



Σχήμα 4.10: *Γραφική Παράσταση Μοντέλου OAF Recc με Augmented Dataset.*



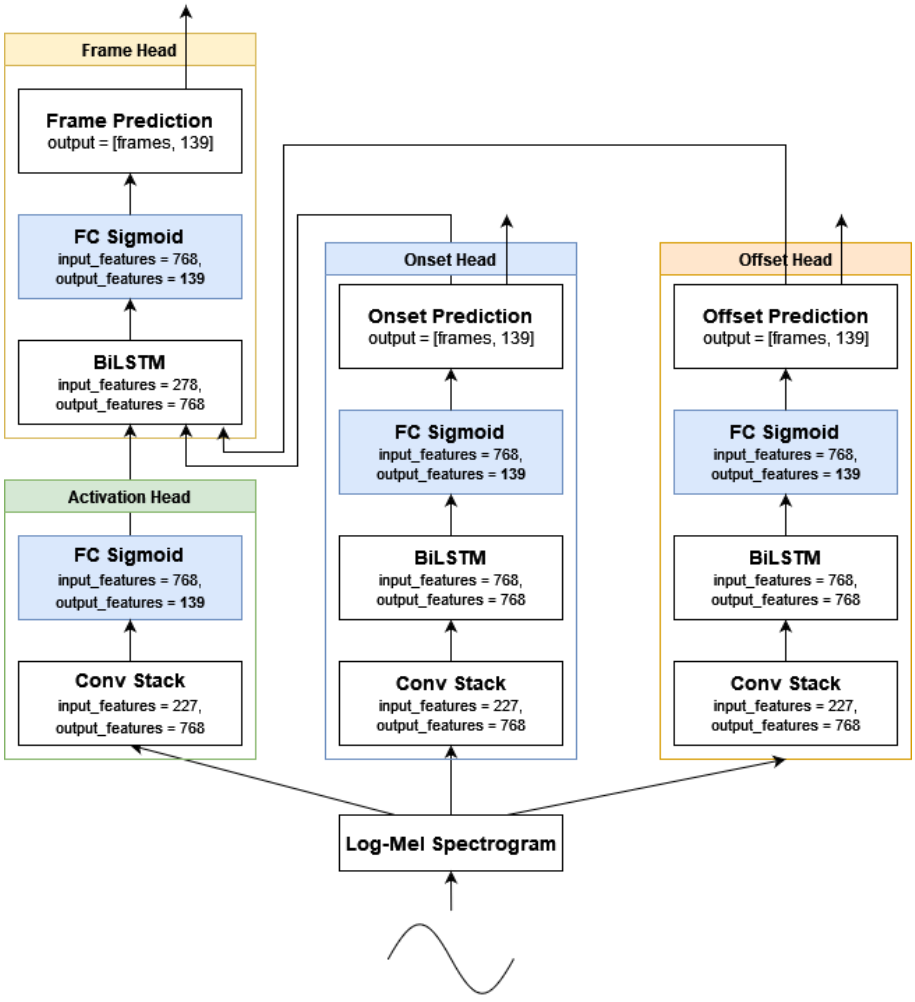
Σχήμα 4.11: *Γραφική Παράσταση Μοντέλου OAF Mixed με Augmented Dataset.*

4.5.2 Μεταφορά Μάθησης από Εκπαιδευμένο Μοντέλο

Μια άλλη τεχνική που χρησιμοποιήθηκε είναι η μεταφορά μάθησης από προ-εκπαιδευμένο μοντέλο. Η μεταφορά μάθησης βασίζεται στην ιδέα πως μπορούμε να χρησιμοποιήσουμε ένα μοντέλο που είναι ήδη εκπαιδευμένο σε ένα πρόβλημα σχετικό με το δικό μας, ως αρχή για να έχουμε καλύτερα αποτελέσματα κατά την εκπαίδευση.

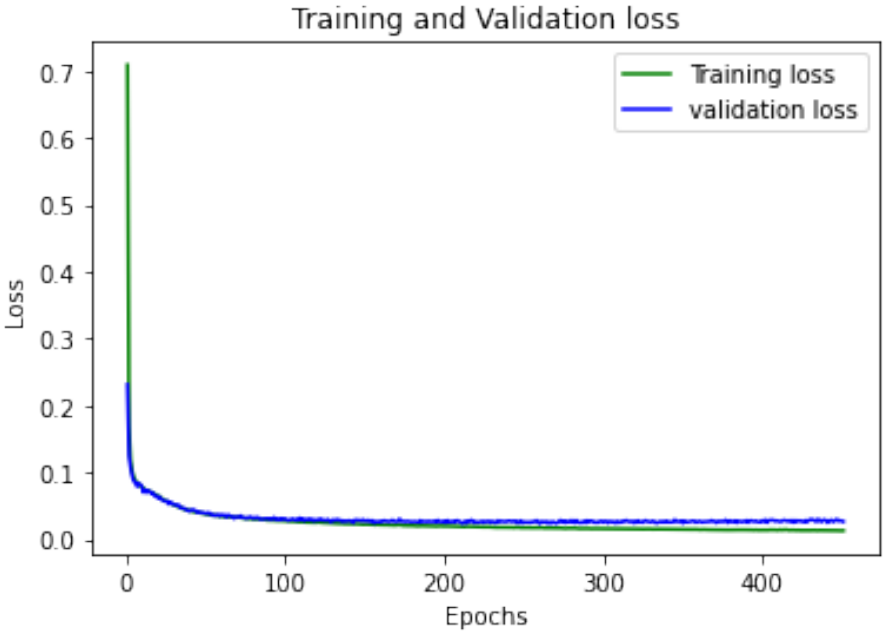
Για το προ-εκπαιδευμένο μοντέλο, χρησιμοποιήθηκε η υλοποίηση του Onsets and Frames από τον Jong Wook Kim [12]. Η υλοποίηση αυτή διαφέρει ελάχιστα από τις δικές μας, στο γεγονός πως υπάρχει ένα επιπλέον Offset Head το οποίο ανιχνεύει τα offsets των νοτών, ενώ είναι εκπαιδευμένο σε μουσική πιάνου. Η χροιά της κιθάρας και του πιάνου δεν είναι ιδιαίτερα διαφορετικές, ενώ ο ήχος παράγεται με παρόμοιο τρόπο (χτύπημα σε χορδή), οπότε περιμένουμε πως η εκπαίδευση που έχει ήδη γίνει θα είναι χρήσιμη.

Για την προσαρμογή στα δικά μας δεδομένα, αλλάζουμε τα επίπεδα που αφορούν τα features του πιάνου με stringfrets, διατηρώντας τα βάρη στα υπόλοιπα εκπαιδευμένα επίπεδα. Το δίκτυο φαίνεται στο σχήμα 4.12, στο οποίο τα σκιαγραφημένα κελιά είναι αυτά που αλλάχθηκαν, ενώ τα υπόλοιπα παρέμειναν ίδια.



