ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Αυτόματη εναρμόνιση μελωδίας με νευρωνικά δίκτυα

Διπλωματική εργασία

Δημήτριος Ν. Συρράφος

Επιβλέπων : Γεώργιος Στάμου
Αν. Καθηγητής ΕΜΠ

Αθήνα, Μάιος 2020

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Αυτόματη εναρμόνιση μελωδίας με νευρωνικά δίκτυα

Διπλωματική εργασία

Δημήτριος Ν. Συρράφος

Επιβλέπων : Γεώργιος Στάμου
Αν. Καθηγητής ΕΜΠ

Εγκρίθηκε από τριμελή εξεταστική επιτροπή την 21η Μαΐου 2020

.....	.....	.....
Γ. Στάμου	Σ. Κόλλιας	Α.-Γ. Σταφυλοπάτης
Αν. Καθηγητής ΕΜΠ	Καθηγητής ΕΜΠ	Καθηγητής ΕΜΠ

Αθήνα, Μάιος 2020

.....
Συρράφος Ν. Δημήτριος
Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Copyright © Συρράφος Ν. Δημήτριος, 2020
Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

Περιεχόμενα

Περίληψη.....7
Abstract.....8
Ευχαριστίες.....9
1. Εισαγωγή.....10
1.1. Γενικά.....10
1.2. Εναρμόνιση μελωδίας με νευρωνικά δίκτυα.....11
1.3. Στόχος της διπλωματικής εργασίας.....11
1.3. Δομή διπλωματικής εργασίας.....12
2. Θεωρητικό υπόβαθρο.....13
2.1. Βασικές έννοιες της μουσικής θεωρίας.....13
2.1.1. Νότες και διαστήματα.....13
2.1.2. Συγχορδίες.....15
2.1.3. Κλίμακες και Τρόποι.....16
2.1.4. Δευτερεύουσες συγχορδίες και δανεισμοί συγχορδιών.....17
2.1.5. Μέτρο και Χρονική αξία.....17
2.2. Αναδρομικά νευρωνικά Δίκτυα.....19
2.2.1. Αρχιτεκτονική και Εκπαίδευση αναδρομικών νευρωνικών δικτύων.....20
2.2.2. Δίκτυο Long Short-Term Memory (LSTM).....22
2.2.3 Sequence to Sequence αρχιτεκτονική.....26
2.3. Συνελικτικά Νευρωνικά Δίκτυα.....31
2.3.1. Γιατί εναρμόνιση μελωδίας με συνελικτικά νευρωνικά δίκτυα.....31
2.3.2. Επίπεδα συνελικτικού νευρωνικού δικτύου.....32
2.3.3. Διασταλμένη Συνέλιξη.....35
2.3.4. Υπολειμματικά δίκτυα.....36
2.3.5. Διαχωριζόμενη συνέλιξη.....38
2.4. Μέθοδοι δειγματοληψίας.....40
3. Αναπαράσταση δεδομένων.....43
3.1. Προέλευση dataset.....43
3.2. Στόχος προεπεξεργασίας δεδομένων.....43
3.3. Προεπεξεργασία δεδομένων.....45
3.4. Πόλωση dataset.....56
4. Εναρμόνιση μελωδίας με αναδρομικά νευρωνικά δίκτυα (RNN).....58
4.1. Αρχιτεκτονική δικτύου.....58
4.1.1. Κωδικοποιητής – Αποκωδικοποιητής – απλά LSTM.....58
4.1.2. Κωδικοποιητής – Αποκωδικοποιητής – αμφίδρομα LSTM.....59
4.1.3. Κωδικοποιητής – Αποκωδικοποιητής – 2 επίπεδα απλών LSTM.....60
4.1.4. Κωδικοποιητής – Αποκωδικοποιητής – απλό LSTM – μηχανισμός προσοχής.....61
4.2. Επιλογή υπερπαραμέτρων.....62
4.2.1. Διαχωρισμός dataset.....62
4.2.2. Συναρτήσεις ενεργοποίησης.....63
4.2.3. Συναρτήσεις κόστους.....63
4.2.4. Βελτιστοποιήτες.....65
4.3. Ανάλυση αποτελεσμάτων.....65
4.4. Αξιολόγηση.....73
4.4.1. Μετρικές.....73
4.4.2. Επίδραση πόλωσης δεδομένων.....75

4.4.3. Επίδραση μεγέθους κελιού....75
4.4.4. Επίδραση LSTM διπλής κατεύθυνσης....79
4.4.5. Επίδραση μηχανισμού προσοχής....80
4.4.6. Επίδραση επιλογής μεθόδου δειγματοληψίας....83
5. Εναρμόνιση μελωδίας με συνελικτικά νευρωνικά δίκτυα (CNN)....87
5.1. Σχετικά με το Coconet....87
5.2. Αρχιτεκτονική νευρωνικού δικτύου....88
5.3. Εκπαίδευση δικτύου....90
5.4. Σύγκριση με αναδρομικό μοντέλο....90
6. Συμπεράσματα....92
7. Υλοποίηση....93
Βιβλιογραφία....94

Περίληψη

Η αυτόματη σύνθεση μουσικής αποτελεί ένα από τα πιο κομβικά αλλά και δύσκολα έργα στον τομέα ανακατασκευής πληροφορίας. Η σύνθεση μουσικής όπως την αντιλαμβάνεται ένας μουσικός μπορεί να χωριστεί σε δύο επιμέρους διαδικασίες: την κατασκευή της μελωδίας του κομματιού και της αρμονίας του. Οι διαδικασίες αυτές είναι αλληλένδετες καθώς η μελωδία και η αρμονία πρέπει να παράγουν μαζί ένα αισθητικά ικανοποιητικό αποτέλεσμα. Μια συνήθης πρακτική είναι η κατασκευή της μελωδίας σε πρώτη φάση και έπειτα η εύρεση κατάλληλων συγχορδιών που να τη συνοδεύουν.
Αυτή η διαδικασία, το έργο δηλαδή της εναρμόνισης της μελωδίας με αυτόματο τρόπο αποτελεί το θέμα της συγκεκριμένης διπλωματικής εργασίας. Όπως με τα περισσότερα έργα του τομέα ανάκτησης μουσικής πληροφορίας, τα σύστημα αυτόματης εναρμόνισης μελωδίας τείνουν να αντικαθιστούν τα στάδια επεξεργασίας σήματος, εξαγωγής χαρακτηριστικών και στατιστικών μοντέλων με μεθόδους βαθειάς μηχανικής μάθησης. Η παρούσα εργασία δεν αποκλίνει από την επικρατούσα τάση να γίνεται αναπαράσταση της μουσικής με συμβολικό τρόπο με τη μορφή αρχείων XML.
Στο πειραματικό μέρος της εργασίας εκπαιδεύτηκαν διαφορετικά νευρωνικά δίκτυα με σκοπό την παραγωγή ακολουθιών συγχορδιών. Η ακολουθιακή φύση της μουσικής μας έκανε να στραφούμε σε αναδρομικά νευρωνικά δίκτυα που ειδικεύονται σε τέτοιου είδους εφαρμογές. Χρησιμοποιήσαμε αρκετές παραλλαγές του μοντέλου κωδικοποιητή αποκωδικοποιητή με απλά, αμφίδρομα και πολυεπίπεδα LSTM καθώς και το μοντέλο κωδικοποιητή αποκωδικοποιητή με μηχανισμό προσοχής.
Εκτός από την απόδοση των παραπάνω εξεταζόμενων μοντέλων, γίνεται και μια αξιολόγηση των τεχνικών διεξαγωγής προβλέψεων από τα μοντέλα αυτά.
Τέλος, επειδή ο χαρακτηρισμός μιας ακολουθίας συγχορδιών σαν σωστός η λανθασμένος δεν είναι ξεκάθαρος, για την συνολική αξιολόγηση των μοντέλων και των τεχνικών διεξαγωγής προβλέψεων χρησιμοποιηθήκαν κάποιες μετρικές συναρτήσεις που δείχνουν μια αίσθηση για το ποσό «ανθρώπινο» είναι το παραγόμενο αποτέλεσμα. Τα αποτελέσματα των εξεταζόμενων δικτύων συγκρίθηκαν με αληθινά δεδομένα καθώς και με τα αποτελέσματα που προήλθαν από ένα state of the art συνελικτικό νευρωνικό δίκτυο, το Coconet.

Λέξεις κλειδιά: τεχνητά νευρωνικά δίκτυα, μουσική σε συμβολική μορφή, βαθειά μηχανική μάθηση, αναδρομικά νευρωνικά δίκτυα, συνελικτικά νευρωνικά δίκτυα, αρχιτεκτονική κωδικοποιητή - αποκωδικοποιητή, μελωδία, αρμονία, συγχορδίες, κλίμακες και τρόποι

Abstract

Automatic music composition is one of the most crucial and difficult tasks in the field of information reconstruction. Music composition, as a musician perceives it, can be divided into two separate processes: the composition of the song's melody and its corresponding harmony. These processes are interrelated as melody and harmony must produce an aesthetically satisfying result. A common practice is to compose the melody in the first place and, consequently, find the appropriate chords to accompany it.

This process, namely the task of harmonizing the melody automatically, is the subject of this diploma thesis. As with most music information recovery projects, automatic melody harmonics tend to replace the stages of signal processing, feature extraction and statistical models with deep machine learning methods. The present work does not depart from the prevailing tendency to represent music in a symbolic way in the format of XML files.

In the experimental part of this diploma thesis, different neural networks were trained in order to produce concordant sequences. The sequential nature of music has made us turn to recursive neural networks that specialize in such applications. We used several variants of the encoder – decoder model, involving simple LSTM, bidirectional and multilayer decoder model as well as the decoder model with attention mechanism.

In addition to the performance of the models considered above, an evaluation of result sampling techniques from these models has been performed.

Finally, because the characterization of a chord sequence in a musical piece as correct or incorrect is unclear (and not entirely objective), we included in the evaluation process some metrics that give us a sense of how "human-like" the result is, as an overall evaluation of the models and prediction techniques. The results of the networks under consideration were compared with real data as well as with results from a state of the art convolutional neural network, Coconet.

Keywords: artificial neural networks, music in symbolic language, deep learning, recursive neural networks, convolutional neural networks, encoder - decoder architecture, melody, harmony, chords, scales and modes

Ευχαριστίες

Θα ήθελα, καταρχάς, να ευχαριστήσω τον καθηγητή κ. Στάμου για την ανάθεση και επίβλεψη αυτής της διπλωματικής εργασίας. Επίσης, τον Γρηγόρη - Έντμουντ Ντερβάκο για την ιδιαίτερα στενή και αποδοτική συνεργασία που είχαμε κατά την εκπόνηση της συγκεκριμένης εργασίας. Ακόμη, θα ήθελα να εκφράσω τις ευχαριστίες μου στους δασκάλους μου Δημήτρη Αντωνόπουλο, Στέλιο Κυριακόπουλο και Χρυσόστομο Καλογριδάκη για την συμβολή τους στη μέχρι τώρα πορεία μου. Τέλος, είμαι ειλικρινά ευγνώμων απέναντι στην οικογένειά μου, τους συμφοιτητές μου, με τους οποίους μοιράστηκα τα χρόνια της φοίτησής μου στο ΕΜΠ και τους φίλους μου, ιδιαίτερα τους Χρήστο Γιακουμή, Νίκο Λαμπέτη και Μαρία Πριμηκύρη.

1.Εισαγωγή

1.1. Γενικά

Στην ταινία “I, Robot” που έκανε πρεμιέρα το 2004, κάποια στιγμή ο Will Smith ρωτάει (ρητορικά) ένα ρομπότ: “Μπορεί ένα ρομπότ να γράψει μία συμφωνία; Μπορεί να μεταμορφώσει έναν άσπρο καμβά σε ένα όμορφο έργο τέχνης;”. Εύλογη απορία καθώς οι περισσότεροι άνθρωποι αντιμετωπίζουν τη δημιουργία μουσικών κομματιών σαν μία σύνθετη διαδικασία που εμπεριέχει τη σύμπραξη πολλαπλών παραγόντων και στοιχείων της ανθρώπινης προσωπικότητας. Γνώση, αισθητική, φαντασία και έμπνευση είναι μόνο μερικά από αυτά. Και αν η γνώση μπορεί να αναπαρασταθεί ψηφιακά και να τροφοδοτηθεί σε έναν υπολογιστικό σύστημα, έννοιες όπως η αισθητική, η φαντασία και η έμπνευση δύσκολα αποτυπώνονται με τη μορφή αλγορίθμων ή δεδομένων.

Παρ’ όλες τις δυσκολίες αυτές που εμφανίζονται εκ πρώτης όψεως, η σύνθεση μουσικής με συστήματα τεχνητής νοημοσύνης και βαθειάς μηχανικής μάθησης έχει προσελκύσει το ενδιαφέρον πολλών ερευνητών που δραστηριοποιούνται στο συγκεκριμένο τομέα. Και αυτό συμβαίνει γιατί παρά τη διαισθητική της φύση, η μουσική μπορεί να μοντελοποιηθεί και να προσεγγιστεί μαθηματικά σε μεγάλο βαθμό. Η θεώρηση της μουσικής ακόμα και από ένα μουσικό δεν είναι σε καμία περίπτωση αφηρημένη. Αντίθετα μοιάζει περισσότερο με ένα σύστημα κανόνων, σαφώς ορισμένων, που αποσκοπούν στο να παράγουν εύηχα αποτελέσματα.

Τα τελευταία χρόνια παρατηρείται ένα αυξημένο ενδιαφέρον από την πλευρά των ερευνητών που ασχολούνται με τον τομέα της βαθειάς μηχανικής μάθησης για εφαρμογές που σχετίζονται με τη μουσική. Πιο συγκεκριμένα, εντοπίζονται προσπάθειες ανάλυσης της μουσικής από εκπαιδευτικούς και ερευνητικούς φορείς. Αυτό συμβαίνει γιατί η δημιουργία ευφυών συστημάτων που αποσκοπούν στην ανάλυση της μουσικής και στην εξαγωγή πληροφορίας από μουσικά δεδομένα βρίσκουν πολλές εφαρμογές σε διαφόρους τομείς της βιομηχανίας. Τέτοια συστήματα είναι για παράδειγμα η κατηγοριοποίηση μουσικών έργων ανάλογα με το είδος μουσικής που ανήκει το καθένα, ο εντοπισμός κοινών στοιχείων διαφορετικών τραγουδιών, η αυτόματη σύνθεση μουσικής, η εύρεση χαρακτηριστικών ενός τραγουδιού όπως ο ρυθμός του, το κλειδί στο οποίο είναι γραμμένο ένα κομμάτι καθώς και τα μουσικά όργανα που ακούγονται σε ένα μουσικό κομμάτι. Τα συστήματα αυτά, με τη σειρά τους, μπορούν να βρουν χρήση σε εφαρμογές που θα εξυπηρετούν είτε μουσικούς, είτε ακόμα και το ευρύ κοινό. Παραδείγματα τέτοιων εφαρμογών είναι η αυτόματη συνοδεία ενός μουσικού από έναν υπολογιστή, η ρυθμοδότηση μιας ορχήστρας από έναν “έξυπνο μετρονόμο”, η αυτόματη ανάλυση της δομής και της αρμονίας ενός κομματιού για εκπαιδευτικούς λόγους, ακόμα και η δημιουργία αλγορίθμων που θα παράγουν συστάσεις τραγουδιών σε χρήστες μουσικών εφαρμογών ανάλογα με της μουσικές προτιμήσεις του χρήστη. Όλα τα παραπάνω είναι μόνο μερικές από τις εφαρμογές που μπορεί να έχουν τα συστήματα γνώσης και βαθειάς μηχανικής μάθησης στη μουσική βιομηχανία.

Επιστρέφοντας έτσι στην αρχική ερώτηση από την οποία εκκίνησε αυτή η εισαγωγή, η ανάλυση της μουσικής και η εξαγωγή πληροφοριών από μουσικά δεδομένα δεν είναι δυνατόν να αποτυπώσει πλήρως τη νοητική διαδικασία που ακολουθεί ένας συνθέτης όταν δημιουργεί ένα μουσικό έργο, ούτε να αντικατοπτρίσει όλους τους παράγοντες που οδήγησαν στην σύνθεσή του. Παρ’ όλα αυτά οι τεχνικές των συστημάτων γνώσης και βαθειάς μηχανικής μάθησης έχουν πετύχει σημαντικά αποτελέσματα πάνω σε tasks τα οποία έχουν μεγάλη χρησιμότητα σε πολλούς και διαφορετικούς τομείς αλλά έχουν κατορθώσει να γίνει καλύτερα κατανοητός ο τρόπος με τον οποίο οι άνθρωποι αλλά και ένα σύστημα τεχνητής νοημοσύνης αντιλαμβάνεται τη μουσική.

Η συγκεκριμένη διπλωματική εργασία έχει σαν θέμα την αυτόματη εναρμόνιση μουσικών μελωδιών με τη χρήση νευρωνικών δικτύων. Εξετάστηκαν διάφορα μοντέλα και αξιολογήθηκε η ικανότητα τους στο να παράγουν ακολουθίες συγχορδιών που να συνοδεύουν με αποδεκτό τρόπο τη μελωδία εισόδου. Η αξιολόγηση των μοντέλων έγινε με τη χρήση ειδικών μετρικών συναρτήσεων που υπολογίζουν τιμές οι οποίες χαρακτηρίζουν το κατά πόσο το παραγόμενο κομμάτι πλησιάζει τα ανθρώπινα μουσικά έργα. Τέλος έγινε σύγκριση των αποτελεσμάτων των μετρικών των αυτόματα παραγόμενων κομματιών με τα αποτελέσματα των μετρικών των δεδομένων εισόδου, τα οποία είναι αληθινά μουσικά κομμάτια.

1.2. Εναρμόνιση μελωδίας με νευρωνικά δίκτυα

Κατ' αρχήν, πριν γίνει οποιαδήποτε αναφορά στα τεχνικά θέματα της εργασίας, είναι χρήσιμο να γίνει σαφές τι σημαίνει ο όρος "εναρμόνιση" μίας μελωδίας. Ουσιαστικά πρόκειται για την διαδικασία επιλογής των κατάλληλων συγχορδιών για να συνοδεύσουν μια μελωδία. Ένας καλύτερος ορισμός, με την έννοια ότι ταιριάζει καλύτερα στο θέμα και το αντικείμενο της διπλωματικής αυτής εργασίας θα μπορούσε να είναι ο εξής: η διαδικασία αντιστοίχισης μιας τιμής της μελωδίας (νότα) σε μια τιμή της αρμονίας (συγχορδία) κάθε χρονική στιγμή. Ο συγκεκριμένος ορισμός αποτυπώνει καλύτερα τη λειτουργία που πρέπει να υλοποιήσει ένα μοντέλο για να φέρει σε πέρας το συγκεκριμένο έργο.