Σχήμα 4.12: *Αρχιτεκτονική Μοντέλου OAF Pre Trained.*

Η γραφική παράσταση της εκπαίδευσης του νευρωνικού φαίνεται στο σχήμα 4.13.



Σχήμα 4.13: *Εκπαίδευση Μοντέλου OAF Pre Trained.*

Κεφάλαιο 5

Αξιολόγηση Μοντέλων

Στο κεφάλαιο αυτό περιγράφεται η αξιολόγηση των μοντέλων που αναπτύχθηκαν και τα συμπεράσματα που προκύπτουν από τα πειράματα. Αναλύονται επιπλέον οι μετρικές που χρησιμοποιήθηκαν για την αξιολόγηση.

5.1 Μετρικές

5.1.1 Βασικές Μετρικές Νευρωνικών Δικτύων

Στα προβλήματα ταξινόμησης, για να κατανοήσουμε την απόδοση ενός δικτύου, κατηγοριοποιούμε τις προβλέψεις σε 4 κατηγορίες που φαίνονται στον πίνακα 5.1. Τα true/false positive/negative αναφέρονται στην κατηγοριοποίηση με δύο κλάσεις.

Πρόβλεψη	Πραγματική Τιμή	Ονομασία	Εξήγηση
1	1	True Positive (TP)	Θετική Πρόβλεψη ήταν Θετική
0	1	True Negative (TN)	Θετική Πρόβλεψη ήταν Αρνητική
1	0	False Positive (FP)	Αρνητική Πρόβλεψη ήταν Θετική
0	0	False Negative (FN)	Αρνητική Πρόβλεψη ήταν Αρνητική

Πίνακας 5.1: *Κατηγοριοποίηση αποτελεσμάτων Ταξινόμησης*

Με βάση αυτά ορίζονται οι βασικές μετρικές που χρησιμοποιούνται για την αξιολόγηση της ταξινόμησής. Το F1-score είναι η βασική μετρική που χρησιμοποιούμε για την αξιολόγηση, και αποτελεί τον αρμονικό μέσο όρο των δύο άλλων μετρικών.

- Accuracy = $\frac{TP+TN}{TP+TN+FP+FN}$ = $\frac{\text{πλήθος σωστών προβλέψεων}}{\text{πλήθος δειγμάτων}}$
- Precision = $\frac{TP}{TP+FP}$ = $\frac{\text{πλήθος σωστών θετικών προβλέψεων}}{\text{πλήθος θετικών προβλέψεων}}$
- Recall = $\frac{TP}{TP+FN}$ = $\frac{\text{πλήθος σωστών θετικών προβλέψεων}}{\text{πλήθος θετικών δειγμάτων}}$
- F1-Score = $2 * \frac{\text{Precission}*\text{Recall}}{\text{Precission}+\text{Recall}}$

5.1.2 Μετρικές Αξιολόγησης Μουσικής

Για την αξιολόγηση των αποτελεσμάτων των δικτύων, χρησιμοποιήθηκαν μετρικές που παρέχονται από την βιβλιοθήκη mir_eval ([13], [14]). Οι μετρικές αυτές βασίζονται στις

μετρικές που αναφέρθηκαν στην προηγούμενη ενότητα, αλλά είναι προσαρμοσμένες στην επεξεργασία ήχου.

Για την αναπαράσταση των νοτών χρησιμοποιείται το piano roll representation, δηλαδή αναπαράσταση σε νότες MIDI, το οποίο εύκολα προσαρμόζεται για την αναπαράσταση stringfret. Δεδομένων δύο συνόλων νοτών, υπολογίζεται το πλήθος των νοτών που ταιριάζουν με τις νότες αναφοράς και το πλήθος που δεν ταιριάζουν. Με βάση αυτά, υπολογίζονται τα precission, recall, f1-score. Ο ορισμός του πότε δύο νότες είναι ίδιες, ορίζεται στο [14].

Ο υπολογισμός αυτός γίνεται με δύο τρόπους. Σαν πρώτη περίπτωση, μια νότα θεωρείται σωστή αν το onset της βρίσκεται σε εύρος $+-50ms$ από τη νότα αναφοράς και η θεμελιώδης συχνότητα της νότας (F0) βρίσκεται σε απόσταση ενός τετάρτου του τόνου από τη νότα αναφοράς (σε αυτό το στάδιο δεν εξετάζεται η διάρκεια). Σε μια δεύτερη περίπτωση, απαιτείται να ισχύουν τα παραπάνω, και επιπλέον η διάρκεια της νότας να έχει το πολύ σφάλμα 20 % από τη νότα αναφοράς, ή το offset να βρίσκεται σε εύρος $+-50ms$ από τη νότα αναφοράς, όποιο είναι μεγαλύτερο.

Επομένως, για τις μετρικές αξιολόγησης των νοτών, χρησιμοποιούμε δύο βασικές μετρικές:

- metric/stringfret/f1 : Μέθοδος mir_eval.transcription.precision_recall_f1_overlap, με επιλογή offset_ratio απενεργοποιημένη, το οποίο αναφέρεται μόνο στα onsets των νοτών και όχι στα offsets.

- metric/stringfret-with-offsets/f1 : Μέθοδος mir_eval.transcription.precision_recall_f1_overlap, με επιλογή offset_ratio ενεργοποιημένη, το οποίο λαμβάνει υπόψιν και τα offsets των νοτών.

Για την αξιολόγηση των frames χρησιμοποιείται μια επιπλέον μετρική, το mir_eval.multipitch, το οποίο αποσκοπεί στην αξιολόγηση κομματιών με περισσότερες από μία θεμελιώδεις συχνότητες σε κάθε frame (όπως προφανώς η μουσική της κιθάρας). Για αυτό το λόγο χρησιμοποιείται το metric/stringfret-frame/f1, το οποίο προκύπτει από τις αντίστοιχες μετρικές.

5.2 Αξιολόγηση των Μοντέλων

5.2.1 Αξιολόγηση Αρχικών Μοντέλων

Σε αυτή την ενότητα παρουσιάζονται τα αποτελέσματα των αρχικών πειραμάτων για τα διάφορα μοντέλα. Οι συνολικές μετρικές προκύπτουν από τον μέσο όρο των μετρικών για κάθε δείγμα του Test Dataset.