Για την εκπόνηση της συγκεκριμένης διπλωματικής εργασίας επιλέχθηκε η χρήση νευρωνικών δικτύων. Τα νευρωνικά δίκτυα είναι μια μορφή ευφυών συστημάτων που τα τελευταία χρόνια προσελκύει το ενδιαφέρον όλο και περισσότερων ερευνητών και κατορθώνουν να παράγουν σημαντικά αποτελέσματα σε ποικίλες εφαρμογές της τεχνητής νοημοσύνης. Η επιτυχία τους αυτή έγκειται σε διάφορους παράγοντες, σημαντικοί όμως είναι η ανάπτυξη περίπλοκων συστημάτων και αρχιτεκτονικών δικτύων που κατορθώνουν να μαθαίνουν και να εξάγουν χαρακτηριστικά από σύνθετα δεδομένα καθώς και η πληθώρα δεδομένων που είναι διαθέσιμη γα την εκπαίδευση αυτών των μοντέλων. Η συλλογή δεδομένων έχει αναπτυχθεί σε μεγάλο βαθμό τα τελευταία χρόνια και αυτό έχει δώσει τη δυνατότητα στους επιστήμονες επεξεργασίας δεδομένων να έχουν πλήθος χρήσιμων δεδομένων για να πραγματοποιούν τα πειράματά τους. Για τους προαναφερθέντες λόγους τα νευρωνικά δίκτυα επελέγησαν για την υλοποίηση ενός συστήματος που θα παράγει συγχορδίες που θα συνοδεύουν τις μελωδίες εισόδου.

Η εκπαίδευση των νευρωνικών δικτύων έγινε πάνω σε δεδομένα σε συμβολική γραφή. Συγκεκριμένα το σύνολο δεδομένων που χρησιμοποιήθηκε περιείχε αρχεία σε format MusicXML τα οποία για της ανάγκες της παρούσας διπλωματικής εργασίας μετατράπηκαν σε πίνακες (numpy arrays).

1.3. Στόχος της διπλωματικής εργασίας

Στόχος της συγκεκριμένης διπλωματικής εργασίας είναι να γίνει μία λεπτομερής σύγκριση διάφορων μεθόδων και μοντέλων που χρησιμοποιούνται για την εναρμόνιση μελωδιών καθώς και να εντοπιστούν τα πλεονεκτήματα και μειονεκτήματα που παρουσιάζει το κάθε μοντέλο. Επίσης αξίζει να σημειωθεί ότι στην συγκεκριμένη εργασία επιλέχθηκαν μοντέλα που χρησιμοποιούν διαφορετικού είδους νευρωνικά δίκτυα (Convolutional – Recurrent Neural Networks) έτσι ώστε να αξιολογηθεί η επιλογή του κάθε δικτύου.

Παρατηρώντας τα διάφορα μοντέλα για εύρεση συγχορδιών που χρησιμοποιούνται ευρέως σήμερα εύκολα διαπιστώνει κάποιος ότι τα δεδομένα να χρησιμοποιούνται για την εκπαίδευσή τους είναι στην συντριπτική τους πλειοψηφία έργα του Bach και για την ακρίβεια συνήθως Chorales. Αυτό συμβαίνει γιατί τα συγκεκριμένα μουσικά κομμάτια έχουν κάποια χαρακτηριστικά που τα καθιστούν τα πλέον κατάλληλα για τα συγκεκριμένα έργα. Τέτοια χαρακτηριστικά είναι οι καθορισμένοι κανόνες που διέπουν την εναρμόνιση της κύριας μελωδίας (μέθοδος της αντίστιξης) και ο σταθερός αριθμός φωνών (4). Αυτό σημαίνει ότι κάθε χρονική στιγμή μπορούν να ηχούν ταυτόχρονα μέχρι 4 νότες. Στα πλαίσια αυτής της διπλωματικής εργασίας χρησιμοποιήθηκε για την εκπαίδευση των διαφόρων μοντέλων ένα dataset περίπου 9 χιλιάδων κομματιών διαφόρων μουσικών ειδών από το HookTheory (www.hooktheory.com). Η επιλογή του συγκεκριμένου dataset έγινε για δύο λόγους: μεγάλος αριθμός τραγουδιών για την εκπαίδευση του δικτύου και σχετικά μεγάλη ποικιλία μουσικών ειδών. “Σχετικά” μεγάλη γιατί τα κομμάτια στην πλειοψηφία τους είναι Pop αλλά σε αντίθεση με τα dataset των chorales η ποικιλία μουσικών ειδών είναι σίγουρα μεγαλύτερη. Έτσι, σαν δευτερεύον στόχος της διπλωματικής εργασίας είναι η σύγκριση των αποτελεσμάτων των δύο dataset και μια εκτίμηση για τα στοιχεία που καθιστούν ένα dataset κατάλληλο για εκπαίδευση νευρωνικών δικτύων.

Εκτός από την αξιολόγηση των νευρωνικών δικτύων και των dataset που χρησιμοποιήθηκαν για την εκπαίδευση των ανωτέρω μοντέλων, στα πλαίσια της αυτής της διπλωματικής εργασίας γίνεται και μία σύγκριση διαφόρων τεχνικών-μεθόδων sampling αποτελεσμάτων.

Τέλος, κρίνεται σημαντικό τα μοντέλα που εκπαιδεύτηκαν κατά τη διάρκεια εκπόνησης της εργασίας να γίνουν διαθέσιμα σε έναν απλό χρήστη μέσω μίας χρηστικής web εφαρμογής. Για το λόγο αυτό δημιουργήθηκε ένα web application στο οποίο ο χρήστης μπορεί να κάνει upload τη μελωδία για την οποία επιθυμεί να βρει συνοδευτικές συγχορδίες σε midi μορφή και να λάβει μέσω download ένα midi αρχείο εξόδου που θα περιέχει την εναρμονισμένη μελωδία. Η εναρμόνιση της μελωδίας θα γίνεται με βάση το μοντέλο που επιλέγει ο χρήστης.

1.3. Δομή διπλωματικής εργασίας

Η παρούσα διπλωματική εργασία αποτελείται από 8 κεφάλαια. Στο κεφάλαιο 2 γίνεται αναφορά σε βασικές έννοιες που θα αναφερθούν στη συνέχεια της εργασίας και η κατανόηση τους αποτελεί απαραίτητη προϋπόθεση για την κατανόηση της εργασίας. Το κεφάλαιο χωρίζεται σε 4 θεματικές ενότητες οι οποίες είναι η μουσική θεωρία, τα αναδρομικά νευρωνικά δίκτυα, τα συνελικτικά νευρωνικά δίκτυα και οι μέθοδοι διεξαγωγής προβλέψεων.

Το κεφάλαιο 3 αφορά τα σύνολα δεδομένων που χρησιμοποιήθηκαν στην εργασία, την προέλευσή τους, τα στάδια προ-επεξεργασίας τους, και τα σημαντικότερα χαρακτηριστικά τους.

Στο κεφάλαιο 4 γίνεται λεπτομερής αναφορά στην πειραματική διαδικασία που ακολουθήθηκε. Παρουσιάζεται η αρχιτεκτονική των αναδρομικών νευρωνικών δικτύων που εξετάστηκαν, τα αποτελέσματα που προέκυψαν,η αξιολόγηση των αποτελεσμάτων αυτών και των μοντέλων γενικότερα.

Το κεφάλαιο 5 εστιάζει στα συνελικτικά νευρωνικά δίκτυα και συγκεκριμένα στο Coconet. Περιγράφεται η αρχιτεκτονική του και τα αποτελέσματα που προέκυψαν από την εκπαίδευση του πάνω στα δεδομένα που συλλέχθηκαν ενώ επιπλέον γίνεται σύγκριση των αναδρομικών μοντέλων και του Coconet και σχολιάζονται τα αποτελέσματα τους.

Στο κεφάλαιο 6 παρουσιάζονται τα συμπεράσματα μας από της συγκεκριμένη διπλωματική εργασία και γίνεται αναφορά σε πιθανές μελλοντικές εφαρμογές.

Τέλος, το κεφάλαιο 7 αναλύονται οι λεπτομέρειες υλοποίησης της εργασίας και των εργαλείων που χρησιμοποιήθηκαν για αυτή.

2.Θεωρητικό υπόβαθρο

2.1. Βασικές έννοιες της μουσικής θεωρίας

2.1.1. Νότες και διαστήματα

Μελωδία είναι μία αλληλουχία μουσικών φθόγγων ή αλλιώς νοτών. Μπορούμε να ορίσουμε ως μουσικό φθόγγο/μουσική νότα ένα ηχητικό κύμα με σταθερή συχνότητα ανάμεσα στα 16 με 8000 Hz. Ορίζουμε ως οκτάβα το διάστημα μεταξύ ενός μουσικού φθόγγου και ενός άλλου με τη μισή ή τη διπλάσια συχνότητα. Στη μουσική σημειογραφία, οι φθόγγοι που απέχουν μια οκτάβα μεταξύ τους έχουν το ίδιο όνομα και λέμε ακόμη πως έχουν τον ίδιο "τόνο", αφού ηχούν πανομοιότυπα. Κάθε οκτάβα στη δυτική μουσική αποτελείται από δώδεκα νότες, οι οποίες απέχουν ίση απόσταση μεταξύ τους. Η απόσταση αυτή ονομάζεται ημιτόνιο. Ένα διάστημα που δημιουργείται από τη σύνθεση δύο ημιτονίων, δηλαδή αν μεταξύ δύο νοτών παρεμβάλλεται ακριβώς μία νότα, ονομάζεται τόνος. Για τις ανάγκες της διπλωματικής εργασίας οι νότες αναπαριστώνται με αριθμούς. Στον παρακάτω πίνακα φαίνεται η αντιστοίχιση νοτών και αριθμών.

Ντο	0
Ντο # / Ρε ♭	1
Ρε	2
Ρε # / Μι ♭	3
Μι	4
Φα	5
Φα # / Σολ ♭	6
Σολ	7
Σολ # / Λα ♭	8
Λα	9
Λα # / Σι ♭	10
Σι	11

Για παράδειγμα, όπως βλέπουμε από τον παραπάνω πίνακα το διάστημα μεταξύ των νοτών Μι και Φα είναι ένα διάστημα ημιτονίου καθώς δεν παρεμβάλλεται κάποια άλλη νότα μεταξύ τους, ενώ αντίθετα το διάστημα μεταξύ Ντο και Ρε είναι διάστημα τόνου καθώς μεταξύ τους παρεμβάλλεται μία νότα, η Ντο # .

Κάθε μουσικό κομμάτι είναι γραμμένο σε μία κλίμακα. Λέγοντας κλίμακα ή σκάλα εννοούμε μία αλληλουχία νοτών που χρησιμοποιούμε για να καλύψουμε το διάστημα της μίας οκτάβας ανεβαίνοντας ή κατεβαίνοντας. Στη δυτική μουσική έχει καθιερωθεί να χρησιμοποιούμε 7 νότες για

να καλύψουμε το διάστημα αυτό, για το λόγο αυτό έχουν οριστεί ονομαστικά 7 νότες, έτσι για τις υπόλοιπες 5 νότες χρησιμοποιούμε δύο σημεία αλλοίωσης, τη δίεση (#) και την ύφεση (♭), τα οποία δηλώνουν ότι μία νότα είναι αυξημένη κατά ένα ημιτόνιο ή ελαττωμένη κατά ένα ημιτόνιο. Προφανώς υπάρχουν κλίμακες με περισσότερες ή λιγότερες από 7 διαφορετικές νότες αλλά δεν θα ασχοληθούμε με αυτές στα πλαίσια της αυτής της διπλωματικής εργασίας. Περισσότερα για τις κλίμακες θα αναφέρουμε στο κεφάλαιο “Τρόποι και δανεισμοί συγχορδιών”. Κάθε κλίμακα έχει μια αρχική νότα. Η νότα αυτή ονομάζεται τονική. Δύο όμοιες κλίμακες με διαφορετική τονική νότα είναι στην πραγματικότητα η ίδια κλίμακα απλά μετατοπισμένη κατά το μέτρο της απόστασης των τονικών νοτών της κάθε κλίμακας. Για την καλύτερη εκπαίδευση των δικτύων σε όλα τα δεδομένα καθορίστηκε για τονική νότα η να Ντο. Μεγαλύτερη αναφορά σε αυτό το κομμάτι θα γίνει στο κεφάλαιο της αναπαράστασης των δεδομένων.

Κάθε κλίμακα αποτελείται από 7 διαφορετικές νότες. Η κάθε νότα αντιστοιχεί σε μία βαθμίδα της κλίμακας (scale degree). Για παράδειγμα, στην Ντο Μείζονα κλίμακα η νότα κάθε βαθμίδας είναι οι ακόλουθες:

Βαθμίδα	Νότα
1η	Ντο
2η	Ρε
3η	Μι
4η	Φα
5η	Σολ
6η	Λα
7η	Σι

Ενώ στην Ντο Ελάσσονα η νότες της κάθε βαθμίδας είναι οι ακόλουθες:

Βαθμίδα	Νότα
1η	Ντο
2η	Ρε
3η	Μι ♭
4η	Φα
5η	Σολ
6η	Λα ♭
7η	Σι ♭

Το ποια νότα αντιστοιχεί σε κάθε βαθμίδα κάθε κλίμακας αναλύεται εκτενέστερα στο κεφάλαιο “Τρόποι και δανεισμοί συγχορδιών”.

2.1.2. Συγχορδίες

Σαν αρμονία ενός κομματιού ορίζεται η αλληλουχία συγχορδιών ή φωνών που συνοδεύουν τη μελωδία, την κύρια φωνή. Η σύνθεση της μελωδίας με την αρμονία δίνει ως αποτέλεσμα το ολοκληρωμένο κομμάτι.

Σαν συγχορδία ορίζεται στη θεωρία της μουσικής το σύνολο τριών η περισσότερων νοτών που ηχούν ταυτόχρονα. Στο δυτικό σύστημα που εξετάζουμε στα πλαίσια αυτής της διπλωματικής εργασία μια συγχορδία σχηματίζεται από νότες ενός τρόπου που απέχουν μεταξύ τους ένα διάστημα τρίτης. Διάστημα τρίτης ονομάζεται το διάστημα μεταξύ δύο νοτών ενός τρόπου, μεταξύ των οποίων παρεμβάλλεται ακριβώς μία νότα. Υπάρχουν πολλά είδη συγχορδιών αλλά τα σημαντικότερα και αυτά που χρησιμοποιήθηκαν στο πλαίσιο της διπλωματικής αυτής εργασίας είναι τρία και είναι τα εξής.

- Μείζονες
- Ελάσσονες
- Ελαττωμένες

Κάθε συγχορδία εκτός από το είδος της προσδιορίζεται και από την βασική της νότα, ή αλλιώς θεμέλιο φθόγγο. Όπως και στις νότες, η τονική συγχορδία είναι η συγχορδία της πρώτης βαθμίδας της κλίμακας. Με αυτόν τον τρόπο, κάποιος μπορεί να αναφερθεί σε μία συγχορδία με το είδος της και την αντίστοιχη τονική βαθμίδα της. Για παράδειγμα η Ντο Μείζονα συγχορδία (C) είναι μία μείζονα συγχορδία με θεμέλιο φθόγγος της τη νότα Ντο. Εκτός από την Ντο, στη Ντο μείζονα συγχορδία θα ανήκει και η νότα της τρίτης βαθμίδας της Ντο μείζονας κλίμακας, δηλαδή η νότα Μι καθώς και η νότα της πέμπτης βαθμίδας της Ντο μείζονας κλίμακας, δηλαδή η νότα Σολ.

Όπως αναφέραμε οι συγχορδίες αποτελούνται από 3 ή περισσότερους μουσικούς φθόγγους. Ο πιο κοινός τύπος συγχορδιών είναι οι τρίφωνες συγχορδίες, δηλαδή αυτές που αποτελούνται από ακριβώς τρεις νότες. Παρ’ όλα αυτά μπορούν να υπάρξουν και συγχορδίες με περισσότερες από τρεις νότες, οι λεγόμενες συγχορδίες με επεκτάσεις. Οι πιο συνηθισμένες είναι οι συγχορδίες με επιπλέον νότα τη νότα της έβδομης βαθμίδας της κλίμακας, αλλά μπορεί να περιέχουν και άλλες νότες (πχ 9η, 11η, 13η βαθμίδα). Δηλαδή στην περίπτωση της Ντο μείζονας συγχορδίας, μπορούμε να δημιουργήσουμε την Ντο μείζονα συγχορδία μεθ’ εβδόμης (Cmaj7) προσθέτοντας την νότα της έβδομης βαθμίδας της Ντο μείζονας κλίμακας, δηλαδή τη νότα Σι.

Τέλος, είναι χρήσιμο, με την έννοια ότι υλοποιήθηκαν στη δημιουργία των dataset για κάποια μοντέλα, να κάνουμε μία σύντομη αναφορά στο τι είναι αναστροφή μίας συγχορδίας. Όταν λέμε ότι μία συγχορδία είναι ανεστραμμένη εννοούμε ότι η πιο μπάσα νότα της δεν είναι η τονική της αλλά κάποια άλλη από της νότες της συγχορδίας, για παράδειγμα η τρίτη (γνωστή και ως πρώτη αναστροφή) ή η πέμπτη (γνωστή και ως δεύτερη αναστροφή. Όπως μπορεί κανείς εύκολα να αντιληφθεί, οι τρίφωνες συγχορδίες έχουν δύο αναστροφές, αλλά δεν ισχύει το ίδιο για τις συγχορδίες με περισσότερες από 3 νότες. Για παράδειγμα αν προσθέσουμε σε μία συγχορδία και την έβδομη νότα της κλίμακάς της, τότε μπορεί να σχηματιστεί μία επιπλέον αναστροφή, αυτή με την νότα της έβδομης βαθμίδας σαν πιο μπάσα νότα (γνωστή και ως τρίτη αναστροφή).