Μετρική	F1	Precission	Recall
stringfret	**0.956**	0.973	0.941
stringfret-with-offsets	**0.767**	0.779	0.757
stringfret-frame	**0.935**	0.887	0.936

Πίνακας 5.2: *Αποτελέσματα μοντέλου OAF Org*

Μετρική	F1	Precission	Recall
stringfret	**0.297**	0.638	0.210
stringfret-with-offsets	**0.003**	0.005	0.002
stringfret-frame	**0.048**	0.944	0.028

Πίνακας 5.3: *Αποτελέσματα μοντέλου OAF Conv*

Μετρική	F1	Precission	Recall
stringfret	**0.929**	0.934	0.928
stringfret-with-offsets	**0.010**	0.010	0.010
stringfret-frame	**0.186**	0.754	0.075

Πίνακας 5.4: *Αποτελέσματα μοντέλου OAF Recc*

Μετρική	F1	Precission	Recall
stringfret	**0.934**	0.942	0.929
stringfret-with-offsets	**0.010**	0.010	0.010
stringfret-frame	**0.190**	0.920	0.064

Πίνακας 5.5: *Αποτελέσματα μοντέλου OAF Mixed*

Στα πρώτα πειράματα παρατηρούμε ξεκάθαρα πως όλα τα μοντέλα εκτός από το OAF Conv έχουν εξίσου καλή απόδοση στην ανίχνευση των Onsets των νοτών. Το μοντέλο OAF Conv από την άλλη έχει πολύ κακή απόδοση σε όλες τις μετρικές, οπότε δεν χρησιμοποιηθεί σε περαιτέρω πειράματα.

Η μεγάλη διαφορά μεταξύ του αρχικού μοντέλου και των μοντέλων που μετατρέπουν την έξοδο MIDI σε έξοδο stringfret, είναι πως έχουν σημαντικά μικρότερο F1-score σε ότι αφορά τις διάρκειες των νοτών.

5.2.2 Αξιολόγηση Μοντέλων με Augmented Dataset

Μετρική	F1	Precission	Recall
stringfret	**0.957**	0.967	0.949
stringfret-with-offsets	**0.840**	0.834	0.848
stringfret-frame	**0.944**	0.998	0.962

Πίνακας 5.6: *Αποτελέσματα μοντέλου OAF Org*

Μετρική	F1	Precission	Recall
stringfret	**0.859**	0.905	0.822
stringfret-with-offsets	**0.010**	0.010	0.010
stringfret-frame	**0.179**	0.957	0.102

Πίνακας 5.7: *Αποτελέσματα μοντέλου OAF Recc*

Παρατηρούμε πως το η εκπαίδευση σε μεγαλύτερο dataset βελτίωσε σε μικρό βαθμό τα αποτελέσματα των μετρήσεων. Συγκεκριμένα, για το μοντέλο OAF Org, το F1-score για τη

Μετρική	F1	Precission	Recall
stringfret	**0.934**	0.942	0.929
stringfret-with-offsets	**0.010**	0.010	0.010
stringfret-frame	**0.190**	0.920	0.064

Πίνακας 5.8: *Αποτελέσματα μοντέλου OAF Mixed*

μετρική stringfret-with-offsets αυξήθηκε από 0.767 σε 0.840, το οποίο αποτελεί σημαντική αύξηση.

Τα αποτελέσματα για τα υπόλοιπα δίκτυα δείχνουν παρόμοια συμπεριφορά με τα προηγούμενα πειράματα. Αναγνωρίζονται σε πολύ καλό βαθμό τα onsets των νοτών, αλλά δεν υπάρχει σωστή αναγνώριση των offsets.

5.2.3 Αξιολόγηση Προεκπαιδευμένου Μοντέλου με Augmented Dataset

Στον ακόλουθο πίνακα παρουσιάζονται τα αποτελέσματα της εκπαίδευσης του προ-εκπαιδευμένου μοντέλου και της μεταφοράς μάθησης, στο αρχικό Dataset.

Μετρική	F1	Precission	Recall
stringfret	**0.657**	0.928	0.521
stringfret-with-offsets	**0.530**	0.755	0.419
stringfret-frame	**0.687**	0.966	0.556

Πίνακας 5.9: *Αποτελέσματα μοντέλου OAF Pre Trained*

Παρατηρούμε ότι το μοντέλο αυτό έχει σχετικά καλή απόδοση σε όλες τις μετρικές, όχι όμως εξίσου καλή με το μοντέλο OAF Org. Επιπλέον, δεν πετυχαίνει εξίσου καλή αναγνώριση των onsets με κανένα από τα υπόλοιπα μοντέλα, που παρόλο που έχουν δυσκολία στην αναγνώριση της διάρκειας, έχουν πολύ καλή απόδοση στην αναγνώριση των onsets. Αυτό το μοντέλο μπορεί να θεωρηθεί ως ένα ενδιάμεσο μεταξύ των δύο, χωρίς όμως να είναι ιδιαίτερα χρήσιμο για καμία από τις εφαρμογές μας.

5.2.4 Αξιολόγηση Μοντέλων με Ακολουθίες 20 Δευτερολέπτων

Μια σημαντική υπερπαράμετρος που επιλέχθηκε στην αρχή των πειραμάτων ήταν το μήκος των ακολουθιών εισόδου. Στη δημοσίευση [1], χρησιμοποιούνται ακολουθίες 20 δευτερολέπτων, αλλά λόγω των μεγάλων κομματιών που χρησιμοποιούνται στο dataset, θεωρήθηκε πως αυτή η επιλογή δεν είναι σωστή για το GuitarSet. Κάποια κομμάτια μάλιστα, έχουν διάρκεια μικρότερη από 20 δευτερόλεπτα. Επομένως, επιλέχθηκε να χρησιμοποιηθούν ακολουθίες 10 δευτερολέπτων.

Ένα τελευταίο πείραμα που έγινε ήταν ο έλεγχος της απόδοσης για ακολουθίες 20 δευτερολέπτων. Τα πειράματα εκτελέστηκαν μόνο για τα δύο καλύτερα σε απόδοση δίκτυα, OAF Original, OAF Mixed, με το αρχικό Dataset. Τα αποτελέσματα φαίνονται στους ακόλουθους πίνακες.