2.1.3. Κλίμακες και Τρόποι

Αρχικά πρέπει να ορίσουμε μία νότα ως τονική. Είδαμε πως αν καλύψουμε ένα διάστημα οκτάβας, φτάνουμε ξανά σε μια νότα με το ίδιο όνομα και τόνο με την τονική. Ο τρόπος δηλώνει τον τύπο σχηματισμού μιας κλίμακας που αποτελείται από 7 μουσικούς φόγγους, δηλαδή τη διαδοχή τόνων και ημιτονίων που επιλέγουμε μέχρι να διανύσουμε ένα διάστημα οκτάβας. Ας εξετάσουμε τώρα το μείζονα τρόπο με τονική νότα τη Ντο, δηλαδή το Ντο μείζονα τρόπο, ο οποίος δημιουργείται με την χρήση όλων των φυσικών νοτών. Σαν φυσικές νότες ορίζουμε τις νότες που δεν έχουν κάποια αλλοίωση. Βλέπουμε, λοιπόν, πως για να διανύσουμε το διάστημα της οκτάβας χρησιμοποιούμε διαδοχικά τις νότες Ντο-Ρε-Μι-Φα-Σολ-Λα-Σι και αντίστοιχα διαστήματα Τ-Τ-Η-Τ-Τ-Τ-Η, όπου το Τ είναι διάστημα τόνου και Η διάστημα ημιτονίου. Η συγκεκριμένη αλληλουχία διαστημάτων ορίζουν τον Μείζονα τρόπο. Αν τώρα ξεκινήσουμε από μια άλλη φυσική νότα πχ τη νότα Ρε, βλέπουμε πως μπορούμε με τις ίδιες νότες να ξαναφτάσουμε στη νότα Ρε αλλά η διαδοχή των διαστημάτων έχει αλλάξει. Τώρα έχουμε μια διαδοχή Τ-Η-Τ-Τ-Τ-Η-Τ, η οποία αποτελεί έναν διαφορετικό τρόπο από το μείζονα για την κάλυψη ενός διαστήματος οκτάβας. Με τονική κάθε φυσική νότα και χρήση μόνο φυσικών νοτών μπορούμε να κατασκευάσουμε επτά διαφορετικούς φυσικούς τρόπους, όσες και οι φυσικές νότες, οι οποίοι είναι οι εξής:

- Μείζων/Ιωνικός τρόπος (τρόπος του Ντο), major mode
- Δώριος τρόπος (τρόπος του Ρε), dorian mode
- Φρύγιος τρόπος (τρόπος του Μι), phrygian mode
- Λύδιος τρόπος (τρόπος του Φα), lydian mode
- Μιξολύδιος τρόπος (τρόπος του Σολ), mixolydian mode
- Ελάσσων/Αιολικός τρόπος (τρόπος του Λα), minor mode
- Λόκριος τρόπος (τρόπος του Σι), locrian mode

Στις αρχές του 16ου αιώνα, με τη μετάβαση από την Αναγέννηση στο Μπαρόκ, η μουσική προσανατολίστηκε κυρίως στη χρήση δύο μόνο μουσικών τρόπων, του Ιωνικού και του Αιολικού, οι οποίοι εν τέλει καθιερώθηκαν ως μείζονας τρόπος και ελάσσονας τρόπος. Αυτοί οι δύο τρόποι εξακολουθούν να χρησιμοποιούνται κατά πλειοψηφία και στη σύγχρονη μουσική.

Είδαμε πώς με τη χρήση των επτά φυσικών νοτών δημιουργούνται επτά διαφορετικοί τρόποι, δηλαδή επτά διαφορετικές ακολουθίες διαστημάτων για την κάλυψη ενός διαστήματος οκτάβας. Ο τρόπος είναι, ουσιαστικά, αυτή η καθορισμένη ακολουθία. Οπότε, από κάθε διαφορετική νότα ως τονική, μπορούμε, τηρώντας αυτήν την ακολουθία, να παράξουμε και τους επτά διατονικούς τρόπους. Για παράδειγμα, ο dorian mode έχει την εξής διαδοχή διαστημάτων: Τ-Τ-Η-Τ-Τ-Τ-Η. Για να κατασκευάσουμε, λοιπόν το Ντο dorian mode ξεκινάμε από τη νότα Ντο ως τονική και επιλέγουμε την επόμενη νότα με βάση τα καθορισμένα διαστήματα. Έτσι, σχηματίζεται η εξής ακολουθία: Ντο-Ρε-Μι ύφεση-Φα-Σολ-Λα-Σι ύφεση. Οι επτά διαφορετικοί τρόποι που παράγονται από μία τονική νότα είναι παράλληλοι μεταξύ τους, δηλαδή για παράδειγμα ο Ντο μείζων τρόπος έχει ως παράλληλο λύδιο τρόπο τον Ντο λύδιο τρόπο.

Κάθε τρόπος, όπως είδαμε, αποτελείται από 7 διαφορετικούς φθόγγους. Στους τρόπους της δυτικής μουσικής έχει επικρατήσει να ονομάζεται βαθμίδα η θέση που καταλαμβάνει ο κάθε φθόγγος στην κλίμακα, και συνήθως αναφερόμαστε στον κάθε φθόγγο με την ονομασία της αντίστοιχης βαθμίδας. Η βαθμίδες προσδιορίζονται συνήθως με λατινικούς αριθμούς, με αρχική βαθμίδα την τονική. Για παράδειγμα στον Ντο μείζονα τρόπο η αντιστοιχία βαθμίδων - νοτών είναι η εξής:

Αριθμός βαθμίδας	Νότα
I	Ντο
II	Ρε
III	Μι
IV	Φα
V	Σολ
VI	Λα
VII	Σι

2.1.4. Δευτερεύουσες συγχορδίες και δανεισμοί συγχορδιών

Η αρμονία ενός κομματιού δεν είναι απαραίτητο να περιέχει μόνο νότες από την βασική κλίμακα πάνω στην οποία είναι γραμμένο το κομμάτι και αντίστοιχα η αρμονία του δεν είναι απαραίτητο να περιέχει μόνο συγχορδίες που ανήκουν στη συγκεκριμένη κλίμακα. Αυτό μπορεί να γίνει με δύο τρόπους.

Ο πρώτος τρόπος είναι με τη χρήση κάποιον συγχορδιών, των λεγόμενων secondary dominants. Secondary dominant chord είναι η συγχορδία της πέμπτης βαθμίδας μιας κλίμακας, η οποία έχει σαν τονική μία νότα (συνήθως την 4η, 5η η 7η) της βασικής κλίμακας και είναι μείζονος τρόπου. Αυτό γίνεται για να προετοιμάσουμε την αλλαγή της κλίμακας σε ένα κομμάτι.

Ο δεύτερος τρόπος είναι με το λεγόμενο δανεισμό συγχορδιών από παράλληλους τρόπους. Όπως είδαμε ο Ντο μείζον τρόπος έχει άλλους 6 παράλληλους τρόπους, οι οποίοι λόγω διαφορετικών διαστημάτων μεταξύ κάθε βαθμίδας έχουν διαφορετικές νότες και άρα διαφορετικές συγχορδίες μεταξύ τους. Δανεισμός μίας συγχορδίας είναι όταν χρησιμοποιούμε μία συγχορδία που δεν ανήκει στον κύριο τρόπο στον οποίο είναι γραμμένο το κομμάτι αλλά αντίθετα ανήκει σε κάποιον άλλο παράλληλο τρόπο.

2.1.5. Μέτρο και Χρονική αξία

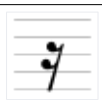
Κάθε κομμάτι χαρακτηρίζεται από το μέτρο του. Το μέτρο είναι ένα κλάσμα και υποδεικνύει τη συνολική των νοτών που συμπεριλαμβάνονται

Κάθε νότα εκτός από την τονικότητα της χαρακτηρίζεται και από τη χρονική της αξία. Η χρονική αξία μίας νότας καθορίζει τη χρονική της διάρκεια. Οι χρονικές αξίες που υπάρχουν είναι οι εξής:

Όνομα	Σύμβολο	Αξία
Ολόκληρο		1
Μισό		1/2
Τέταρτο		1/4
Όγδοο		1/8
Δέκατο έκτο		1/16

Υπάρχουν και μικρότερες αξίες (π.χ. τριακοστό δεύτερο κ.ο.κ.) αλλά στα πλαίσια της συγκεκριμένης διπλωματικής εργασίας η μικρότερη υποδιαίρεση της χρονικής αξίας μιας νότας είναι το δέκατο έκτο.

Είναι δυνατό σε κάποιο σημείο ενός μουσικού κομματιού να μην ηχεί κάποια νότα. Αυτό στην μουσική ορίζεται ως παύση. Όπως οι νότες, έτσι και οι παύσεις χαρακτηρίζονται από τη χρονική τους αξία καθώς θα πρέπει να ορίζεται σαφώς το χρονικό διάστημα κατά το οποίο δεν ηχεί καμία νότα. Ακριβώς όπως και στην περίπτωση των νοτών, έτσι και στην περίπτωση των παύσεων οι χρονικές αξίες είναι οι εξής:

Όνομα	Σύμβολο	Αξία
Παύση ολοκλήρου		1
Παύση μισού		1/2
Παύση τετάρτου		1/4
Παύση ογδόου		1/8
Παύση δεκάτου έκτου		1/16

2.2. Αναδρομικά νευρωνικά Δίκτυα

Οι άνθρωποι, όταν διαβάζουν ένα κείμενο για παράδειγμα, αντιλαμβάνονται το νόημα κάθε λέξης βασιζόμενοι στην κατανόηση των προηγούμενων λέξεων. Δεν επικεντρώνονται σε κάθε λέξη ξεχωριστά, ξεχνώντας τις προηγούμενες. Τα παραδοσιακά εμπροσθοτροφοδοτούμενα νευρωνικά δίκτυα δεν λειτουργούν όμως με αυτόν τον τρόπο, και αυτό είναι μία βασική τους αδυναμία. Για να αντιμετωπιστεί η αδυναμία αυτή δημιουργήθηκαν τα αναδρομικά νευρωνικά δίκτυα.

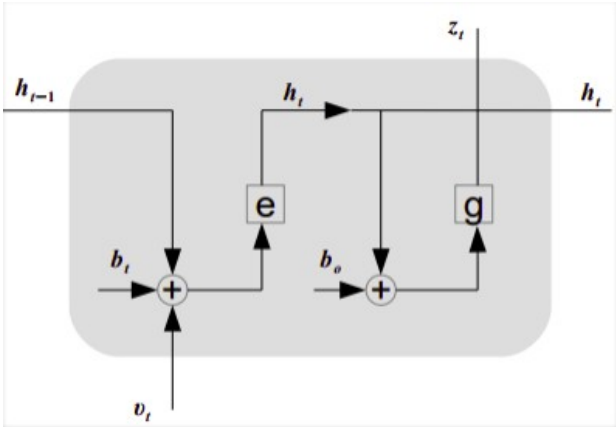
Το αναδρομικό νευρωνικό δίκτυο (Recurrent Neural Network, RNN) αποτελεί μία αρχιτεκτονική μεγάλων δυνατοτήτων που μπορεί να χειριστεί δεδομένα ακολουθιών και για αυτό το λόγο μπορεί και έχει χρησιμοποιηθεί σε μοντέλα που αποσκοπούν στην παραγωγή μουσικής με αυτόματο τρόπο. Αυτό έγκειται στο γεγονός ότι ένα μουσικό κομμάτι μπορεί να γίνει αντιληπτό και να αναπαρασταθεί σαν σύνθεση δύο ακολουθιών, μία της μελωδίας και μία της αρμονίας. Η διεπαφή του επαναληπτικού δικτύου είναι απλή και συνήθης στα νευρωνικά, δέχεται ένα διάνυσμα σταθερού μεγέθους ως είσοδο και παράγει διάνυσμα σταθερού μεγέθους στην έξοδο (για παράδειγμα, κατανομή πιθανότητας). Το ξεχωριστό με τα επαναληπτικά δίκτυα είναι η ικανότητα χειρισμού χρονικών ακολουθιών τέτοιων διανυσμάτων, όπως παρουσιάζεται αναλυτικά στη συνέχεια.

Για να κατανοήσουμε τη λειτουργία των αναδρομικών νευρωνικών δικτύων κρίνεται σκόπιμη μία σύντομη αναφορά στα εμπροσθοτροφοδοτούμενα νευρωνικά δίκτυα. Και αυτό γιατί και τα δύο αυτά είδη νευρωνικών δικτύων αντλούν την ονομασία τους από τον τρόπο με τον οποίο μεταφέρουν την πληροφορία μέσα από μια σειρά μαθηματικών πράξεων που λαμβάνουν χώρα στους νευρώνες του δικτύου. Τα εμπροσθοτροφοδοτούμενα νευρωνικά δίκτυα προωθούν πληροφορίες προς τα επόμενα επίπεδα του δικτύου ενώ τα αναδρομικά νευρωνικά δίκτυα ανακυκλώνουν την πληροφορία μεταδίδοντας της από επόμενα επίπεδα του δικτύου σε προηγούμενα. Άρα, αν θέλουμε να δώσουμε έναν σύντομο ορισμό, ένα νευρωνικό δίκτυο λέγεται αναδρομικό, εάν υπάρχει έστω και μια σύνδεση από έναν νευρώνα επιπέδου i προς έναν νευρώνα επιπέδου j, όπου $j \leq i$. Ένα εμπροσθοτροφοδοτούμενο νευρωνικό δίκτυο εκπαιδεύεται πάνω σε επισημασμένα δεδομένα μέχρι να ελαχιστοποιηθεί η συνάρτηση κόστους. Ένα εκπαιδευμένο εμπροσθοτροφοδοτούμενο νευρωνικό δίκτυο που μπορεί να κληθεί να κάνε προβλέψεις για ένα σύνολο δεδομένων, θα κάνει την πρόβλεψή του για οποιοδήποτε δείγμα χωρίς να επηρεάζεται από τις προβλέψεις που έκανε στα προηγούμενα δείγματα. Αυτό συμβαίνει γιατί ένα εμπροσθοτροφοδοτούμενο δίκτυο δεν έχει αίσθηση της χρονικής σειράς και κάθε φορά η μόνη είσοδος που λαμβάνει υπόψη είναι το δείγμα που του δίνεται σαν είσοδος εκείνη τη χρονική στιγμή.

Εδώ έγκειται η χρησιμότητα και, εν τέλη, ο λόγος ύπαρξης των αναδρομικών νευρωνικών δικτύων. Τα αναδρομικά νευρωνικά δίκτυα δέχονται σαν είσοδο όχι μόνο την είσοδο εκείνη τη χρονική στιγμή αλλά και ο,τι έχουν δεχτεί τις προηγούμενες χρονικές στιγμές. Αυτό σημαίνει ότι η πρόβλεψη που κάνει ένα αναδρομικό νευρωνικό δίκτυο μία χρονική στιγμή t επηρεάζεται από την πρόβλεψη που έκανε τη χρονική στιγμή $t-1$. Άρα τα αναδρομικά νευρωνικά δίκτυα έχουν δύο ειδών εισόδους, την παρούσα και τις προηγούμενες προβλέψεις, των οποίων ο συνδυασμός καθορίζει το πως θα αντιδράσουν σε νέα δεδομένα.

2.2.1. Αρχιτεκτονική και Εκπαίδευση αναδρομικών νευρωνικών δικτύων

Το απλό επαναληπτικό νευρωνικό δίκτυο (Simple Recurrent Network, SRN) αποτελεί ένα μη γραμμικό δυναμικό σύστημα το οποίο παραμετροποιείται από τρεις πίνακες βαρών (weights: $\boldsymbol{W_{hv}}$, $\boldsymbol{W_{hh}}$, $\boldsymbol{W_{oh}}$) και τρία σταθερά διανύσματα (bias: $\boldsymbol{b_h}$, $\boldsymbol{b_o}$, $\boldsymbol{h_o}$). Το απλό νευρωνικό δίκτυο δεχόμενο σαν είσοδο μία ακολουθία (v_1, ..., v_t), η οποία συμβολιζεται διανυσματικά σαν $\boldsymbol{v^T}$, υπολογίζει μία ακολουθία κρυφών καταστάσεων (hidden states) καθώς και μία ακολουθία εξόδων $\boldsymbol{z^T}$. Οι ακολουθίες αυτές προκύπτουν μετά από μαθηματικές πράξεις οι οποίες περιγράφονται αναλυτικά στον παρακάτω αλγόριθμο.



Αλγόριθμος 1: Εμπρόσθια τροφοδότηση αναδρομικού δικτύου

1: for t from 1 to T do
2: $\boldsymbol{u_t} \leftarrow \boldsymbol{W_{hvvt}} + \boldsymbol{W_{hhht}} - 1 + \boldsymbol{b_h}$
3: $\boldsymbol{h_t} \leftarrow e(\boldsymbol{u_t})$
4: $\boldsymbol{o_t} \leftarrow \boldsymbol{W_{ohht}} + \boldsymbol{b_o}$
5: $\boldsymbol{z_t} \leftarrow g(\boldsymbol{o_t})$
6: end for

όπου οι συναρτήσεις *e* και *g* είναι μη γραμμικές συναρτήσεις ενεργοποίησης της κρυφής κατάστασης και της εξόδου του απλού αναδρομικού νευρωνικού δικτύου, ενώ υπάρχει και ένα διάνυσμα h_0 το οποίο αντιστοιχεί στην αρχική κρυφή κατάσταση του νευρωνικού δικτύου. Η συνάρτηση απώλειας (loss function) του αναδρομικού νευρωνικού δικτύου είναι συνήθως το άθροισμα των συναρτήσεων κόστους (cost function) ανά χρονική στιγμή.

$$L(z,y)=\sum_{t=1}^{T} L(z_t;y_t)$$

όπου $\boldsymbol{z_t}$ όπως είδαμε είναι η πρόβλεψη του δικτύου και $\boldsymbol{y_t}$ η επιθυμητή έξοδος.

Η οπισθοδιάδοση στα εμπροσθοτροφοδοτούμενα νευρωνικά δίκτυα εκκινεί από την έξοδο του δικτύου και το υπολογισμό του τελικού σφάλματος και διατρέχει τους νευρώνες του δικτύου από τα

τελευταία επίπεδα προς τα πρώτα, αναθέτοντας στους πίνακες βαρών του κάθε νευρώνα ένα μέρος της ευθύνης για το τελικό σφάλμα. Αυτό γίνεται με τον υπολογισμό των μερικών παραγώγων του τελικού σφάλματος προς τα βάρη του κάθε νευρώνα. Στην περίπτωση των αναδρομικών νευρωνικών δικτύων χρησιμοποιείται μία παραλλαγή του αλγορίθμου οπισθοδιάδοσης που χρησιμοποιείται στα εμπροσθοδτροφοδοτούμενα νευρωνικά δίκτυα. Ο αλγόριθμος αυτός ονομάζεται οπισθοδιάδοση στο χρόνο (Backpropagation through time, BPTT) και διαφέρει από την κλασσική οπισθοδιάδοση στο γεγονός ότι εκτός από την επίσκεψη των κόμβων του δικτύου από τα τελευταία επίπεδα προς τα προηγούμενα, στα αναδρομικά δίκτυα ο αλγόριθμος οπισθοδιάδοσης στον χρόνο συνδέει κάθε χρονική στιγμή με την προηγούμενή της. Αναλυτικά ο αλγόριθμος της οπισθοδιάδοσης στο χρόνο ενός αναδρομικού νευρωνικού δικτύου παρουσιάζεται παρακάτω.

Αλγόριθμος 2: Οπισθοδιάδοση στο χρόνο βαρών επαναληπτικού δικτύου (BPTT; Werbos, 1990; Rumelhart et al., 1986)

1: for** t **from** T **downto** 1 **do

2: $do_t \leftarrow g'(o_t).dL\frac{(z_t;y_t)}{dz_t}$

3: $db_o \leftarrow db_o + do_t$

4: $dW_{oh} \leftarrow dW_{oh} + do_t h_t^T$

5: $dh_t \leftarrow dh_t + W_{oh}^T do_t$

6: $dz_t \leftarrow e'(z_t).dh_t$

7: $dW_{hu} \leftarrow dW_{hu} + dz_t v_t^T$

8: $db_h \leftarrow db_h + dz_t$

9: $dW_{hh} \leftarrow dW_{hh} + dz_t h_{t-1}^T$

10: $dh_{t-1} \leftarrow W_{hh}^T dz_t$

11: ***end for***

12: ***return*** $d\theta = [dW_{hu}, dW_{hh}, dW_{oh}, db_h, db_o, dh_o]$

Διόγκωση και συρρίκνωση παραγώγων

Παρόλο το γεγονός ότι οι μερικοί παράγωγοι ενός αναδρομικού νευρωνικού δικτύου είναι εύκολο να υπολογιστούν, τα αναδρομικά νευρωνικά δίκτυα παρουσιάζουν από τη φύση τους δυσκολίες κατά τη διαδικασία εκπαίδευσής τους, και ιδιαίτερα σε προβλήματα με χρονικές εξαρτήσεις μεγάλου εύρους (με μεγάλες ακολουθίες εισόδου παραδείγματος χάρη). Η δυσκολία στην εκπαίδευση του αναδρομικού νευρωνικού δικτύου έγκειται στην επαναληπτική μη γραμμική φύση του.

Τα δύο συχνότερα προβλήματα που προκύπτουν όταν πραγματευόμαστε πολύ μεγάλες ακολουθίες είναι αυτά της διόγκωσης και της συρρίκνωσης των παραγώγων όπως περιγράφονται από τους (Bengio, 1994 - Hochreiter and Schmidhuber, 1997). Εν συντομία, η διόγκωση των παραγώγων αναφέρεται στο φαινόμενο όταν οι παράγωγοι αυξάνονται εκθετικά καθώς η οπισθοδιάδοση προχωρά στο χρόνο, μετατρέποντάς την σε “κολοβή” οπισθοδιάδοση στο χρόνο, η οποία είναι ανίκανη να συγκρατήσει εξαρτήσεις μεγάλης έκτασης στις ακολουθίες. Η βασική αιτία των προβλημάτων αυτών έγκειται στην γραμμή 10 του αλγόριθμου οπισθοδιάδοσης βαρών στο χρόνο (Αλγόριθμος 2). Αν αγνοήσουμε τις ενδιάμεσες απώλειες, ένα σήμα που έχει οπισθοδιαδοθεί

από την τρέχουσα κατάσταση, για K βήματα, θα γίνει $dh_{t-k} = \prod_{i=1}^{K} \left(W_{hh}^{T} . dz_t \right)$. Βέβαια, η z_t είναι μια συνάρτηση (πχ. tanh ή σιγμοειδής) η οποία είναι ομαλή, άρα η συμπεριφορά του δικτύου καθορίζεται από τα χαρακτηριστικά του επαναληπτικού πίνακα Whh και οι περισσότερες αναλύσεις τον εξετάζουν αναφορικά με την μεγαλύτερη ιδιοτιμή του. Αν αυτή είναι πολύ μεγάλη ή υπερβολικά μικρή, υπάρχει ο κίνδυνος διόγκωσης ή συρρίκνωσης παραγώγων αντίστοιχα. (Bengio κ.α. 1994, Hochreiter and Schmidhuber, 1997, Martens and Sutskever, 2011, Pascanu κ.α. 2013).

Διόγκωση παραγώγων

Γενικά, είναι αρκετά εύκολο να χειριστούμε το πρόβλημα διόγκωσης παραγώγων με την εφαρμογή διάφορων μορφών ψαλιδίσματος παραγώγων. Η πρώτη προσέγγιση προτάθηκε από τον (Mikolov 2012) με τη μορφή του “ανά στοιχείο” ψαλιδίσματος: κάθε χρονική στιγμή, κατά τη διάρκεια της οπισθοδιάδοσης, κάθε συντεταγμένη του *dh* που είναι μεγαλύτερη από ένα κατώφλι τ, όπου τ > 0, ή μικρότερη από -τ, τίθεται ίση με τ ή -τ αντίστοιχα. Μια άλλη προσέγγιση είναι το ψαλίδισμα νόρμας παραγώγου που προτάθηκε από τους (Pascanu κ.α. 2013). Σ' αυτή την περίπτωση, η νόρμα του διανύσματος της παραγώγου g που έχει υπολογιστεί “ανά υποσύνολο των δεδομένων” συγκρίνεται με ένα θετικό κατώφλι τ. Αν το ξεπερνά, τότε το διάνυσμα της παραγώγου αντικαθίσταται από την ποσότητα $\tau \frac{g}{\|g\|}$. Σε πολλές περιπτώσεις οι δύο προσεγγίσεις συνδυάζονται.

Συρρίκνωση παραγώγων

Για το ζήτημα της συρρίκνωσης παραγώγων έχουν, επίσης, προταθεί αρκετοί τρόποι αντιμετώπισης, οι περισσότεροι από τους οποίους προσπαθούν να μετριάσουν τη μη γραμμικότητά των δικτύων καθιστώντας ωστόσο την εκπαίδευση λιγότερο βαθιά. Η καλύτερη προσέγγιση φαίνεται να είναι η προσθήκη μνήμης μακρού και βραχέος όρου (Long Short – Term Memory LSTM) που εφευρέθηκε από τους (Hochreiter και Schmidhuber, 1997) και βελτιώθηκε από τους (Gers κ.α., 2000), με την οποία θα ασχοληθούμε αναλυτικά στην ακόλουθη υποενότητα.

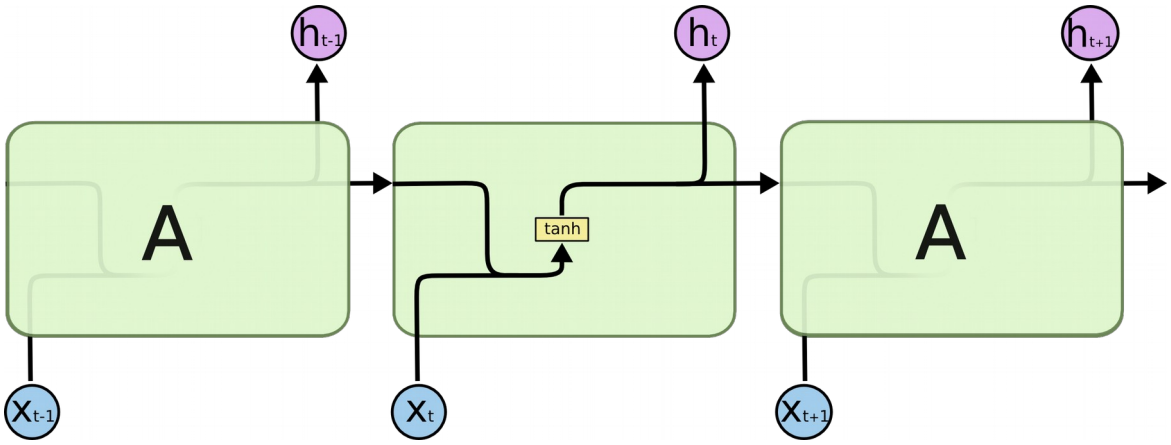
2.2.2. Δίκτυο Long Short-Term Memory (LSTM)

Τα δίκτυα μακράς μνήμης βραχέος όρου (Long Short-Term Memory, LSTM) αποτελούν ένα συγκεκριμένο είδος αναδρομικού νευρνικού δικτύου, το οποίο είναι ικανό να ανταποκριθεί στην μάθηση χρονικών εξαρτήσεων μεγάλης κλίμακας. Εφευρέθηκαν από τους Hochreiter και Schmidhuber το 1997 και δέχθηκαν βελτιώσεις από πολλούς ερευνητές στο μέλλον. Παρουσιάζουν αξιοπρόσεκτη απόδοση σε πολλών ειδών εφαρμογές και προβλήματα και πλέον χρησιμοποιούνται κατά κόρων από τους ερευνητές βαθειάς μηχανικής μάθησης.

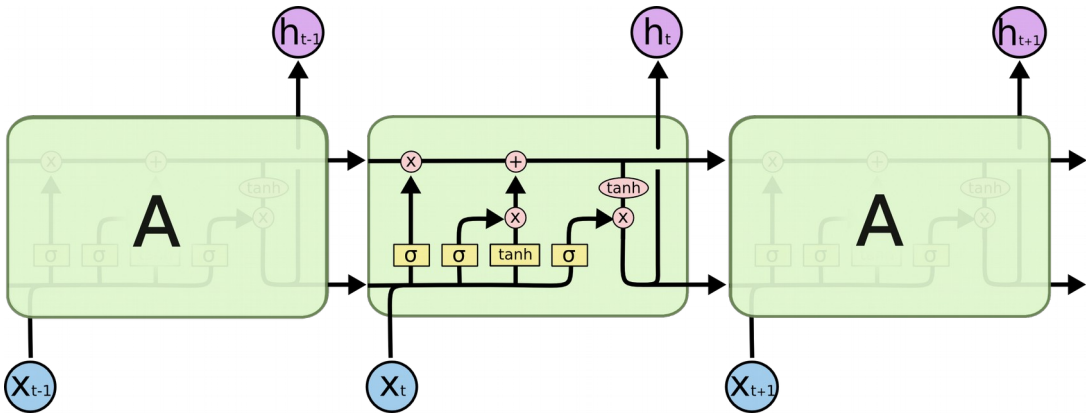
Τα LSTM νευρωνικά δίκτυα είναι σχεδιασμένα για αποφεύγουν τα προβλήματα που δημιουργούν οι μεγάλης κλίμακας χρονικές εξαρτήσεις. Για την ακρίβεια, το να θυμούνται πληροφορίες για μεγάλο χρονικό διάστημα είναι το βασικό τους χαρακτηριστικό και όχι κάτι το οποίο δυσκολεύει τη λειτουργία τους.

Όλα τα αναδρομικά νευρωνικά δίκτυα όταν ξεδιπλωθούν (unfold) έχουν τη μορφή μία αλυσίδας επαναλαμβανόμενων κόμβων, όπου κάθε κόμβος αντιπροσωπεύει ένα στοιχείο του νευρωνικού

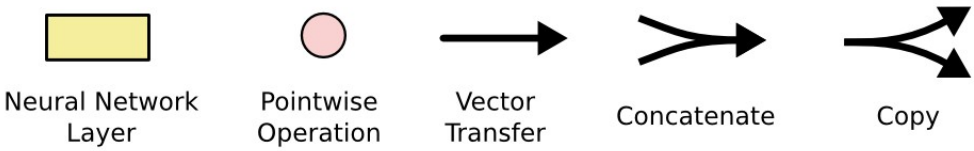
δικτύου. Τα κλασσικά αναδρομικά νευρωνικά δίκτυα κάθε κελί (cell) έχει απλή δομή, όπως είδαμε και σε προηγούμενη ενότητα.



Τα LSTM δίκτυα έχουν επίσης την ίδια μορφή αλυσίδας όμως τα cells του δικτύου έχουν διαφορετική αρχιτεκτονική από αυτή του απλού αναδρομικού νευρωνικού δικτύου. Αντί να υπάρχει μόνο ένα επίπεδο υπάρχουν τέσσερα, τα οποία αλληλεπιδρουν μεταξύ τους με πολύ συγκεκριμένο τρόπο έτσι ώστε να υλοποιούν την λειτουργία του LSTM.

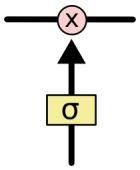


Στο παραπάνω σχήμα φαίνεται η δομή ενός LSTM cell. Τα σύμβολα που εμφανίζονται αντιπροσωπεύουν τις παρακάτω μαθηματικές πράξεις:



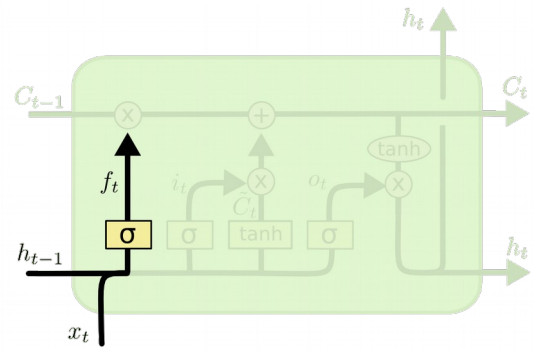
Στο παραπάνω διάγραμμα, κάθε γραμμή συμβολίζει την κίνηση των διανυσμάτων στο δίκτυο. Το βασικό χαρακτηριστικό των LSTM δικτύων που καθορίζει τη λειτουργία τους είναι το cell state, δηλαδή η οριζόντια γραμμή στο πάνω μέρος του διαγράμματος του LSTM. Το cell state μεταφέρει πληροφορία από τον ένα κόμβο της αλυσίδας στον επόμενο χωρίς η πληροφορία αυτή να αλλάζει σε μεγάλο βαθμό. Αντίθετα υποβάλλεται σε μερικές μικρές γραμμικές μεταβολές. Το κύτταρο του LSTM έχει τη δυνατότητα να προσθέτει ή να αφαιρεί πληροφορία στο cell state μέσω των πυλών του. Οι πύλες αυτές ουσιαστικά καθορίζουν ποιες πληροφορίες θα περάσουν προς το cell state και

ποιες όχι. Οι πύλες του LSTM αποτελούνται από μια σιγμοειδή συνάρτηση ενεργοποίησης και έναν πολλαπλασιαστή pointwise multiplication.



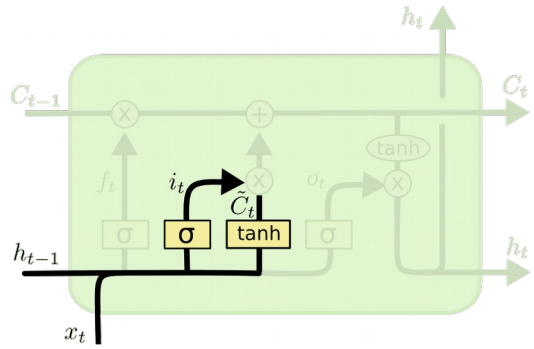
Η σιγμοειδής συνάρτηση ενεργοποίησης συμπιέζει την πληροφορία που δέχεται μεταξύ του 0 και του 1, και ουσιαστικά καθορίζει το ποσοστό της πληροφορίας που θα περάσει από την πύλη προς το cell state. Αν η τιμή είναι μηδέν αυτό σημαίνει ότι δεν περνάει καθόλου πληροφορία από την πύλη ενώ η τιμή είναι ένα σημαίνει ότι η πληροφορία περνάει ολόκληρη από την πύλη. Ένα LSTM έχει 3 τέτοιες πύλες, την πύλη εισόδου (input gate), την πύλη λήθης (forget gate) και την πύλη εξόδου (output gate).

Σε αυτό το σημείο θα εξηγήσουμε ακριβώς τις πράξεις που συμβαίνουν σε ένα κύτταρο ενός LSTM. Η πρώτη ενέργεια ενός LSTM δικτύου είναι να αποφασίσει ποιες από τις εισερχόμενες πληροφορίες θα πετάξει και δεν θα τις αφήσει να περάσουν στο cell state του. Η απόφαση αυτή λαμβάνεται από ένα σιγμοειδές φίλτρο, την πύλη λήθης. Συγκεκριμένα η πύλη αυτή δέχεται σαν είσοδο τα $\boldsymbol{h_{t-1}}$ (κρυφή κατάσταση / hidden state της προηγούμενης χρονικής στιγμής) και $\boldsymbol{x_t}$ (είσοδος αυτή τη χρονική στιγμή) και δίνει σαν έξοδο έναν αριθμό ανάμεσα στο 0 και το 1 για κάθε αριθμό του cell state $\boldsymbol{c_{t-1}}$ που εκφράζει το ποσοστό της εισόδου της πύλης που περνάει στο cell state.



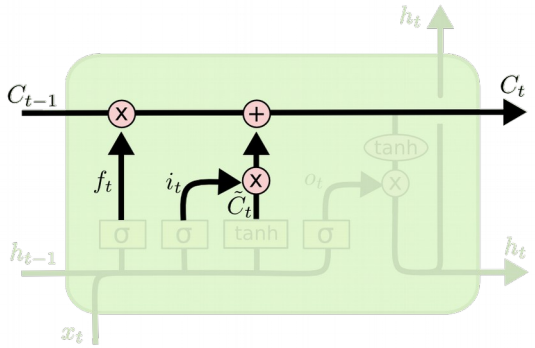
$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \ + \ b_f\right)$$

Το επόμενο βήμα είναι να αποφασίσει ποιες από τις καινούριες πληροφορίες θα περάσουν και θα αποθηκευτούν στο cell state. Το στάδιο αυτό χωρίζεται σε δύο μέρη. Το πρώτο είναι ένα σιγμοειδές φίλτρο που καθρίζει ποιες τιμές του cell state θα ενημερωθούν. Στη συνέχεια ένα *tanh* φίλτρο δημιουργεί κάποιες τιμές $\boldsymbol{C_t}$, οι οποίες είναι υποψήφιες να προστεθούν στο διάνυσμα κατάστασης. Οι έξοδοι των δύο αυτών φίλτρων προστίθενται μεταξύ τους.



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

Στο επόμενο στάδιο συμβαίνει η ενημέρωση του διανύσματος κατάστασης. Αρχικά πολλαπλασιάζεται το παλιό διάνυσμα κατάστασης C_{t-1} με την έξοδο της πύλης λήθης f_t, “ξεχνώντας” τις πληροφορίες που πρέπει. Στη συνέχεια προστίθεται η τιμή $i_t * C_t$. Αυτές είναι οι υποψήφιες τιμές πολλαπλασιασμένες με ένα scale factor i_t.