Και στις δύο περιπτώσεις, τα αποτελέσματα είναι σχεδόν ίδια για τις δύο αρχιτεκτονικές. Επομένως, το μήκος των ακολουθιών δεν είχε σημαντικό ρόλο στα αποτελέσματα της εκπα-

Μετρική	F1	Precission	Recall
stringfret	**0.952**	0.963	0.944
stringfret-with-offsets	**0.749**	0.755	0.745
stringfret-frame	**0.941**	0.984	0.988

Πίνακας 5.10: *Αποτελέσματα μοντέλου OAF Org*

Μετρική	F1	Precission	Recall
stringfret	**0.869**	0.821	0.821
stringfret-with-offsets	**0.014**	0.015	0.013
stringfret-frame	**0.171**	0.998	0.105

Πίνακας 5.11: *Αποτελέσματα μοντέλου OAF Org*

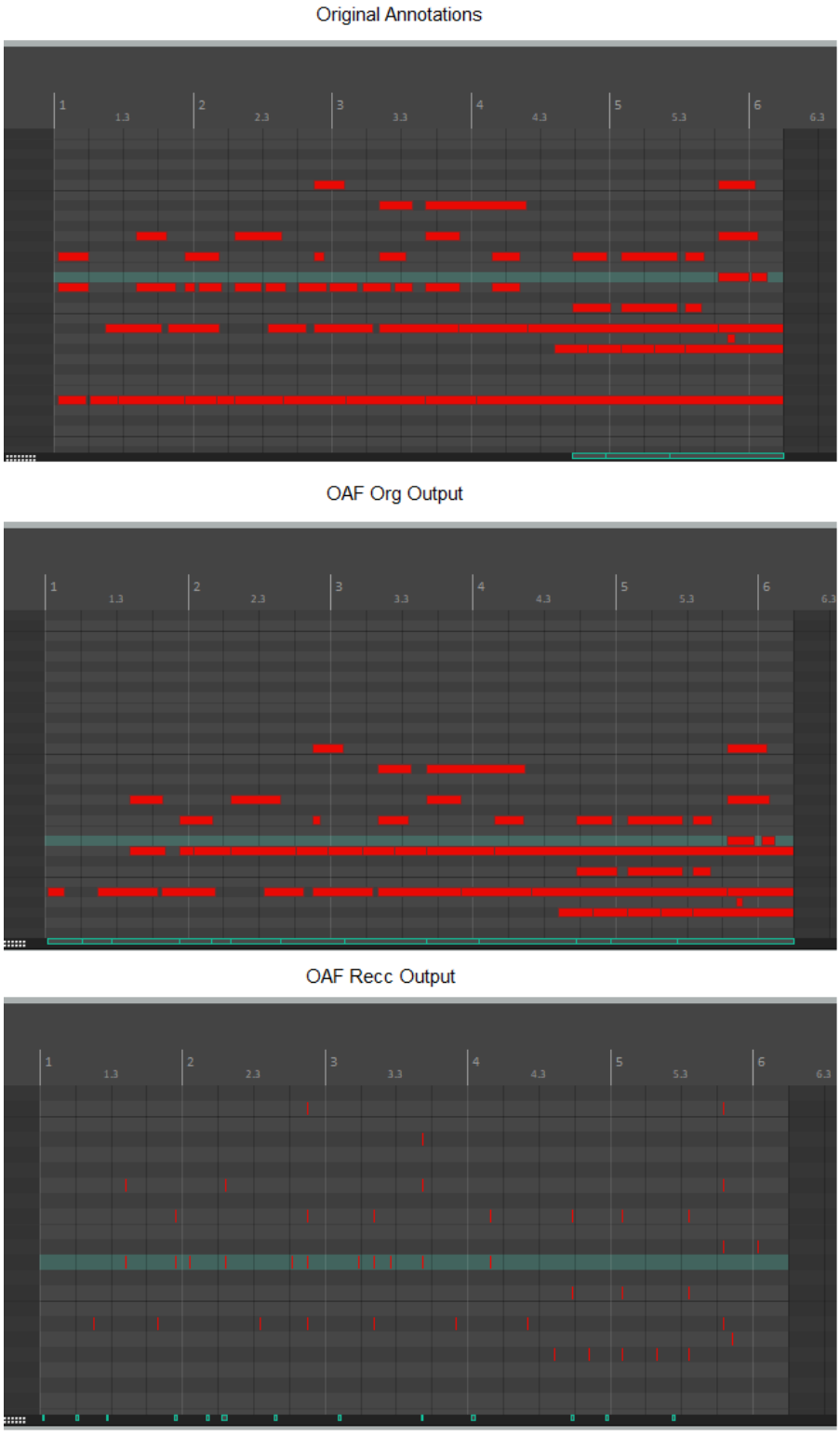
ίδευσης. Αντιθέτως, οι μεγαλύτερες ακολουθίες οδήγησαν σε πιο χρονοβόρα εκπαίδευση του δικτύου, επομένως η επιλογή των ακολουθιών 10 δευτερολέπτων ήταν η σωστή.

5.3 Παράδειγμα μεταγραφής

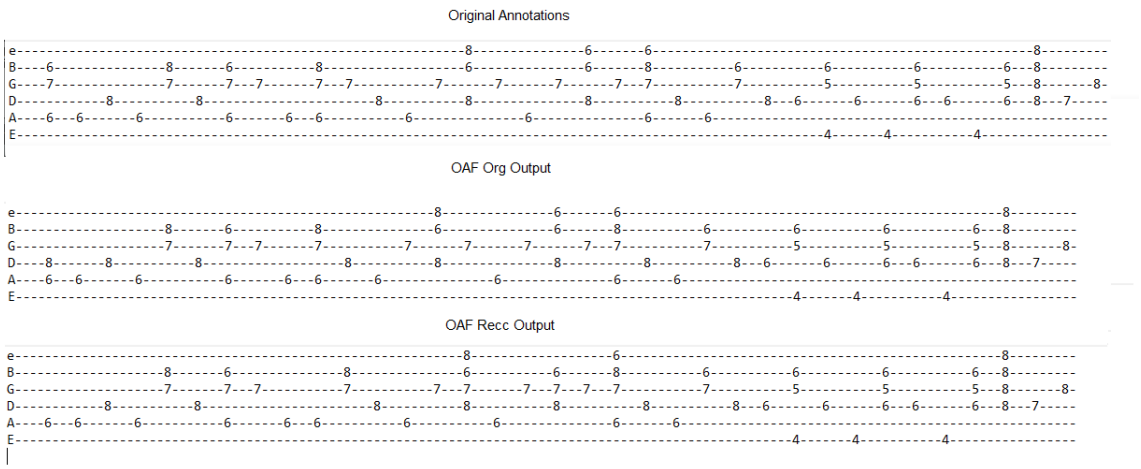
Στη συνέχεια θα παρουσιάσουμε τις δυνατότητες μεταγραφής των πιο αποδοτικών μοντέλων. Αρχικά, το καλύτερο μοντέλο που έχουμε στη διάθεση μας, είναι το OAF Org. Στην εικόνα 5.1 φαίνεται η μεταγραφή ενός μουσικού κομματιού σε μορφή Piano Roll, δηλαδή νότες MIDI, με την πάνω μεταγραφή να προκύπτει από τις δοσμένες σημειώσεις, ενώ η κάτω μεταγραφή προκύπτει από την έξοδο του δικτύου. Παρατηρούμε πως υπάρχει αρκετά μεγάλη ακρίβεια μεταξύ των δύο αυτών μεταγραφών, επιβεβαιώνοντας τα πολύ υψηλά F1-scores που προέκυψαν στα προηγούμενα πειράματα. Η τρίτη μεταγραφεί προκύπτει από την έξοδο του δικτύου OAF Recc. Παρατηρούμε ότι τα onsets των νοτών συμπίπτουν με τα δεδομένα, αλλά λόγω της αδυναμίας στην ανίχνευση των offsets, οι νότες ουσιαστικά δεν έχουν διάρκεια.