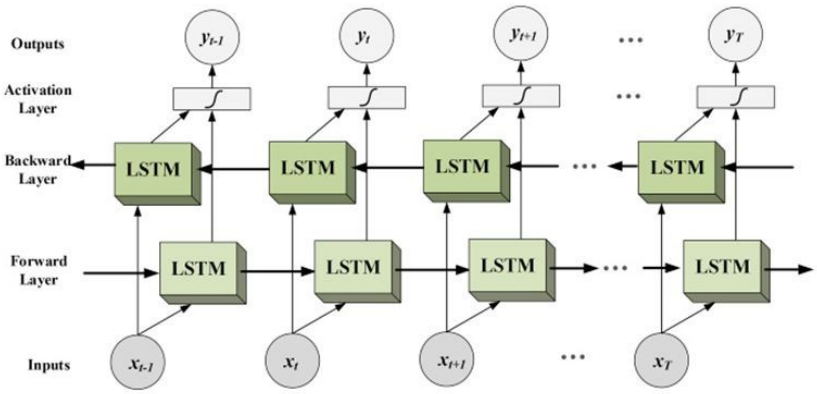


$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Τέλος, το LSTM αποφασίζει τι έξοδο θα δώσει. Η έξοδος του LSTM καθορίζεται από το διάνυσμα κατάστασης αλλά φιλτραρισμένο. Αρχικά ένα σιγμοειδές φίλτρο καθορίζει ποιες τιμές του διανύσματος κατάστασης θα βγουν στην έξοδο του LSTM. Στη συνέχεια το διάνυσμα κατάστασης περνάει από ένα φίλτρο υπερβολικής εφαπτομένης για να μετατραπεί το εύρος των τιμών του σε (-1,1) και το πολλαπλασιάζεται με την έξοδο του σιγμοειδούς φίλτρου για να προκύψει τελικά η συνολική έξοδος.

LSTM διπλής κατεύθυνσης

Τα κλασσικά LSTM, όπως αναφέρθηκε εκτενώς στην προηγούμενη ενότητα, κάνουν προβλέψεις λαμβάνοντας υπόψη την πληροφορία που έχουν λάβε από τις προηγούμενες χρονικές στιγμές. Η ιδέα πίσω από τα LSTM διπλής κατεύθυνσης είναι πολλή απλή και έχει ως στόχο να αντιμετωπίσει τον συγκεκριμένο κατασκευαστικό περιορισμό των LSTM. Τα LSTM διπλής κατεύθυνσης προκύπτουν ουσιαστικά αν διπλασιάσουμε το LSTM αλλά τροφοδοτήσουμε το καινούριο LSTM την είσοδο που τροφοδοτούμε στο αρχικό με την αντίστροφη σειρά. Στη συνέχεια οι έξοδοι των δύο δικτύων ενώνονται σε μία έξοδο και έτσι προκύπτει η συνολική έξοδος του LSTM διπλής κατεύθυνσης.

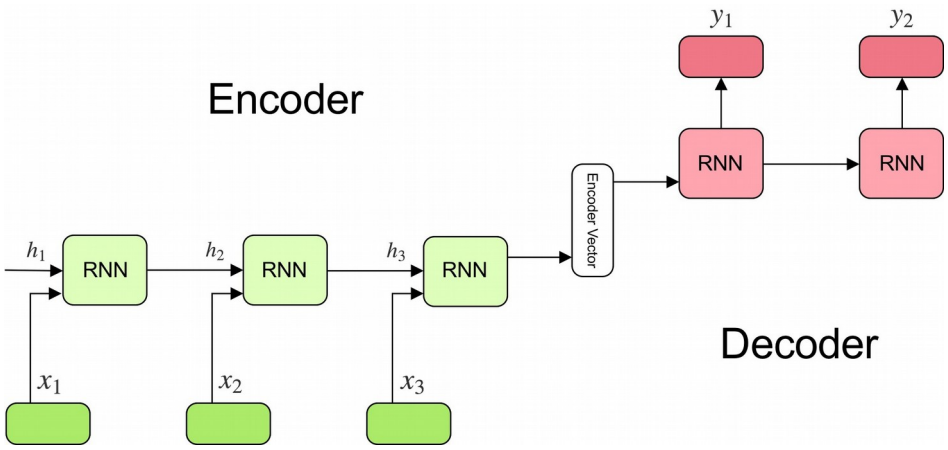


Η χρησιμότητα των LSTM διπλής κατεύθυνσης είναι προφανής. Συγκεκριμένα στην περίπτωση της εναρμόνισης μελωδίας, η ανάγνωση της ακολουθίας νοτών της μελωδίας και από τις δύο πλευρές έχει πρακτική σημασία. Αυτό συμβαίνει γιατί κατά τη διαδικασία εναρμόνισης μίας μελωδίας, ο συνθέτης δεν επιλέγει τις συγχορδίες που θα πλαισιώσουν τη μελωδία με τη σειρά αλλά αντιθέτως μπορεί να επιλέξει μία προγενέστερη συγχορδία με βάση μία μεταγενέστερη.

2.2.3 Sequence to Sequence αρχιτεκτονική

Η αρχιτεκτονική sequence to sequence (ακολουθία σε ακολουθία) είναι μία πολύ διαδεδομένη αρχιτεκτονική νευρωνικών δικτύων που σχετίζονται με ακολουθίες. Έκανε την εμφάνιση της για πρώτη φορά το 2014 και από τότε χρησιμοποιείται σε πληθώρα εφαρμογών όπως αυτόματες μεταφράσεις κειμένων (Neural Machine Translation), αναγνώριση λόγου (speech recognition) κ.α.. Η αρχιτεκτονική αυτή, όπως φανερώνεται και από την ονομασία της, στοχεύει στο να συσχετίζει μία ακολουθία εσόδου με μία ακολουθία εξόδου, όπου το μήκος των δύο ακολουθιών μπορεί να είναι διαφορετικό. Εκτός από την ονομασία sqeuence to sequence η συγκεκριμένη αρχιτεκτονική είναι γνωστή και ως αρχιτεκτονική encoder – decoder (κωδικοποιητής – αποκωδικοποιητή) λόγω των 2 αναδρομικών δικτύων που διαθέτει και που λειτουργούν σαν κωδικοποιητής και αποκωδικοποιητής αντίστοιχα. Στη συνέχεια θα περιγράψουμε αναλυτικότερα την συγκεκριμένη αρχιτεκτονική.

Η αρχιτεκτονική sequence to sequence, όπως φαίνεται και στο παρακάτω διάγραμμα, αποτελείται από τρία μέρη, τον κωδικοποιητή (encoder), το κωδικοποιμένο διάνυσμα (encoded vector) και τον αποκωδικοποιητή (decoder).



Κωδικοποιητής / Encoder:

Ο κωδικοποιητής του δικτύου αποτελείται από από μία αλυσίδα αναδρομικών στοιχείων (στην πλειονότητα των περιπτώσεων χρησιμοποιούνται LSTMs ή GRUs λόγω της φύσης των προβλημάτων στα οποία χρησιμοποιείται η συγκεκριμένη αρχιτεκτονική), όπου κάθε κόμβος της αλυσίδας δέχεται σαν είσοδο ένα στοιχείο της ακολουθίας εισόδου, και ενημερώνει τα states του και μεταφέρει την πληροφορία στον επόμενο κόμβο της αλυσίδας. Το διάνυσμα κρυφής κατάστασης κάθε μονάδας LSTM ενημερώνεται σύμφωνα με τον παρακάτω μαθηματικό τύπο:

$$h_t = f(W^{(hh)}h_{t-1} + W^{(hx)}x_t)$$

Ο παραπάνω τύπος εκφράζει το μαθηματικό μετασχηματισμό που πραγματοποιείται σε ένα απλό αναδρομικό νευρωνικό δίκτυο. Ουσιαστικά απλά γίνεται πολλαπλασιασμός των πινάκων βαρών με το προηγούμενο κρυφό διάνυσμαα κατάστασης $\boldsymbol{h_{t-1}}$ και με το διάνυσμα εισόδου αυτή τη χρονική στιγμή $\boldsymbol{x_t}$.

Κωδικοποιημένο διάνυσμα / Encoded Vector

Το κωδικοποιημένο διάνυσμα $\boldsymbol{h_t}$ είναι ουσιαστικά η έξοδος του κωδικοποιητή του μοντέλου. Το διάνυσμα αυτό είναι το διάνυσμα κρυφής κατάστασης του αναδρομικής μονάδας μετά την είσοδο του τελευταίου όρου της ακολουθίας εισόδου. Το διάνυσμα κρυφής κατάστασης ουσιαστικά περιέχει πληροφορίες από όλες τις εισόδους που έχει δεχτεί το νευρωνικό δίκτυο μέχρι εκείνη τη στιγμή. Άρα το τελικό διάνυσμα κρυφής κατάστασης $\boldsymbol{h_t}$ εμπεριέχει όλη την απαραίτητη πληροφορία από την είσοδο κωδικοποιημένη. Το διάνυσμα αυτό γεφυρώνει τον κωδικοποιητή με τον αποκωδικοποιητή καθώς περνάει στον αποκωδικοποιητή ως η αρχική κρυφή κατάσταση (initial hidden state) του αποκωδικοποιητή.

Αποκωδικοποιητής / Decoder

Όπως ο κωδικοποιητής, έτσι ακριβώς και ο αποκωδικοποιητής αποτελείται από ένα αναδρομικό νευρωνικό δίκτυο. Κάθε κόμβος του δικτύου αυτού κάνει μία πρόβλεψη που αντιστοιχεί σε κάθε χρονική στιγμή. Κάθε recurrent unit δέχεται το διάνυσμα κρυφής κατάστασης από το το προηγούμενο recurrent unit αλλά σε αντίθεση με τον κωδικοποιητή που απλά έστελνε το δικό του hidden state στο επόμενο unit, εδώ δίνει το hidden state του και σαν έξοδο. Τα διανύσματα κρυφής κατάστασης υπολογίζονται από τον παρκάτω τύπο:

$$h_t = f(W^{(hh)}h_{t-1})$$

Η έξοδος του μοντέλου συνήθως προκύπτει μετά την εφαρμογή μιας συνάρτησης ενεργοποίησης στις εξόδους του αποκωδικοποιητή, για παράδειγμα:

$$y_t = softmax(W^S h_t)$$

Η μεγάλη χρησιμότητα αυτού του μοντέλου έγκειται στο γεγονός ότι μπορεί να συσχετίσει ακολουθίες εισόδου και εξόδου με διαφορετικό μήκος. Όπως φάνηκε και στην παραπάνω ανάλυση, το μοντέλο δεν παράγει μία έξοδο για κάθε είσοδο που του δίνεται αλλά αντίθετα κωδικοποιεί την είσοδο σε ένα διάνυσμα και στη συνέχεια αποκωδικοποιεί αυτό το διάνυσμα παράγοντας την έξοδο χωρίς να συγχέει το μήκος της ακολουθίας εισόδου με αυτό της ακολουθίας εξόδου.

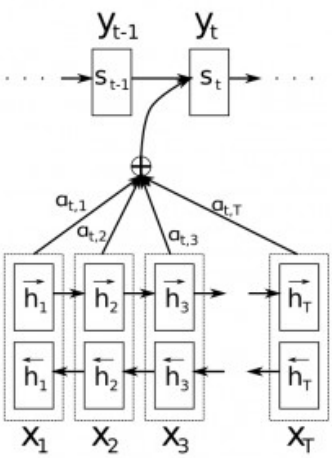
Μηχανισμός προσοχής

Όπως είδαμε στην προηγούμενη ενότητα, η αρχιτεκτονική κωδικοποιητή – αποκωδικοποιητή εξειδικεύεται στο να συσχετίζει ακολουθίες με πιθανότατα διαφορετικό μήκος. Στην συγκεκριμένη

αρχιτεκτονική, η ακολουθία εισόδου κωδικοποιείται σε ένα σταθερού μεγέθους διάνυσμα και στη συνέχεια παράγεται η ακολουθία εξόδου σαν αποτέλεσμα της αποκωδικοποίησης αυτού του διανύσματος. Η συγκεκριμένη μεθοδολογία παρουσιάζει το εξής πλεονέκτημα. Το δίκτυο μαθαίνει τη συνολική συμπεριφορά της ακολουθίας. Αυτό είναι ιδιαίτερα χρήσιμο σε αρκετές εφαρμογές όπως για παράδειγμα η μετάφραση κειμένων.

Σύμφωνα με τα προηγούμενα, ο αποκωδικοποιητής πρέπει να παράγει την έξοδο του δικτύου βασισμένος σε ένα και μόνο διάνυσμα, το τελικό διάνυσμα κρυφής κατάστασης του κωδικοποιητή. Το διάνυσμα αυτό υποτίθεται ότι θα πρέπει να εμπεριέχει όλες τις απαραίτητες πληροφορίες που θα χρειαστεί ο αποκωδικοποιητής προκειμένου να παράξει την έξοδο του μοντέλου. Παρ' όλα αυτά, ακούγεται λίγο δύσκολο και παράλογο να πρέπει να μπορέσουμε να κωδικοποιήσουμε όλες τις πληροφορίες μιας ακολουθίας σε ένα και μόνο διάνυσμα. Αυτό γίνεται ακόμα δυσκολότερο όσο μεγαλύτερο είναι το μήκος της ακολουθίας εισόδου. Υποτίθεται πως για αυτόν ακριβώς το λόγο δημιουργήθηκαν τα LSTM και τα GRU δίκτυα που έχουν τη δυνατότητα να "ξεχνούν" πληροφορίες οι οποίες αποδεικνύεται ότι δεν συμβάλουν στην αύξηση της αποδοτικότητας του δικτύου. Όμως στην πράξη οι ακολουθίες μεγάλου μεγέθους παρουσιάζουν προβλήματα ακόμα και όταν χρησιμοποιούνται προχωρημένα αναδρομικά δίκτυα. Το πρόβλημα που παρατηρείται είναι ότι οι πληροφορίες των αρχικών στιγμιοτύπων της ακολουθίας δεν συμβάλουν το ίδιο με αυτές των τελευταίων στιγμιοτύπων. Μία λύση για αυτό το πρόβλημα είναι η χρήση αναδρομικών δικτύων διπλής κατεύθυνσης αλλά δεν αντιμετωπίζει το πρόβλημα καθολικά, καθώς απλά παρουσιάζει ικανοποιητικά αποτελέσματα σε λίγο μεγαλύτερου μήκους ακολουθίες. Μια πραγματική λύση θα έπρεπε να είναι αποτελεσματική ανεξαρτήτως μήκους ακολουθίας εισόδου.

Αυτό ακριβώς επιτυγχάνει ο μηχανισμός προσοχής (attention mechanism). Με το μηχανισμό προσοχής ο στόχος δεν είναι να να κωδικοποιήσουμε ολόκληρη την ακολουθία εισόδου σε ένα διάνυσμα σταθερού μεγέθους. Αντίθετα, δίνεται η δυνατότητα στον αποκωδικοποιητή να δώσει προσοχή σε κάποιο διαφορετικά μέρη της ακολουθίας εισόδου σε κάθε χρονική στιγμή της παραγωγής της ακολουθίας εξόδου. Ουσιαστικά κατά την εκπαίδευση ολόκληρου του μοντέλου, το μοντέλο "μαθαίνει" που να δώσει προσοχή σύμφωνα με την ακολουθία εισόδου και το τι έχει παράξει μέχρι τη συγκεκριμένη χρονική στιγμή.



Στο παραπάνω διάγραμμα απεικονίζεται η λειτουργία του μηχανισμού προσοχής. Τα $\boldsymbol{y_t}$ είναι η έξοδοι του αποκωδικοποιητή για κάθε στιγμιότυπο. Τα $\boldsymbol{x_t}$ είναι τα στιγμιότυπα της ακολουθίας εισόδου. Η παραπάνω υλοποίηση χρησιμοποιεί ένα αναδρομικό νευρωνικό δίκτυο διπλής κατεύθυνσης αλλά αυτό δεν είναι απαραίτητο. Αυτό που διαφέρει από ένα απλό μοντέλο

κωδικοποιητή – αποκωδικοποιητή είναι το γεγονός ότι η κάθε μονάδα του αποκωδικοποιητή εξαρτάται από έναν βεβαρυμένο συνδυασμό όλων των διανυσμάτων κρυφής κατάστασης του κωδικοποιητή και όχι μόνο από το τελευταίο. Τα $\boldsymbol{\alpha_t}$ είναι τα βάρη που δηλώνουν πόση προσοχή δείχνει η κάθε μονάδα του αποκωδικοποιητή σε κάθε διάνυσμα κρυφής κατάστασης του κωδικοποιητή. Τα βάρη αυτά για περισσότερη ευχρηστία κανονικοποιούνται ώστε να δίνουν άθροισμα 1.

Υπάρχουν διάφορες υλοποιήσεις του μηχανισμού προσοχής. Η πιο διαδεδομένη μορφή του είναι η Bahdanau et al. 2015. Στη συγκεκριμένη διπλωματική εργασία χρησιμοποιήσαμε μία παραλλαγή της συγκεκριμένης υλοποίησης από τους Luong et al. 2015 λόγω της δυνατότητας υλοποίησης της με τη χρήση ήδη υπαρχόντων layers από το Keras. Υπάρχουν δύο είδη μηχανισμών προσοχής, ο μηχανισμός καθολικής (global attention) και τοπικής (local attention) προσοχής. Στην πρώτη περίπτωση ο μηχανισμός προσοχής εφαρμόζεται σε ολόκληρο το σύνολο δεδομένων που τροφοδοτούνται στο δίκτυο ενώ στη δεύτερη περίπτωση ο μηχανισμός εφαρμόζεται σε κάποιο υποσύνολο των δεδομένων αυτών. Σε αυτή τη διπλωματική εργασία χρησιμοποιήθηκε ο μηχανισμός καθολικής προσοχής.

Η ιδέα πίσω από τον μηχανισμό καθολικής προσοχής είναι το μοντέλο να λαμβάνει υπόψη όλα τα hidden states του κωδικοποιητή όταν δημιουργεί το context vector. Για το λόγο αυτό δημιουργούμε ένα διάνυσμα $\boldsymbol{\alpha_t}$ για κάθε timestep t της αποκωδικοποίησης εφαρμόζοντας μια συνάρτηση dot μεταξύ του hidden state του αποκωδικοποιητή $\boldsymbol{h_t}$ και κάθε hidden state του κωδικοποιητή $\boldsymbol{h_s}$ και μιας συνάρτησης *softmax:*

$$\boldsymbol{a_t}(s) = align(\boldsymbol{h_t}, \boldsymbol{h_s}) = softmax(\boldsymbol{h_t} \times \boldsymbol{h_s})$$

όπου $\boldsymbol{h_t}, \boldsymbol{h_s}$ τα διανύσματα κρυφής κατάστασης αποκωδικοποιητή και κωδικοποιητή αντίστοιχα. Στη συνέχεια το context vector $\boldsymbol{c_t}$ προκύπτει σαν ο σταθμισμένος μέσος όλων των διανυσμάτων κρυφής κατάστασης του κωδικοποιητή.

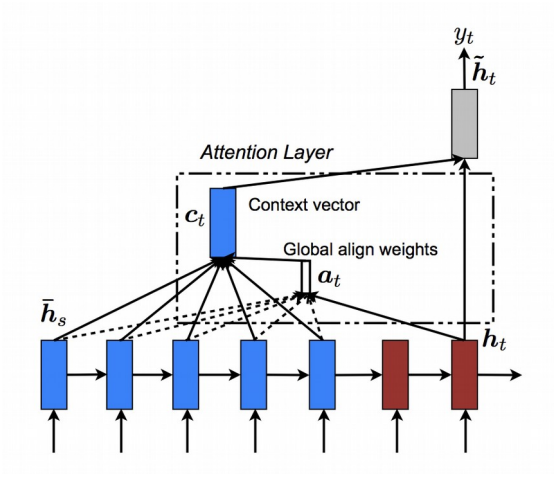
$$\boldsymbol{c_t} = \boldsymbol{h_s} \times \boldsymbol{a_t}$$

Στη συνέχεια, έχοντας το διάνυσμα κρυφής κατάστασης του αποκωδικοποιητή $\boldsymbol{h_t}$ και το διάνυσμα περιεχομένου $\boldsymbol{c_t}$ με την ένωση των δύο διανυσμάτων και την εφαρμογή μιας συνάρτησης ενεργοποίησης υπερβολικής εφαπτομένης παράγουμε το hidden state του μηχανισμού προσοχής ως εξής:

$$\boldsymbol{h_t} = \tanh(\boldsymbol{W_c}[\boldsymbol{c_t}; \boldsymbol{h_t}])$$

Τέλος, παράγουμε την κατανομή των πιθανοτήτων κάθε κλάσης με την προσθήκη ενός επιπέδου ενεργοποίησης που εφαρμόζει τη συνάρτηση *softmax*, ως εξής:

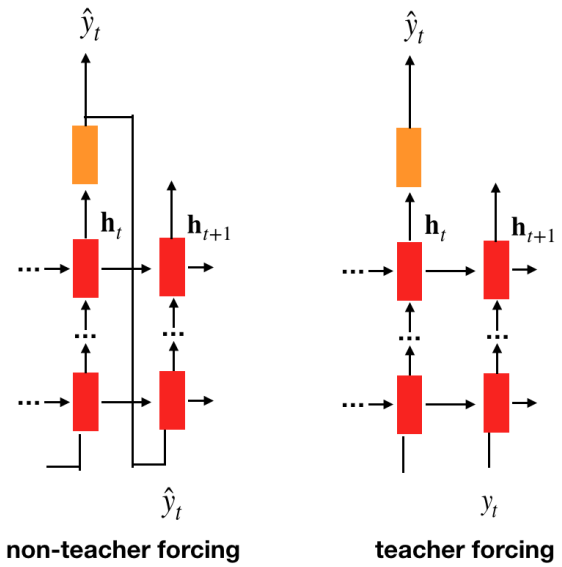
$$p(y_t | y_{\tau<t}, x) = softmax(\boldsymbol{W_s} \boldsymbol{h_t})$$



Ο μηχανισμός προσοχής είναι πολύ χρήσιμος και χρησιμοποιείται ευρεία σε πληθώρα εφαρμογών σε βαθμό που τείνει να θεωρείται αναπόσπαστο μέρος ενός μοντέλου κωδικοποιητή – αποκωδικοποιητή, έχει κάποιο σημαντικό υπολογιστικό κόστος. Αυτό διαφέρει ριζικά με την "ανθρώπινη προσοχή". Για έναν άνθρωπο, το να επικεντρώνεται σε ένα πράγμα και να αγνοεί άλλα ουσιαστικά τον κάνει να "εξοικονομεί μνήμη" και "υπολογιστικούς πόρους". Αντίθετα, με τον μηχανισμό προσοχής στα νευρωνικά δίκτυα, το μοντέλο ελέγχει λεπτομερώς όλα τα διανύσματα κρυφής κατάστασης πριν επιλέξει που θα "δώσει προσοχή". Αυτό απαιτεί πολλή περισσότερη μνήμη και υπολογιστικούς πόρους.

Μέθοδος Teacher Forcing

Η μέθοδος Teacher Forcing χρησιμοποιείται για την εκπαίδευση μοντέλων κωδικοποιητή – αποκωδικοποιητή. Τελευταία μάλιστα, η συγκεκριμένη μέθοδος εκπαίδευσης γίνεται όλο και πιο δημοφιλής. Η ιδέα πίσω από την μέθοδο αυτή είναι πολύ απλή. Τα μοντέλα κωδικοποιητή – αποκωδικοποιητή εξειδικεύονται στην παραγωγή μίας ακολουθίας εξόδου δεδομένης μίας ακολουθίας εισόδου. Κατά τη διάρκεια της εκπαίδευσης, αφού η είσοδος του κωδικοποιητή έχει κωδικοποιηθεί σε ένα διάνυσμα που συνοψίζει το περιεχόμενο της εισόδου (context vector), ο κωδικοποιητής τροφοδοτείται με την ακολουθία εξόδου που έχει παράξει μέχρι τη συγκεκριμένη χρονική στιγμή *(t-1)* και κάνει μία πρόβλεψη για την επόμενη χρονική στιγμή *(t)*. Όμως αυτή η μεθοδολογία παρουσιάζει ένα βασικό μειονέκτημα. Αν ο αποκωδικοποιητής κάνει μία λανθασμένη πρόβλεψη, στη συνέχεια κάθε πρόβλεψη που θα κάνει θα εξαρτάται και από αυτή τη λανθασμένη πρόβλεψη. Αυτό έχει ως αποτέλεσμα, ακόμα και αν το μοντέλο μαθαίνει κανονικά και κάνει κατά κύριο λόγο σωστές προβλέψεις, μία λάθος πρόβλεψη θα επηρεάσει αρνητικά τις επόμενες προβλέψεις και να ρίχνει την ακρίβεια του μοντέλου, παρ' όλο που αυτό "μαθαίνει" κανονικά. Αυτό το πρόβλημα αντιμετωπίζει αποτελεσματικά η μέθοδος του teacher forcing. Πιο συγκεκριμένα, ο αποκωδικοποιητής κάνει την πρόβλεψη του τη χρονική στιγμή *t* δεχόμενος σαν είσοδο τη σωστή ακολουθία για τις χρονικές στιγμές πριν τη συγκεκριμένη χρονική στιγμή *(< t)*. Με τον τρόπο αυτό κάνει την πρόβλεψή του βάση της σωστής ακολουθίας χωρίς να επηρεάζεται αρνητικά από προηγούμενες λανθασμένες προβλέψεις. Αυτό επιταχύνει σημαντικά τη διαδικασία εκπαίδευσης του δικτύου.



2.3. Συνελικτικά Νευρωνικά Δίκτυα

2.3.1. Γιατί εναρμόνιση μελωδίας με συνελικτικά νευρωνικά δίκτυα

Το συνελικτικό νευρωνικό δίκτυο (Convolutional Neural Network, CNN) αποτελεί ένα είδος νευρωνικού δικτύου που χρησιμοποιείται ευρέως σε ποικίλες εφαρμογές που έχουν να κάνουν με δεδομένα που παρουσιάζουν χωρική εξάρτηση μεταξύ τους. Για το λόγο αυτό χρησιμοποιούνται κατά κόρον σε εφαρμογές όπως αναγνώριση εικόνων και βίντεο και κατηγοριοποίηση εικόνων καθώς οι εικόνες μπορούν να αναπαρασταθούν πολύ φυσιολογικά σε δισδιάστατους (ή τρισδιάστατους) πίνακες, όπου η τιμή του κάθε στοιχείου του πίνακα αντιστοιχεί στην τιμή του αντίστοιχου pixel.

Τα συνελικτικά νευρωνικά δίκτυα μπορούν να αντιληφθούν τις χωρικές και χρονικές εξαρτήσεις. Αυτό τα καθιστά ιδιαίτερα αποτελεσματικά και σε εφαρμογές σύνθεσης μουσικής. Παλιότερα, η επικρατούσα τάση στο πεδίο των νευρωνικών δικτύων είναι να αναπαρίσταται ένα μουσικό κομμάτι στη μορφή ακολουθιών νοτών. Όμως μία νότα ενός κομματιού μπορεί να εξαρτάται όχι μόνο από τις νότες που παίχθηκαν πριν από αυτή αλλά και από αυτές που παίζονται ταυτόχρονα ή θα παιχθούν μετά από αυτή. Επίσης, η επιρροή που έχει μία νότα πάνω σε κάποια άλλη είναι μεγαλύτερη αν αυτές οι νότες είναι χρονικά κοντά η μία στην άλλη και μικρότερη όσο μεγαλύτερη είναι η χρονική τους απόσταση. Τα συνελικτικά νευρωνικά δίκτυα μπορούν να αποδώσουν καλύτερα από τα αναδρομικά ή άλλα νευρωνικά δίκτυα αυτή τη σχέση εξάρτησης μίας νότας με της γειτονικές της νότες, όπως ακριβώς κάνει και με τα pixels μίας εικόνας, με τη διαφορά ότι αντί για μία εικόνα δέχεται σαν είσοδο ένα πίνακα που προσομοιώνει ένα επίπεδο με άξονες τον χρονικό και τονικό (pitch) άξονα.

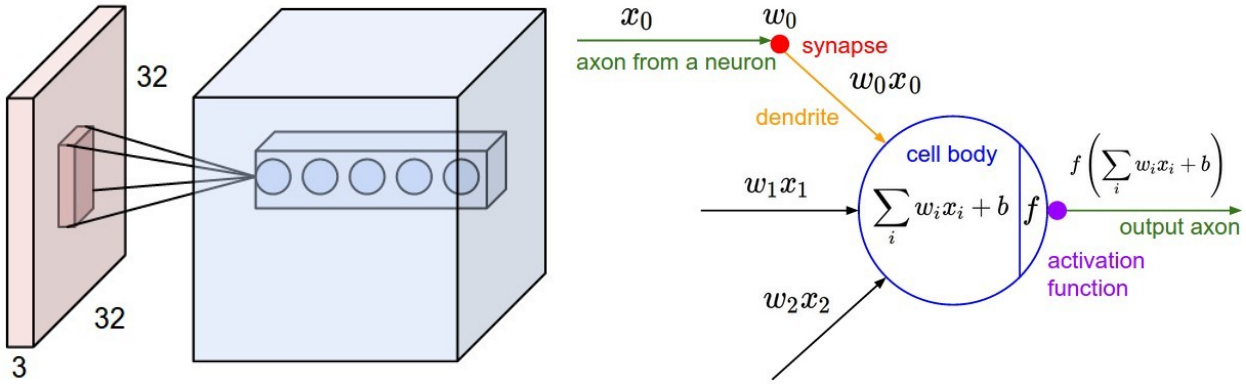
2.3.2. Επίπεδα συνελικτικού νευρωνικού δικτύου

Στο σημείο αυτό θα αναφερθούν μερικές βασικές πληροφορίες για τα επί μέρους επίπεδα που μπορεί να διαθέτει ένα συνελικτικό νευρωνικό δίκτυο και τις λειτουργίες που μπορεί να έχει το καθένα.

Συνελικτικό επίπεδο – Πυρήνας

Το συνελικτικό επίπεδο είναι το πιο βασικό επίπεδο του δικτύου. Αξίζει να σημειωθεί ότι η ονομασία του μπορεί να είναι κάπως παραπλανητική , καθώς δεν αντικατοπτρίζει ακριβώς τη λειτουργία που επιτελεί. Συγκεκριμένα, το συνελικτικό επίπεδο παράγει την έξοδό του εφαρμόζοντας κάποιου είδους dot – multiplication μεταξύ του πίνακα εισόδου και του συνελικτικού πυρήνα. Η απόσταση του προηγούμενου με το επόμενο στοιχείο του πίνακα εισόδου που επισκέπτεται ο συνελικτικός πυρήνας λέγεται βήμα (stride). Όταν το βήμα είναι ίσο με 1 τότε ο πυρήνας επισκέπτεται όλα τα στοιχεία του πίνακα εισόδου και εφαρμόζει τη συνελικτική πράξη.

Το συνελικτικό επίπεδο αποτελείται από ένα set από εκπαιδεύσιμα φίλτρα. Κάθε φίλτρο έχει βάθος ίσο με το βάθος του διανύσματος εισόδου και κάποιο μικρό μέγεθος στους άλλους δύο άξονες (ύψος και πλάτος). Η λειτουργία του συνελικτικού επιπέδου είναι η εξής. Τα φίλτρα αυτά κινούνται κατά τον άξονα του πλάτους και του ύψους και υπολογίζουν ένα dot product μεταξύ του φίλτρου και το διανύσματος εισόδου σε κάθε θέση. Καθώς το κάθε φίλτρο κινείται κατά πλάτος και ύψος τους διανύσματος εισόδου παράγεται ένα διάνυσμα δύο διαστάσεων. Το τελικό αποτέλεσμα είναι η δημιουργία τόσων τέτοιων διανυσμάτων όσος και ο αριθμός των φίλτρων. Τα διανύσματα αυτά στη συνέχεια στοιβάζονται δημιουργώντας το διάνυσμα εξόδου του συνελικτικού επιπέδου.



Κάθε στοιχείο των φίλτρων του συνελικτικού επιπέδου αποτελεί ένα σύνολο εκπαιδεύσιμων παραμέτρων. Ο αριθμός των παραμέτρων ενός επιπέδου δίνεται από την παρακάτω σχέση:

$$p = k * k_s * k_s * i_d$$

όπου k ο αριθμός των φίλτρων, k_s το μέγεθος του κάθε φίλτρου και i_d η διάσταση βάθους της εισόδου.

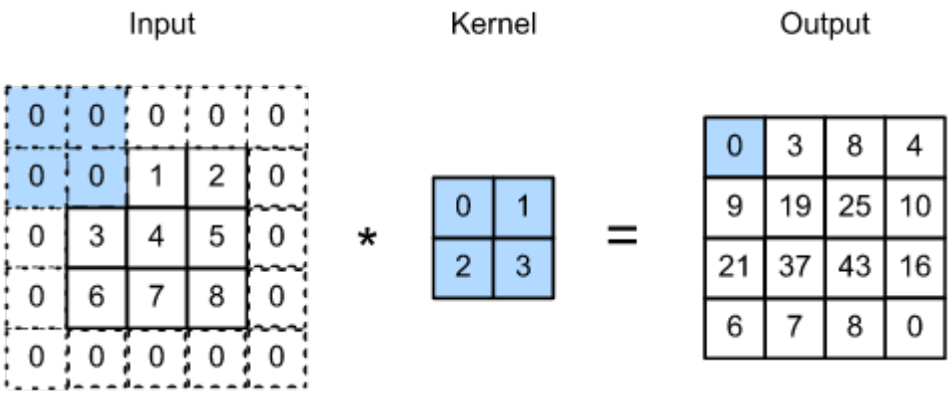
Συνυπολογίζοντας και το bias, το οποίοι προσθέτει μία επιπλέον παράμετρο σε κάθε στοιχείο του κάθε φίλτρου ο συνολικός αριθμός των παραμέτρων ενός συνελικτικού επιπέδου ανέρχεται σε:

$$p = k * k_s * k_s * i_d + k$$

Βασικό χαρακτηριστικό των συνελικτικών νευρωνικών δικτύων είναι η μεταβολή των διαστάσεων των διανυσμάτων μετά το πέρασμά τους από κάθε συνελικτικό φίλτρο. Το μέγεθος της εξόδου ενός συνελικτικού φίλτρου εξαρτάται από ένα σύνολο υπερπαραμέτρων. Αυτές είναι ο πλήθος των φίλτρων του συνελικτικού επιπέδου, το βήμα συνέλιξης (stride) και το padding. Πιο συγκεκριμένα:

- Το πλήθος των φίλτρων κάθε επιπέδου καθορίζει το βάθος του διανύσματος εξόδου. Κάθε φίλτρο του επιπέδου "μαθαίνει" ένα διαφορετικό χαρακτηριστικό της εισόδου.

- Μερικές φορές είναι χρήσιμο να περικλείουμε κάθε επίπεδο του διανύσματος εισόδου με μηδενικά στοιχεία. Η χρησιμότητα της συγκεκριμένης κίνησης έγκειται στο γεγονός ότι με αυτόν τον τρόπο μπορούμε να διατηρήσουμε (ή να μεταβάλουμε κατά το δοκούν) το μέγεθος του επιπέδου του διανύσματος εξόδου.

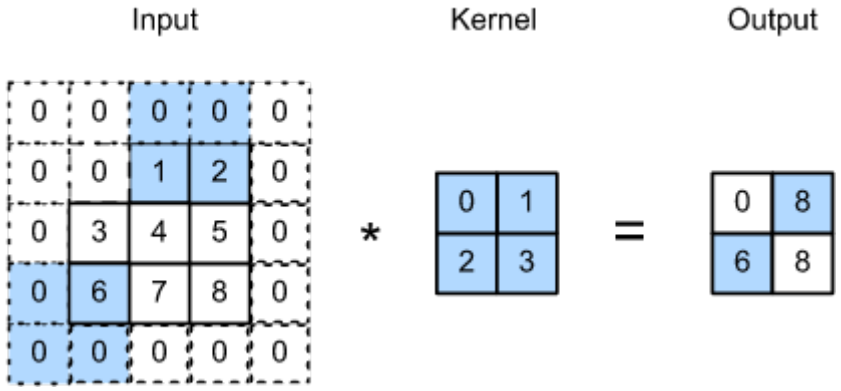


Γενικά αν προσθέσουμε p_h σειρές σαν padding πάνω από την πρώτη σειρά και κάτω από την τελευταία και p_w στήλες σαν padding αριστερά από την πρώτη και δεξιά από την τελευταία τότε οι διαστάσεις την εξόδου του φίλτρου θα είναι οι εξής:

$$(n_h - k_h + p_h + 1) \times (n_w - k_w + p_w + 1).$$

Αυτό σημαίνει ότι το ύψος και το πλάτος του διανύσματος εξόδου θα αυξηθούν κατά p_h και p_w αντίστοιχα. Στην πιο συνήθη περίπτωση οι τιμές του padding είναι οι $p_h = k_h - 1$ και $p_w = k_w - 1$. Με τις συγκεκριμένες τιμές οι διαστάσεις τις εξόδου είναι οι ίδιες με αυτές της εισόδου.

- Το συνελικτικό βήμα δηλώνει το πόσο μετακινείται το φίλτρο σε κάθε βήμα. Η τιμή 1 του συνελικτικού βήματος αντιστοιχεί στην κλασσική συνέλιξη. Αν η τιμή είναι 2, τότε σε κάθε βήμα το φίλτρο μετακινείται 2 θέσεις στον άξονα του πλάτους ή του ύψους αντί για μία. Όσο μεγαλώνει το συνελικτικό βήμα, τόσο μειώνεται το μέγεθος του πλάτους και του ύψους του διανύσματος εξόδου.