Η αναπαράσταση σε μορφή Piano Roll έγινε μέσω της εφαρμογής Digital Audio Workstation (DAW) Reaper. Χρειάστηκε επίσης μια εφαρμογή που να αποτυπώνει τη μεταγραφή σε μορφή ταμπλατούρας. Προς αυτό, αναπτύχθηκε μια συνάρτηση που παράγει ταμπλατούρες σε μορφή ASCII με βάση τους κανόνες που αναφέρθηκαν στην ενότητα 5.2.2. Συγκεκριμένα, θεωρούμε ότι δύο νότες ανήκουν σε μία συγχορδία, εάν η απόσταση μεταξύ τους είναι μικρότερη των 50 ms. Με βάση αυτό, αποτυπώνουμε σε μορφή ASCII τις νότες εισόδου.

Στο σχήμα 5.2 φαίνεται η μεταγραφή του παραπάνω κομματιού σε μορφή ταμπλατούρας. Όπως και πριν, η πάνω μεταγραφή προκύπτει από τις σημειώσεις ενώ η δεύτερη μεταγραφή προκύπτει από την έξοδο του νευρωνικού. Όπως και πριν, παρατηρούμε αρκετά καλή απόδοση, με τις ταμπλατούρες που προκύπτουν να είναι σχεδόν ίδιες. Στην τρίτη μεταγραφή που προκύπτει από την έξοδο του δικτύου OAF Recc, παρατηρούμε πως ενώ δεν είχαμε την επιθυμητή μεταγραφή σε μορφή Piano Roll που θα θέλαμε, η μεταγραφή σε απλοποιημένη ταμπλατούρα είναι πολύ πιστή με την αρχική είσοδο. Αυτό ήταν αναμενόμενο, καθώς σε αυτή την αναπαράσταση χάνεται πληροφορία για τη διάρκεια των νοτών.



Σχήμα 5.1: *Παράδειγμα μεταγραφής σε αναπαράσταση Piano Roll*



Σχήμα 5.2: *Παράδειγμα μεταγραφής σε αναπαράσταση ταμπλατούρας*

5.4 Σύγκριση με υπάρχοντα αποτελέσματα

Όπως αναφέρθηκε, τα μοντέλα μας στηρίζονται σε μεγάλο βαθμό στο έργο που παρουσιάζεται στο [1]. Σε αυτή την ενότητα θα συγκρίνουμε τα αποτελέσματά μας με αυτά που παρουσιάζονται στη δημοσίευση αυτή.

Μοντέλο	Note F1	Frame F1	Note with offset
Onsets And Frames	0.783	0.823	0.502
OAF Org	0.956	0.935	0.767
OAF Conv	0.297	0.003	0.048
OAF Recc	0.929	0.186	0.186
OAF Mixed	0.934	0.010	0.190
OAF Pre Trained	0.657	0.530	0.687

Πίνακας 5.12: *Σύγκριση απόδοσης νέων μοντέλων σε σχέση με ήδη υπάρχων.*

Παρατηρούμε ότι η απόδοση του μοντέλου OAF Org σε όλες τις μετρικές είναι πολύ καλύτερη από το μοντέλο που παρουσιάζεται στην [1], με τα υπόλοιπα μοντέλα να έχουν εξίσου καλή απόδοση στη μετρική που αφορά τα onsets, με εξαίρεση το OAF Conv. Αυτό μας δείχνει σε πρώτο στάδιο, πως η αναπαράσταση που χρησιμοποιήθηκε ήταν κατάλληλη, αφού επιτύχαμε καλύτερη απόδοση από το αρχικό μοντέλο. Παρόλα αυτά, πρέπει να λάβουμε υπόψη πως το Dataset που χρησιμοποιήσαμε ήταν σημαντικά μικρότερο από το MAESTRO Dataset ([15]), οπότε οι μετρικές αυτές είναι πολύ πιο περιορισμένες. Συνεπώς, παρόλο που η απόδοση των δικτύων είναι πολύ καλή, δε μπορούμε να παρουσιάσουμε με σιγουριά πως το μοντέλο θα είχε καλύτερη απόδοση σε ένα εξίσου μεγάλο Dataset.

Μέρος III

Επίλογος

Κεφάλαιο 6

Επίλογος

Στο κεφάλαιο αυτό γίνεται σύνοψη της διπλωματικής εργασίας και παρουσιάζονται τα συμπεράσματα που προκύπτουν από τα πειράματα που διεξήχθησαν. Τέλος, παρουσιάζονται κάποιες μελλοντικές επεκτάσεις που μπορούν να εφαρμοστούν.

6.1 Σύνοψη

Στη διπλωματική αυτή μελετήθηκε το πρόβλημα της μεταγραφής μουσικής κιθάρας σε συμβολική αναπαράσταση και συγκεκριμένα σε αναπαράσταση ταμπλατούρας. Για το σκοπό αυτό χρησιμοποιήθηκε μια νέα αναπαράσταση των νοτών, παρόμοιας λογικής με την αναπαράσταση MIDI, ώστε πλέον για την μεταγραφή της ταμπλατούρας να απαιτείται μόνο το onset και ο αριθμός της νότας, με τη χορδή να είναι κωδικοποιημένη μέσα στον αριθμό αυτό. Στα πειράματα έγινε προσπάθεια να καταγραφεί και η πληροφορία της διάρκειας των νοτών, παρόλο που αυτό δεν απαιτείται για το απλοποιημένο είδος ταμπλατούρας.

Η διαδικασία της αντιμετώπισης του προβλήματος αυτού χωρίστηκε σε δύο τμήματα. Στο πρώτο τμήμα, έγινε η επεξεργασία των δεδομένων του Dataset, και έγινε η μετατροπή στην επιθυμητή αναπαράσταση τόσο της εισόδου, όσο και των ετικετών. Έγιναν επίσης οι απαραίτητες μετατροπές ώστε να αποκτήσουμε περισσότερα δεδομένα με τη μορφή του Data Augmentation, καθώς και η προετοιμασία των ηχητικών κομματιών εισόδου ώστε να είναι ευκολότερη η διαχείρισή τους.