Γενικά, όταν το συνελικτικό βήμα για το ύψος του διανύσματος εισόδου είναι s_h και για το πλάτος s_w τότε οι σιαστάσεις της εξόδου του συνελικτικού φίλτρου είν οι παρακάτω:

$$\lfloor (n_h - k_h + p_h + s_h)/s_h \rfloor \times \lfloor (n_w - k_w + p_w + s_w)/s_w \rfloor .$$

Αν θέσουμε για τις τιμές του padding τις τιμές $p_h = k_h - 1$ και $p_w = k_w - 1$, τότε η παραπάνω σχέση μετασχηματίζεται στην παρακάτω:

$$\lfloor (n_h + s_h - 1)/s_h \rfloor \times \lfloor (n_w + s_w - 1)/s_w \rfloor$$

Αν τώρα το ύψος και πλάτος του διανύσματος εισόδου διαιρούνται με τα αντίστοιχα βήματα, τότε οι διαστάσεις της εξόδου θα είναι $(n_h/s_h) \times (n_w/s_w)$.

Μπορούμε να υπολογίσουμε το μέγεθος ενός επίπεδου του διανύσματος εξόδου ενός συνελικτικού φίλτρου σαν συνάρτηση του μεγέθους του διανύσματος εισόδου (W), του δεκτικού πεδίου (F) ή όπως είναι πιο γνωστό receptive field, του συνελικτικού βήματος (S) και του padding (P). Η συνάρτηση αυτή είναι η $(W - F + 2P)/S + 1$. Συνολικά, αν έχουμε ένα συνελικτικό επίπεδο με τις εξής υπερπαραμέτρους: K πλήθος φίλτρων, S μέγεθος βήματος, P padding, και διαστάσεις διανύσματος εισόδου W1×H1×D1, τότε το διάνυσμα εξόδου θα έχει διαστάσεις $W_2 \times H_2 \times D_2$ όπου:

$$W_2 = (W_1 - F + 2P)/S + 1$$

$$H_2 = (H_1 - F + 2P)/S + 1$$

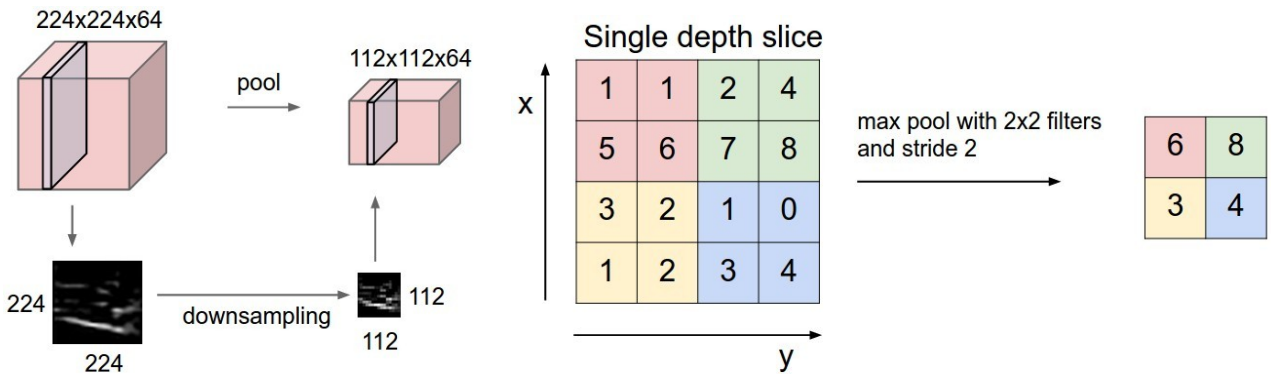
$$D2 = K$$

Συγκεντρωτικό επίπεδο

Είναι πολύ συχνό φαινόμενο στα συνελικτικά νευρωνικά δίκτυα να εισάγουμε μεταξύ διαδοχικών συνελικτικών φίλτρων συγκεντρωτικά φίλτρα (pooling layers). Η λειτουργία των φίλτρων αυτών είναι να μειώνουν το μέγεθος του επιπέδου του διανύσματος εξόδου του συνελικτκού φίλτρου, με σκοπό τη μείωση των παραμέτρων των φίλτρων και του υπολογιστικού κόστους που αυτές επισύρουν. Επίσης λόγω της μείωσης του αριθμού των παραμέτρων του δικτύου αποτρέπει το over-

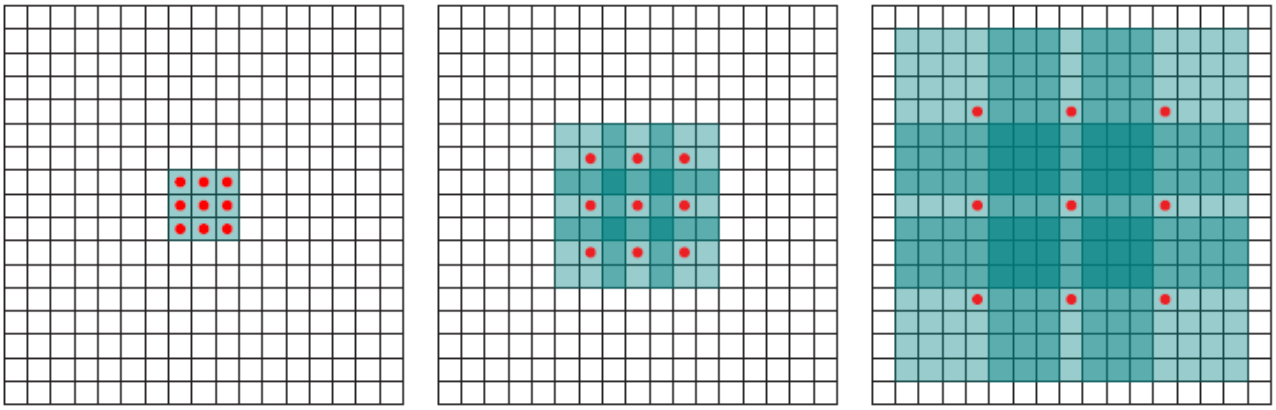
fitting στα δεδομένα εκπαίδευσης. Το συγκεντρωτικό φίλτρο επιδρά ξεχωριστά σε κάθε επίπεδο του διανύσματος εισόδου και του μεταβάλει τις διαστάσεις εφαρμόζοντας σε αυτό μία συνάρτηση (π.χ. max ή avg). Η διάσταση βάθους δεν μεταβάλλεται. Γενικά, ένα συγκεντρωτικό φίλτρο:

- Δέχεται σαν είσοδο ένα διάνυσμα διαστάσεων W1×H1×D1
- Χρειάζεται δύο υπερπαραμέτρους
  - το εμβαδόν του φίλτρου F
  - το βήμα S
- Παράγει σαν έξοδο ένα διάνυσμα διαστάσεων W2×H2×D2 όπου:
  - W2=(W1−F)/S+1
  - H2=(H1−F)/S+1
  - D2=D1
- Δεν έχει εκπαιδεύσιμες παραμέτρους γιατί απλά υπολογίζει μία συνάρτηση



2.3.3. Διασταλμένη Συνέλιξη

Η διασταλμένη συνέλιξη (η όπως είναι γνωστή dialated convolution) διαφέρει από την κανονική συνέλιξη που γίνεται σε ένα συνελικτικό επίπεδο ως προς την απόσταση που έχουν μεταξύ τους τα στοιχεία του συνελικτικού πυρήνα. Συγκεκριμένα, ένα διασταλμένο συνελικτικό επίπεδο δημιουργείται προσθέτοντας κενά (μηδενικά βάρη για την ακρίβεια) μεταξύ των στοιχείων του πυρήνα ενός συνελικτικού επιπέδου. Ο ρυθμός διαστολής (dialation rate) είναι συνάρτηση μίας παραμέτρου d. Η συνάρτηση αυτή από υλοποίηση σε υλοποίηση αλλά συνήθως ο αριθμός των κενών που προστίθενται μεταξύ δύο στοιχείων του συνελικτικού πυρήνα είναι *d-1*, έτσι ώστε για *d=1* η συνέλιξη να είναι η κανονική συνέλιξη που χρησιμοποιείται σε ένα συνελικτικό επίπεδο.



Για να γίνει καλύτερα αντιληπτή η σχέση του συντελεστή διαστολής *d* και το μέγεθος της εξόδου ενός φίλτρου διασταλμένης συνέλιξης (έστω *o*) είναι σκόπιμη η σύγκριση της επίδρασης του *d* στο ουσιαστικό μέγεθος του συνελικτικού πυρήνα. Συγκεκριμένα ένας συνελικτικός πυρήνας μεγέθους *k* διασταλμένος κατά έναν παράγοντα *d* έχει ουσιαστικό μέγεθος $\hat{k}$ που δίνεται από τον παρακάτω τύπο:

$$\hat{k} = k + (k-1)(d-1) \quad .$$

Από το παραπάνω προκύπτει ότι ότι το μέγεθος της εξόδου ο δίνεται από την παρακάτω σχέση:

$$o = i + 2p - k - (k-1)(d-1)s + 1 \quad .$$

όπου i το μέγεθος της εισόδου, s το βήμα (stride) και p το padding

Η διασταλμένη συνέλιξη παρουσιάζει γενικά βελτιωμένα αποτελέσματα από την απλή συνέλιξη που πραγματοποιείται σε ένα συνελικτικό επίπεδο. Χρησιμοποιείται για να μεγαλώσει το μέγεθος του δεκτικού πεδίου (receptive field) χωρίς να απαιτείται αύξηση του μεγέθους του πυρήνα συνέλιξης χωρίς να αυξάνεται το υπολογιστικό κόστος ούτε η απαιτούμενη μνήμη και χωρίς να υπάρχει μείωση της . Αυτό είναι ιδιαίτερα αποτελεσματικό όταν περισσότερα επίπεδα διασταλμένης συνέλιξης στοιβάζονται σε ένα δίκτυο. Χάρη στην ικανότητα της διασταλμένης συνέλιξης να αυξάνει το μέγεθος του δεκτικού πεδίου έχει ως αποτέλεσμα την ευρεία χρήση τέτοιων φίλτρων σε εφαρμογές όπου ενδιαφέρει περισσότερο η αποτύπωση και ανάλυση του γενικού περιεχομένου με χαμηλό κόστος.

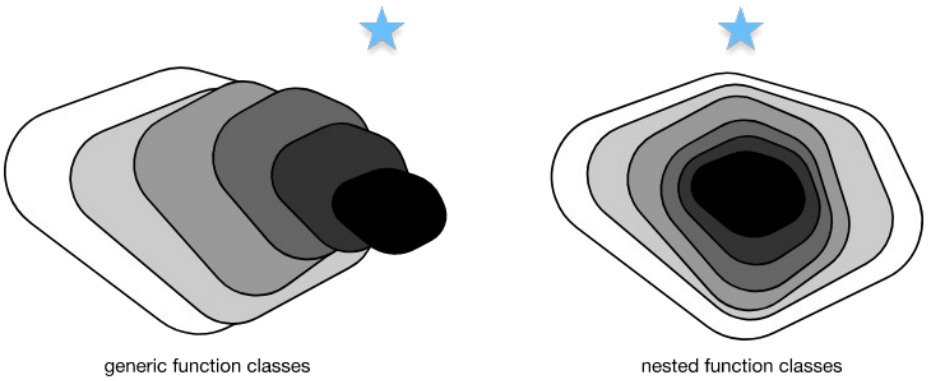
2.3.4. Υπολειμματικά δίκτυα

Τα νευρωνικά δίκτυα επιτελούν το σκοπό της προσέγγισης μιας συνάρτησης. Παρατηρείται ότι όσο πιο βαθύ είναι ένα νευρωνικό δίκτυο, δηλαδή όσο περισσότερο αυξάνεται ο αριθμός των επιπέδων του, τόσο αυξάνεται η ικανότητα του να προσεγγίζει περίπλοκες συναρτήσεις. Όμως πολλές φορές αυτό επηρεάζει αρνητικά την ικανότητα του δικτύου να προσεγγίζει με μεγάλη ακρίβεια πιο απλές συναρτήσεις. Επιπρόσθετα, προσθέτοντας επίπεδα σε ένα νευρωνικό δίκτυο προσπαθώντας να βελτιώσουμε την ακρίβεια με την οποία προσεγγίζει μία συνάρτηση μπορεί να πετύχουμε το αντίθετο. Να μειώσουμε την ακρίβεια του δικτύου. Αλλά γιατί γίνεται αυτό;

Ας θεωρήσουμε το F, την κλάση συναρτήσεων που μπορεί να φτάσει μια συγκεκριμένη αρχιτεκτονική δικτύου (μαζί με το ρυθμό εκμάθησης και άλλες ρυθμίσεις υπερπαραμέτρων). Δηλαδή, για όλα τα $f \in F$ υπάρχει κάποιο σύνολο παραμέτρων W που μπορούν να ληφθούν με την εκπαίδευση σε ένα κατάλληλο σύνολο δεδομένων. Ας υποθέσουμε ότι το είναι η f^b συνάρτηση που θα θέλαμε πραγματικά να βρούμε. Αν είναι σε F, έχει καλώς αλλά συνήθως δεν συμβαίνει αυτό. Αντ 'αυτού, θα προσπαθήσουμε να βρούμε κάποια f_F^b που είναι η καλύτερη προσέγγιση της επιθυμητής συνάρτησης στην κλάση F. Για παράδειγμα, ίσως να προσπαθήσουμε να τη βρούμε επιλύοντας το ακόλουθο πρόβλημα βελτιστοποίησης:

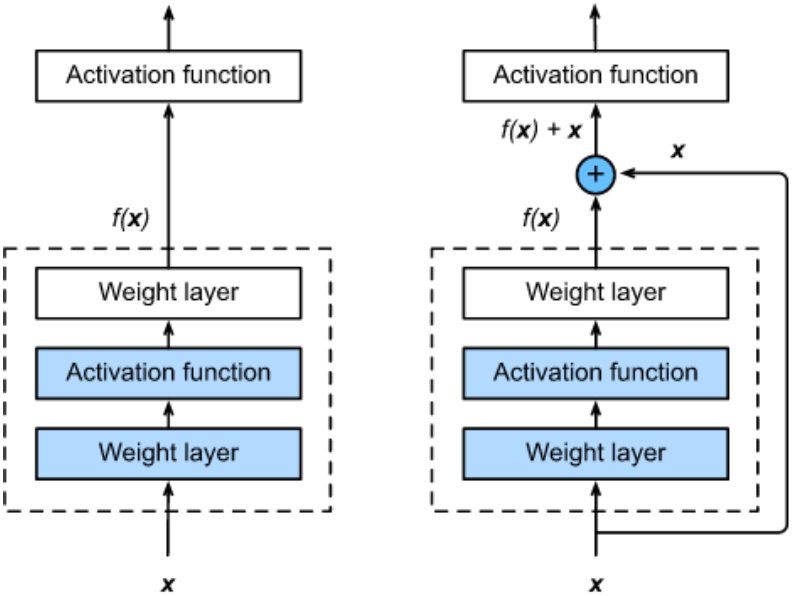
$$f_F^b := argmin\, L(X, Y, f) \quad subject\ to \quad f \in F$$

Είναι λογικό να υποθέσουμε ότι αν σχεδιάσουμε μια διαφορετική και πιο ισχυρή αρχιτεκτονική, θα πρέπει να καταλήξουμε σε ένα καλύτερο αποτέλεσμα. Με άλλα λόγια, αναμένουμε ότι το $f * F$ 'είναι "καλύτερο" από το f_F^b . Ωστόσο, αν $F \not\subseteq F$ δεν υπάρχει καμία εγγύηση ότι αυτό θα πρέπει να συμβεί . Στην πραγματικότητα, το f_F^b 'μπορεί να είναι χειρότερο. Μόνο αν οι μεγαλύτερες κλάσεις συναρτήσεων περιέχουν τις μικρότερες, εγγυόμαστε ότι η αύξηση του βάθους του δικτύου θα μας οδηγήσει πιο κοντά στην επιθυμητή συνάρτηση.



Η ιδέα πίσω από τα υπολειμματικά νευρωνικά δίκτυα είναι ότι κάθε πρόσθετο επίπεδο θα πρέπει να περιέχει τη ταυτοτική συνάρτηση (f(x) = x) ως ένα από τα στοιχεία του. Αυτό σημαίνει ότι αν μπορέσουμε να εκπαιδεύσουμε το νεοεισαγόμενο επίπεδο στη συνάρτηση f(x) = x, το νέο μοντέλο θα είναι τουλάχιστον εξίσου αποτελεσματικό με το αρχικό μοντέλο.

Ας επικεντρωθούμε σε ένα νευρωνικό δίκτυο, όπως απεικονίζεται παρακάτω όπου x η είσοδος του δικτύου. Υποθέτουμε ότι η ιδανική συσχέτιση που θέλουμε να αποκτήσουμε με την εκμάθηση είναι f(x), για να χρησιμοποιηθεί ως είσοδος στη συνάρτηση ενεργοποίησης. Το τμήμα μέσα στο κουτί διακεκομμένης γραμμής στην αριστερή εικόνα πρέπει να ταιριάζει άμεσα στη συσχέτιση f(x). Αυτό μπορεί να είναι δυσκολότερο απ' ότι ακούγεται αν στην πραγματικότητα δεν χρειαζόμαστε αυτό το συγκεκριμένο επίπεδο και θα ήταν προτιμότερο να διατηρήσουμε την είσοδο x. Το τμήμα μέσα στο κουτί διακεκομμένης γραμμής στη δεξιά εικόνα τώρα χρειάζεται μόνο να παραμετροποιήσει την απόκλιση από την ταυτότητα, αφού επιστρέφουμε x + f (x). Στην πράξη, η συσχέτισή διαφορών είναι συχνά ευκολότερη στη βελτιστοποίηση. Απαιτείται μόνο να ορίσουμε το f(x) = 0. Η δεξιά εικόνα απεικονίζει το βασικό υπολειμματικό μπλοκ του ResNet. Παρόμοιες αρχιτεκτονικές προτάθηκαν αργότερα και για μοντέλα αναδρομικών δικτύων αλλά χρησιμοποιήθηκαν για πρώτη φορά και χρησιμοποιούνται ευρέως τα συνελικτικά νευρωνικά δίκτυα.



Συνοψίζοντας, είναι δυνατόν να παραληφθεί η εκπαίδευση κάποιον επιπέδων ενός δικτύου χρησιμοποιώντας υπολειμματικές συνδέσεις, όπως φαίνεται και παραπάνω. Ουσιαστικά, οι υπολειμματικές συνδέσεις επιτρέπουν τη ροή πληροφοριών από τα αρχικά επίπεδα του δικτύου προς τα τελευταία. Αυτό προσφέρει σημαντική βελτίωση στη διαδικασία της οπισθοδιάδωσης των μερικών παραγώγων και συμβάλει στην ομαλότερη εκπαίδευση ενός νευρωνικού δικτύου πολλών επιπέδων.