Στο δεύτερο βήμα, έχοντας ως βάση τη δουλειά που έγινε στο [1], αναπτύχθηκαν μοντέλα που βασίζονται κατά κύριο λόγο στην αρχιτεκτονική του μοντέλου Onsets and Frames. Στο πρώτο είδος μοντέλου που χρησιμοποιήθηκε, αποσκοπούσε στην απευθείας μεταγραφή σε αναπαράσταση κιθάρας, ενώ στο δεύτερο είδος μοντέλου, εκτελούμε πρώτα μεταγραφή σε αναπαράσταση MIDI (δηλαδή ταμπλατούρα) και στη συνέχεια προσπαθούμε να μετατρέψουμε την αναπαράσταση αυτή σε μορφή ταμπλατούρας. Χρησιμοποιήθηκαν διάφορες τεχνικές για την αποφυγή του overfitting, όπως το Early Stopping και το Data Augmentation και ελέγχθηκε η μεταφορά μάθησης από εκπαιδευμένο μοντέλο [1].

Τέλος, χρησιμοποιήθηκαν οι κατάλληλες μετρικές για την αξιολόγηση της απόδοσης των μοντέλων που αναπτύχθηκαν, με βάση τη βιβλιοθήκη mir_eval ([13], [14]) . Τα αποτελέσματα συγκρίθηκαν με υπάρχουσα μοντέλα που επιτελούν παρόμοιο έργο.

6.2 Συμπεράσματα

Αρχικά, θα σχολιάσουμε την απόδοση των επιμέρους μοντέλων στη μεταγραφή μουσικής για κιθάρα.

Το μοντέλο OAF Org είναι με διαφορά το καλύτερο σε απόδοση μοντέλο που αναπτύχθηκε, με πολύ καλή αναγνώριση τόσο του onset των νοτών, όσο και της διάρκειας. Αυτό μπορεί να δικαιολογηθεί λόγου του γεγονότος ότι βασίζεται σε υπάρχουσα δοκιμασμένη αρχιτεκτονική, αλλά και στο ότι είναι το απλούστερο από τα μοντέλα. Το γεγονός ότι ένα μοντέλο λιγότερες παραμέτρους, συχνά οδηγεί σε καλύτερη απόδοση.

Το μοντέλο OAF Conv είχε τη χειρότερη απόδοση από όλα τα μοντέλα, το οποίο οδηγεί στο συμπέρασμα πως για τη μετατροπή από την αναπαράσταση παρτιτούρας σε αναπαράσταση ταμπλατούρας, απαιτείται η χρήση των αναδρομικών δικτύων. Αυτό είναι λογικό, καθώς τα συμφραζόμενα έχουν μεγάλη σημασία στη μουσική, οπότε είναι λογικό πως η έλλειψη ανίχνευσης αυτών να οδηγεί σε χειρότερη απόδοση.

Τα μοντέλα OAF Recc και OAF Mixed πετυχαίνουν παρόμοια απόδοση μεταξύ τους. Παρόλο που πετυχαίνουν αντάξια απόδοση στην αναγνώριση των onsets των νοτών με το OAF Org, αποτυγχάνουν να αναγνωρίσουν την πληροφορία σε σχέση με τη διάρκεια των νοτών. Αυτό μπορεί να οφείλεται στην πιο περίπλοκη αρχιτεκτονική τους, καθώς και στην αυξημένη πολυπλοκότητα του προβλήματος. Η απευθείας μεταγραφή σε ταμπλατούρα είναι πιο απλή από τη μεταγραφή σε δύο στάδια.

Τα μοντέλα OAF Pre Trained εμφάνισε καλή απόδοση και στις δύο μετρικές, γεγονός που δείχνει πως είναι πιο ευσταθές από τα OAF Recc και OAF Mixed, αφού δεν χάνει εντελώς την πληροφορία για τη διάρκεια. Παρόλα αυτά, η σχετικά κακή απόδοση του στο πιο σημαντικό πρόβλημα, το οποίο είναι η αναγνώριση των onsets, μας οδηγεί στο να επιλέξουμε κάποιο από τα άλλα μοντέλα ως πιο επιθυμητό.

Αναφορικά με το Data Augmentation, το μεγαλύτερο πλήθος δεδομένων βοήθησε σε μικρό βαθμό στην καταπολέμηση του φαινομένου overfitting, καθώς αυτό εμφανίστηκε στο ίδιο πλήθος εποχών, αλλά για πολύ περισσότερα δεδομένα σε κάθε εποχή. Τέλος, το μήκος των δειγμάτων εισόδου παρατηρήθηκε πως δεν επιφέρει κάποια διαφορά στην απόδοση των δικτύων, ενώ αυξάνει το χρόνο που χρειάζονται για να εκπαιδευτούν, οπότε η αρχική επιλογή των 10 δευτερολέπτων ήταν κατάλληλη.

Συνολικά, μπορούμε να συμπεράνουμε πως τα νέα μοντέλα επιτελούν το σκοπό της εργασίας μας σε ικανοποιητικό βαθμό. Η αναπαράσταση που χρησιμοποιήθηκε ήταν κατάλληλη και αυτό φαίνεται από την πολύ καλή απόδοση της υπάρχουσας αρχιτεκτονικής του μοντέλου OAF Org, η οποία προσαρμόστηκε πλήρως στο πρόβλημά μας.

Για το συγκεκριμένο σκοπό της μεταγραφής σε ταμπλατούρα, μπορεί να αγνοηθεί η παράμετρος της διάρκειας στις νότες, στην περίπτωση που υπάρχει η απλουστευμένη αναπαράσταση τύπου ASCII. Σε αυτή την περίπτωση, τα μοντέλα OAF Org, OAF Recc και OAF Mixed έχουν εξίσου καλή απόδοση στη μεταγραφή, αφού λαμβάνεται ουσιαστικά υπόψη μόνο η κεφαλή onset. Επομένως, μπορούμε να θεωρήσουμε πως για το συγκεκριμένο αυτό πρόβλημα, οι 3 αυτές αρχιτεκτονικές αποτελούν μια καλή λύση.

Παρόλο που με την ανάπτυξη των νέων μοντέλων δεν καταφέραμε να πετύχουμε καλύτερη απόδοση από το OAF Org, μπορούμε να καταλήξουμε στο συμπέρασμα πως με τη χρήση του

νέου Dataset GuitarSet και της αναπαράστασης σε stringfret, επιλύουμε σε ικανοποιητικό βαθμό το πρόβλημα μεταγραφής μουσικής κιθάρας, τόσο σε αναπαράσταση ταμπλατούρας, όσο και σε αναπαράσταση MIDI.

6.3 Μελλοντικές Επεκτάσεις

Οι μελλοντικές επεκτάσεις θα μπορούσαν να αφορούν διάφορες κατευθύνσεις. Μια πρώτη κατεύθυνση αφορά την αρχιτεκτονική των δικτύων και είναι η αναγνώριση των offsets των νοτών με ξεχωριστή κεφαλή του δικτύου. Όπως αναφέρθηκε, τα onsets των νοτών είναι ένα από τα πιο βασικά χαρακτηριστικά τους, αλλά σημαντικό ρόλο παίζουν και τα offsets, τα οποία ορίζουν τη διάρκεια των νοτών. Πιθανών με τη βοήθεια της πληροφορίας των offsets να βοηθηθεί η απόδοση των δικτύων που αδυνατούν να αποκτήσουν πληροφορία για τη διάρκεια των νοτών.

Ένα άλλο μειονέκτημα των υλοποιήσεών μας είναι πως απαιτούν ολόκληρο το ηχογραφημένο κομμάτι ως είσοδο, αφού απαιτούν συγκεκριμένο μέγεθος εισόδου. Η τροποποίηση των αρχιτεκτονικών θα μπορούσε να γίνει ώστε αυτές να μπορούν να κάνουν τη μεταγραφή σε πραγματικό χρόνο, το οποίο θα βοηθούσε σε διάφορους τομείς, όπως η εκμάθηση της μουσικής.

Μια άλλη μελλοντική επέκταση που χρειάζεται είναι ένα μεγαλύτερο dataset. Όπως αναφέρθηκε, το GuitarSet είναι πολύ μικρό σε σχέση με datasets τα οποία είναι διαθέσιμα για μουσική πιάνου (MAPS, MAESTRO), το οποίο περιορίζει σαφώς τη δυνατότητα εκπαίδευσης. Ένα μεγαλύτερο dataset, τόσο σε πλήθος, όσο και σε διάρκεια κομματιών θα βοηθούσε σημαντικά στην εκπαίδευση ισχυρότερων δικτύων.

Μια άλλη κατεύθυνση στην οποία μπορούμε να κινηθούμε είναι η μεταφορά χροιάς των οργάνων. Στο [16] παρουσιάζεται η δυνατότητα μεταφοράς χροιάς ενός οργάνου σε ένα άλλο. Παρόλο που τα όργανα στα οποία η μετατροπή αυτή είναι δυνατή αυτή τη στιγμή, είναι ίσως δυνατό να μετατρέψουμε τον ήχο της κιθάρας σε ήχο πιάνου, και με βάση αυτό να κάνουμε μεταγραφή της μουσικής (πλέον σε μορφή πιάνου) με τη χρήση έτοιμων και ισχυρών μοντέλων. Παρόλα αυτά, θα χρειαστεί ξανά μετατροπή σε αναπαράσταση ταμπλατούρας, αυξάνοντας την πολυπλοκότητα περαιτέρω.

Μια τελευταία επέκταση θα ήταν η χρήση διαφορετικού τύπου εισόδου. Προς το παρών, χρησιμοποιούμε ως είσοδο ολόκληρο το φασματογράφημα του ήχου εισόδου. Παρόλα αυτά, είναι δυνατό μέσω τεχνικών επεξεργασίας σήματος (F0 - detection) να αναγνωρίσουμε τις νότες του κομματιού και να περάσουμε αυτές ως είσοδο στο δίκτυο. Πιθανόν μια τέτοια αναπαράσταση να διευκολύνει την εκπαίδευση, αφού τα πρότυπα θα είναι πιο ξεκάθαρα, και η είσοδος μικρότερη.

Παραρτήματα

Παράρτημα Α΄

Τεχνολογίες που Χρησιμοποιήθηκαν

Για την υλοποίηση της διπλωματικής χρησιμοποιήθηκε κατά κύριο λόγο η γλώσσα προγραμματισμού Python. Για την ανάπτυξη και εκπαίδευση των δικτύων χρησιμοποιήθηκε το framework pytorch, καθώς και άλλες βιβλιοθήκες, όπως το scikit-learn που παρέχει μεθόδους μηχανικής μάθησης και το matplotlib για την παρουσίαση των δεδομένων σε γραφικές παραστάσεις. Επιπλέον, χρησιμοποιήθηκε η βιβλιοθήκη jams για το διάβασμα των αρχείων εισόδου από το dataset. Η εκπαίδευση των δικτύων έγινε στην πλατφόρμα Google Colab, με σκοπό την εκμετάλλευσή των πόρων που παρέχονται.

Χρησιμοποιήθηκαν επιπλέον αρκετές βιβλιοθήκες επεξεργασίας σήματος. Αρχικά, με το λογισμικό ffmpeg [11] έγιναν οι απαραίτητες μετατροπές στα ηχητικά δεδομένα εισόδου, όπως η αλλαγή συχνότητας δειγματοληψίας. Μέσω της βιβλιοθήκης librosa ([17]) έγιναν διάφοροι μετασχηματισμοί και επεξεργασία σήματος.. Επίσης χρησιμοποιήθηκαν οι βιβλιοθήκες mido και soundfile για την ανάγνωση και εγγραφή αρχείων MIDI.

Το Dataset ([5]) που χρησιμοποιήθηκε για την εκπαίδευση και την αξιολόγηση των μοντέλων είναι το GuitarSet. Το dataset αυτό αποτελείται από κομμάτια κιθάρας με τις αντίστοιχες σημειώσεις για κάθε νότα που παίζεται. Όπως αναφέρεται στην αντίστοιχη δημοσίευση [4], παρόλο που η κιθάρα και το πιάνο είναι εξίσου δημοφιλή, τα dataset που αφορούν επιβλεπόμενη μάθηση μουσικής αφορούν αποκλειστικά το πιάνο (MAPS, MAESTRO). Επομένως, το GuitarSet είναι ιδιαίτερα χρήσιμο για τη δημιουργία μοντέλων που δεν αφορούν αποκλειστικά το πιάνο.

Το GuitarSet αποτελείται από 360 κομμάτια μήκους σχεδόν 30 δευτερολέπτων. Τα κομμάτια αυτά δημιουργήθηκαν από έξι διαφορετικούς παίχτες, δεδομένων 30 διαφορετικών οδηγών (lead sheets) που αποτελούνται από διάφορα είδη μουσικής (Rock, Singer-Songwriter, Bossa Nova, Jazz, Funk), ακολουθίες και ρυθμούς. Κάθε κομμάτι έχει δύο παραλλαγές, μία είναι η συνοδεία, η οποία περιέχει κυρίως συγχορδίες, και η άλλη είναι το solo, που περιέχει κυρίως μελωδίες.

Οι ηχογραφήσεις έγιναν μέσω μαγνητών και μικρόφωνου, με τους μαγνήτες να προσδίδουν την καλύτερη πιστότητα. Χρησιμοποιήθηκε ειδικός εξαφωνικός μαγνήτης που καταγράφει το σήμα της κάθε χορδής ξεχωριστά, το οποίο χρησιμοποιείται στο annotation των δεδομένων. Το annotation δίνεται σε μορφή αρχείου JAMS, τα οποία περιέχουν πληροφορίες για το τονικό ύψος και τη χροιά της νότας που παίζεται, μεταξύ άλλων.

Η κύρια κατηγορία annotation που αφορά την εφαρμογή είναι τα midi note annotiations. Σε αυτά καταγράφεται για κάθε χορδή ξεχωριστά πληροφορία για τις νότες που

παίζονται, σε αναπαράσταση midi, καθώς και τα onsets και offsets της νότας, τα οποία είναι τα βασικά χαρακτηριστικά που χρειαζόμαστε για τη μάθηση.

Βιβλιογραφία

[1] Curtis Hawthorne, Erich Elsen, Jialin Song, Adam Roberts, Ian Simon, Colin Raffel, Jesse Engel, Sageev Oore και Douglas Eck. *Onsets and Frames: Dual-Objective Piano Transcription*. *ArXiv*, 2017.

[2] Rainer Kelz, Matthias Dorfer, Filip Korzeniowski, Sebastian Böck, Andreas Arzt και Gerhard Widmer. *On the Potential of Simple Framewise Approaches to Piano Transcription*. *ISMIR*, 2016.

[3] Siddharth Sigtia, Emmanouil Benetos και Simon Dixon. *An End-to-End Neural Network for Polyphonic Piano Music Transcription*. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24:927–939, 2016.

[4] Qingyang Xi, Rachel M. Bittner, Johan Pauwels, Xuzhou Ye και Juan Pablo Bello. *GuitarSet: A Dataset for Guitar Transcription*. *ISMIR*, 2018.

[5] *GuitarSet*. https://guitarset.weebly.com/. Ημερομηνία πρόσβασης: 4-2-2022.

[6] Gordon C. Bruner II. *Music, Mood, and Marketing*. *Journal of Marketing*, 54(4):94–104, 1990.

[7] Patricia J. Flowers. *Attention to Elements of Music and Effect of Instruction in Vocabulary on Written Descriptions of Music by Children and Undergraduates*. *Psychology of Music*, 12(1):17–24, 1984.

[8] Muhammad Huzaifah Md Shahrin. *Comparison of Time-Frequency Representations for Environmental Sound Classification using Convolutional Neural Networks*. *ArXiv*, 2017.

[9] S. S. Stevens. *A Scale for the Measurement of a Psychological Magnitude: Loudness*. *Psychological Review*, 43(5):405–416, 1936.

[10] Anidhya Athaiya Siddharth Sharma, Simone Sharma. *Activation Functions in Neural Networks*. *International Journal of Engineering Applied Sciences and Technology*, 4(12):310–316, 2020.

[11] *ffmpeg*. https://ffmpeg.org/. Ημερομηνία πρόσβασης: 5-2-2022.

[12] *PyTorch Implementation of Onsets and Frames*. https://github.com/jongwook/onsets-and-frames. Ημερομηνία πρόσβασης: 5-2-2022.

[13] Colin Raffel, Brian Mcfee, Eric Humphrey, Justin Salamon, Oriol Nieto, Dawen Liang και Daniel Ellis. *mir_eval: A Transparent Implementation of Common MIR Metrics*. 2014.

[14] *mir_eval*. https://github.com/craffel/mir_eval. Ημερομηνία πρόσβασης: 5-2-2022.

[15] Curtis Hawthorne, Andriy Stasyuk, Adam Roberts, Ian Simon, Cheng Zhi Anna Huang, Sander Dieleman, Erich Elsen, Jesse Engel και Douglas Eck. *Enabling Factorized Piano Music Modeling and Generation with the MAESTRO Dataset*. *International Conference on Learning Representations*, 2019.

[16] Michelle Carney, Chong Li, Edwin Toh, Ping Yu και Jesse Engel. *Tone Transfer: In-Browser Interactive Neural Audio Synthesis*. 2021.

[17] Brian McFee, Colin Raffel, Dawen Liang, Daniel Ellis, Matt Mcvicar, Eric Battenberg και Oriol Nieto. *librosa: Audio and Music Signal Analysis in Python*. σελίδες 18–24, 2015.

[18] Simon Haykin. *Νευρωνικά Δίκτυα και Μηχανική Μάθηση*. Παπασωτηρίου, Αθήνα, 3η έκδοση, 2010.

[19] Jarmo Lähdevaara. *The Science of Electric Guitars And Guitar Electronics*. 2012.

[20] *The Physics of Everyday Stuff*. http://http://www.bsharp.org/physics/guitar. Ημερομηνία πρόσβασης: 2-2-2022.

[21] H.J. Pain. *Φυσική των ταλαντώσεων και των κυμάτων*. Συμμετρία, Αθήνα, 3η έκδοση, 1997.

[22] Σ. Κουτουπης. *Εξέταση Τεχνικών Αυτόματης Μεταγραφής Ακουστικού Σήματος Κιθάρας σε Συμβολική Αναπαράσταση με Χρήση Μεθόδων Ψηφιακής Επεξεργασίας Σήματος και Μηχανικής Μάθησης*. Διπλωματική εργασία, Εθνικό Μετσόβιο Πολυτεχνείο, 2021.

[23] Elias Mistler. *Generating Guitar Tablatures with Neural Networks*. Διπλωματική εργασία, University of Edinburgh, 2017.

[24] Elias Mistler. *Generating Guitar Tablatures with Neural Networks*. Διπλωματική εργασία, University of Edinburgh, 2017.

[25] Thomas M. Maaiveld. *Automatic Tablature Estimation with Convolutional Neural Networks: Approaches and Limitations*. Διπλωματική εργασία, Vrije Universiteit Amsterdam, 2021.

Συντομογραφίες - Αρκτικόλεξα - Ακρωνύμια

βλπ	βλέπε
κ.λπ.	και λοιπά
κ.ο.κ	και ούτω καθεξής
MIR	Music Information Retrieval
MIDI	Musical Instrument Digital Interface
MLP	Multi Layer Perceptron
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
LSTM	Long Term Short Term Memory
DAW	Digital Audio Workstation

Απόδοση ξενόγλωσσων όρων

Απόδοση	Ξενόγλωσσος όρος
Ανάκτηση Μουσικής Πληροφορίας	Music Information Retrieval
Συνολο Δεδομενων	Dataset
μεταγραφή	transcription
παρτιτουρα	sheet music
ταμπλατούρα	tablature
έναρξη	onset
λήξη	offset
διάρκεια	duration
ηχόχρωμα	timbre
φασματογράφημα	spectrogram
οπλισμός	key signature
συχνότητα δειγματοληψίας	sample rate
κάδοι συχνοτήτων	frequency bins
δίκτυα πρόσθιας τροφοδότησης	feedforward networks
πόλωση	bias
βαθμωτή κατάβαση	gradient descent
ταξινόμηση	classification
κλάση	class
ταμπέλα	label
οδηγός	lead sheet
επαναδειγματοληψια	resampling
χαρακτηριστικό	feature
ρυθμός μάθησης	learning rate
δείγμα	sample