2.3.5. Διαχωριζόμενη συνέλιξη

Σε ένα τυπικό συνελικτικό επίπεδο ενός νευρωνικού δικτύου ο αριθμός παραμέτρων υπολογίζεται από τον τύπο: μέγεθος εισόδου * μέγεθος εξόδου * πλάτος φίλτρου * ύψος φίλτρου. Γίνεται εύκολα κατανοητό ότι ο αριθμός αυτός μπορεί να γίνει αρκετά μεγάλος πολύ εύκολα. Αυτό έχει ως συνέπεια την αύξηση του υπολογιστικού κόστους του μοντέλου λόγω του μεγάλου αριθμού υπολογισμών που πρέπει να γίνουν καθώς και τον κίνδυνο να προκύψει over-fitting λόγω του μεγάλου αριθμού παραμέτρων που εμφανίζονται στο συνελικτικό επίπεδο. Για να αντιμετωπιστεί το πρόβλημα αυτό έχουν μελετηθεί διαφόρων ειδών συνελίξεις, μία εκ των οποίων είναι η διαχωριζόμενη συνέλιξη (separable convolution). Υπάρχουν δύο είδη διαχωριζόμενης συνέλιξης, η χωρική (spatial) και η εις βάθος (depth-wise) διαχωριζόμενη συνέλιξη, Θα αναφερθούμε και στις δύο αλλά θα επικεντρωθούμε στη δεύτερη καθώς η χρήση της είναι πιο διαδεδομένη, ενώ είναι και η μόνη από τις δύο που χρησιμοποιήθηκε στα πλαίσια τις συγκεκριμένης διπλωματικής εργασίας.

Χωρική διαχωριζόμενη συνέλιξη

Η χωρική διαχωριζόμενη συνέλιξη, ουσιαστικά, χωρίζει μία συνέλιξη σε δύο επι μέρους συνελίξεις, των οποίων η σύνθεση δίνει την αρχική πράξη. Συγκεκριμένα, στο συγκεκριμένο είδος συνέλιξης, ο πυρήνας συνέλιξης διασπάται σε δύο, ο πολλαπλασιασμός των οποίων επιστρέφει τον αρχικό πυρήνα, όπως φαίνεται παρακάτω.

$$\begin{bmatrix} 3 & 6 & 9 \\ 4 & 8 & 12 \\ 5 & 10 & 15 \end{bmatrix} = \begin{bmatrix} 3 \\ 4 \\ 5 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}$$

Με τον τρόπο αυτό επιτυγχάνεται η μείωση των απαιτούμενων πράξεων (από 9 σε 3+3=6 στο παράδειγμα), άρα και η μείωση του υπολογιστικού κόστους. Παρ' όλα αυτά δεν μπορεί να εφαρμοστεί σε οποιονδήποτε πυρήνα, καθώς δεν είναι όλοι οι πυρήνες διαχωρίσιμοι.

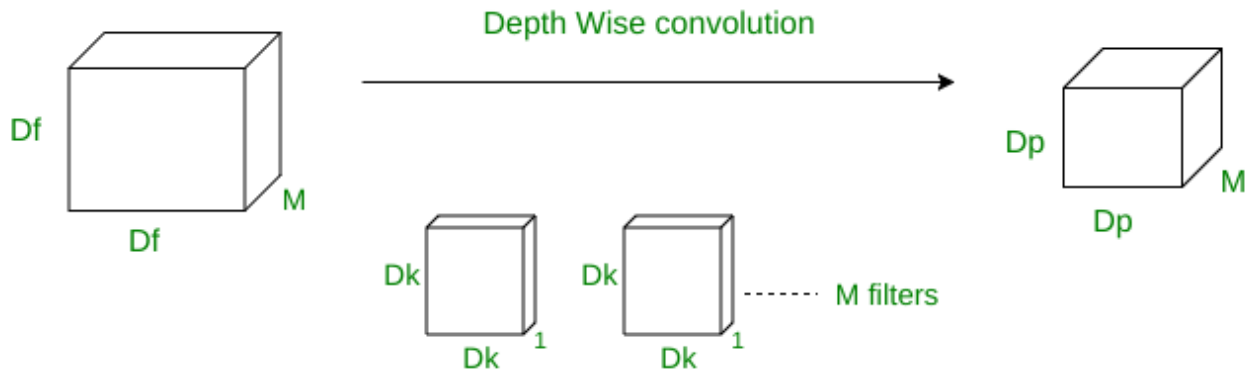
Εις βάθος διαχωριζόμενη συνέλιξης

Η εις βάθος διαχωριζόμενη συνέλιξη (depthwise separable convolution) έχει την ίδια φιλοσοφία με την χωρική, αλλά την υλοποιεί με διαφορετικό τρόπο.

Όπως είδαμε στην κανονική συνέλιξη που πραγματοποιεί ένα convolutional layer η συνέλιξη γίνεται στον χώρο, δηλαδή όλα τα δισδιάστατα επίπεδα του τρισδιάστατου διανύσματος εισόδου μεγέθους (πλάτος, ύψος, 1) παράγουν ένα διάνυσμα εξόδου διαστάσεων (πλάτος, ύψος, 1) για κάθε φίλτρο. Άρα η συνολικές διαστάσεις της εξόδου είναι (πλάτος, ύψος, αριθμός φίλτρων). Στην εις βάθος διαχωριζόμενη συνέλιξη η συνελικτική πράξη γίνεται σε δύο στάδια, την εις βάθος συνέλιξη (depthwise convolution) και την σημειακή συνέλιξη (pointwise convlution).

Εις βάθος συνέλιξη

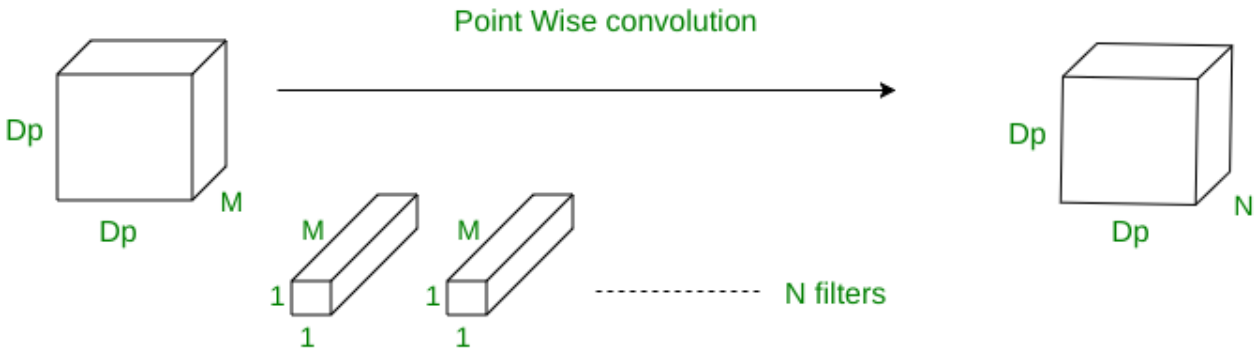
Στην εις βάθος συνέλιξη, η συνελικτική πράξη εφαρμόζεται σε κάθε κανάλι της εισόδου ξεχωριστά. Έτσι το κάθε φίλτρο έχει διαστάσεις ($Dk \, x \, Dk \, x \, 1$) όπου Dk το μέγεθος του φίλτρου. Δεδομένου ότι υπάρχουν M κανάλια στο διάνυσμα εισόδου, χρειάζονται M τέτοια φίλτρα. Οι διαστάσεις του διανύσματος εξόδου θα είναι ($Dp \, x \, Dp \, x \, M$).



Μία συνέλιξη απαιτεί $Dk \, x \, Dk$ πολλαπλασιασμούς. Δεδομένου το διάνυσμα εξόδου έχει διαστάσεις $Dp \, x \, Dp$ πρέπει να υπολογιστούν τόσοι παράμετροι για κάθε κανάλι. Άρα ο συνολικός αριθμός πολλαπλασιασμών που πρέπει να γίνουν για την εις βάθος συνέλιξη είναι $M \, x \, Dk_2 \, x \, Dp_2$.

Σημειακή συνέλιξη

Στην σημειακή συνέλιξη, εφαρμόζεται η πράξη με ένα $1 \, x \, 1$ φίλτρο για κάθε ένα από τα M κανάλια του ενδιάμεσου διανύσματος. Άρα οι διαστάσεις του κάθε φίλτρου θα είναι $1 \, x \, 1 \, x M$. Δεδομένου ότι έχουμε N φίλτρα το διάνυσμα εξόδου θα έχει διαστάσεις $Dp \, x \, Dp \, x \, N$.



Κάθε συνέλιξη απαιτεί M πολλαπλασιασμούς. Δεδομένου ότι το φίλτρο κινείται $Dp\,x\,Dp$ φορές ο συνολικός αριθμός πολλαπλασιασμών για τη σημειακή συνέλιξη είναι $M\,x\,Dp_2\,x\,N$.

Άρα για για ολόκληρη τη διαχωριζόμενη εις βάθος συνέλιξη απαιτούνται $M\,x\,Dp_2\,x\,(Dk_2+N)$ πολλαπλασιασμοί. Για μία κανονική συνέλιξη απαιτούνται $N\times Dp_2\times Dg_2\times M$ πολλαπλασιασμοί. Για μεγάλου μεγέθους εισόδους η διαχωριζόμενη εις βάθος συνέλιξη απαιτεί σημαντικό μικρότερο αριθμό πολλαπλασιασμών, κάτι που σημαίνει ότι προφέρει ταχύτητα χωρίς να επηρεάζει αρνητικά την ακρίβεια του δικτύου

2.4. Μέθοδοι δειγματοληψίας

Η εκπαίδευση ενός νευρωνικού δικτύου μας παρέχει μια κατανομή πιθανοτήτων πάνω στα διάφορα labels του πειράματος. Το πως θα διαχειριστούμε αυτές τις κατανομές για να εξάγουμε συμπεράσματα για τα test data, δεδομένα δηλαδή που δεν χρησιμοποιήθηκαν κατά τη διαδικασία εκπαίδευσης και άρα δεν τα γνωρίζει το δίκτυο, ορίζει την μέθοδο δειγματοληψίας. Η μέθοδος δειγματοληψίας παίζει καθοριστικό ρόλο στην παραγωγή αποτελεσμάτων καθώς καθορίζουν τη σχέση μεταξύ προβλέψεων του εκπαιδευμένου μοντέλου (κατανομή πιθανοτήτων) και τελικών προβλέψεων (πραγματική πρόβλεψη). Για τα μοντέλα με τα οποία ασχοληθήκαμε στην συγκεκριμένη διπλωματική εργασία χρησιμοποιήσαμε διάφορες μεθόδους δειγματοληψίας με σκοπό να συγκρίνουμε τα αποτελέσματά τους και τις αξιολογήσουμε στο task της αυτόματης εναρμόνισης μελωδίας. Συγκεκριμένα, οι μέθοδοι που χρησιμοποιήσαμε είναι οι εξής:

Επιλογή μέγιστης πιθανότητας

Σε αυτή τη μέθοδο δειγματοληψίας επιλέγουμε σαν τελική πρόβλεψη την πρόβλεψη του μοντέλου με την μεγαλύτερη τιμή πιθανότητας. Αυτό γίνεται εφαρμόζοντας τη συνάρτηση *argmax* στο διάνυσμα εξόδου του μοντέλου που αντιστοιχεί στην κατανομή πιθανότητας του κάθε label. Προκύπτει ένα διάνυσμα με μηδενικά σε όλες τις θέσεις του εκτός από τη θέση του label (ή των label για περιπτώσεις multi label classification) με την μεγαλύτερη τιμή πιθανότητας. Το label (τα labels) αυτό είναι και η τελική πρόβλεψη. Τα τελικά αποτελέσματα της επιλογής αυτή είναι πάντα ίδια, καθώς πάντα επιλέγεται το label (τα labels) με την μεγαλύτερη πιθανότητα που προκύπτει κατευθείαν από το μοντέλο. Αυτό συμβαίνει λόγω της ντετερμινιστικής φύσης των νευρωνικών δικτύων. Αυτό το γεγονός παρουσιάζει διάφορα μειονεκτήματα σε σχέση με μία άλλη μη ντετερμινιστική επιλογή προβλέψεων. Βασικό μειονέκτημα είναι το γεγονός ότι θα θέλαμε ένα σύστημα παραγωγής αλληλουχιών συγχορδιών να μπορεί να δώσει διαφορετικές προβλέψεις για μία μελωδία, δεδομένου ότι διαισθητικά δεν υπάρχει μόνο μία σωστή ακολουθία συγχορδιών για

μία μελωδία. Επίσης, ένα άλλο σημαντικό μειονέκτημα είναι το γεγονός ότι αν χρησιμοποιήσουμε μεγάλες σε μήκος ακολουθίες μπορεί να εμφανιστεί το φαινόμενο αναδίπλωσης των προβλέψεων της εξόδου του δικτύου, όπου το δίκτυο αδυνατεί να παράξει κάτι διαφορετικό αν κολλήσει σε μία συγκεκριμένη ακολουθία.

Για το λόγο αυτό, στα πλαίσια της διπλωματικής αυτής εργασίας, χρησιμοποιήσαμε και κάποιες μη – ντετερμινιστικές στρατηγικές επιλογής συγχορδιών που παρουσιάζονται παρακάτω.

Πολυωνυμική επιλογή με θερμοκρασία

Η μέθοδος αυτή χρησιμοποιήθηκε την περίπτωση multi-class single-label classification, δηλαδή ο αριθμός των κλάσεων είναι μεγαλύτερος του δύο αλλά κάθε είσοδος αντιστοιχίζεται σε ένα label. Σε αυτή την περίπτωση το μοντέλο προβλέπει μια τιμή πιθανότητας για κάθε κλάση και τ άθροισμα των τιμών αυτών όλων των κλάσεων ανέρχεται σε 1. Σε αυτή τη μέθοδο δειγματοληψίας επιλέγεται σαν τελική πρόβλεψη κάποιο ή κάποια label. Η πιθανότητα επιλογής του κάθε label αντιστοιχεί στην πιθανότητα που προέβλεψε το μοντέλο για το κάθε label δημιουργώντας ένα διάνυσμα $\boldsymbol{p}=(p_1,\ldots,p_n)$ για τα n labels. Η τιμή αυτή προήλθε μετά την εφαρμογή μια συνάρτησης *softmax*

$$p_i=\frac{\exp(z_i/T)}{\sum_j^n \exp(z_j/T)}$$

πάνω στις αρχικές προβλέψεις $\boldsymbol{z}=(z_{1,}\ldots,z_n)$, όπου T η παράμετρος της θερμοκρασίας. Για $T=1$ η παραπάνω συνάρτηση είναι η κανονική *softmax*. Μεγαλύτερες τιμές του T οδηγούν σε μικρότερες τιμές του $(z\ /\ T)$. Η εφαρμογή της συνάρτησης *softmax* σε μικρότερες τιμές οδηγεί σε μία πιο ισορροπημένη κατανομή όπου η διαφορά μεγάλων και μικρών πιθανοτήτων είναι μικρή. Αυτό έχει ως συνέπεια οι προβλέψεις να έχουν μεγαλύτερη διακύμανση και ποικιλία, αλλά αυξάνεται και ο αριθμός των λάθος προβλέψεων. Αντίθετα μικρότερες τιμές του T οδηγούν σε μεγαλύτερες τιμές του $(z\ /\ T)$. Η εφαρμογή της συνάρτησης *softmax* σε μεγαλύτερες τιμές οδηγεί σε αύξηση των μεγάλων τιμών και σμίκρυνση των μικρότερων τιμών οδηγώντας σε πιο σίγουρες αλλά και περισσότερο συντηρητικές προβλέψεις. Για πολύ μεγάλες τιμές του T όλα τα labels τείνουν να έχουν την ίδια πιθανότητα ενώ για πολύ μικρές τιμές του T η μεγαλύτερη τιμή από τις z_i τείνει να γίνει ίση με 1.

Gibbs sampler

Η μέθοδος δειγματοληψίας Gibbs είναι μία πιο σύνθετη μέθοδος επιλογής δειγμάτων από μία κατανομή δεσμευμένων πιθανοτήτων. Αποτελεί μία μέθοδο Markov Chain Monte Carlo. Η ιδέα πίσω από αυτή τη μέθοδο δειγματοληψίας είναι η επιλογή δειγμάτων από διάφορες μεταβλητές μίας κατανομής, κάνοντας τυχαίες προβλέψεις για κάθε μεταβλητή ξεχωριστά ενώ παράλληλα διατηρώντας τις τιμές των υπολοίπων μεταβλητών σταθερές. Για να γίνει αυτό καλύτερα κατανοητό παραθέτουμε το παρακάτω παράδειγμα.

Έστω 3 τυχαίες μεταβλητές X_1, X_2 και X_3. Αρχικά αρχικοποιούμε τις μεταβλητές αυτές είτε τυχαία είτε σύμφωνα με μία προγενέστερη κατανομή. Σε κάθε επανάληψη i επιλέγουμε δείγματα από κάθε μεταβλητή, δηλαδή $x_1(i) \sim p(X_1 = x_1|X_2 = x_2(i-1)\ ,\ X_3 = x_3(i-1)\ )$, $x_2(i) \sim p(X_2 = x_2|X_1 = x_1(i)\ ,\ X_3 = x_3(i-1))$, $x_2(i) \sim p(X_3 = x_3|X_1 = x_1(i)\ ,\ X_2 = x_2(i)$. Η διαδικασία αυτή συνεχίζεται μέχρι να συγκλίνουν οι τιμές που προέκυψαν από τη δειγματοληψία με τις πραγματικές τιμές της κατανομής. Ο αριθμός των επαναλήψεων είναι πάρα πολύ μεγάλος, τυπικά άπειρος, αλλά στην πράξη μετά από έναν αριθμό επαναλήψεων της τάξης του 10^4 επιτυγχάνεται η επιθυμητή σύγκλιση

των τιμών. Όπως όλοι οι Markov Chain Monte Carlo αλγόριθμοι, έτσι και ο αλγόριθμος Gibbs, παράγουν δείγματα προσεγγίζοντας ασυμπτωτικά τις πραγματικές κατανομές πιθανοτήτων. Σε πολλές περιπτώσεις όμως αποτελούν την μοναδική πρακτική λύση που υπάρχει. Ο αλγόριθμος της μεθόδου δειγματοληψίας Gibbs είναι ο παρακάτω:

Gibbs sampler Algorithm

Initialize $x(0) \sim q(x)$
for *iteration* $i = 1, 2, \ldots$ ***do***
$x_1(i) \sim p(X_1 = x_1 | X_2 = x_2(i-1), X_3 = x_3(i-1), \ldots, X_D = x_D(i-1))$
$x_2(i) \sim p(X_2 = x_2 | X_1 = x_1(i), X_3 = x_3(i-1), \ldots, X_D = x_D(i-1))$
. . .
$x_D(i) \sim p(X_D = x_D | X_1 = x_1(i), X_2 = x_2(i), \ldots, X_D = x_{D-1}(i))$
end for

Δειγματοληψία με χρήση κατωφλίου

Για την περίπτωση του multi label classification προβλήματος, η νότες που προβλέπει το σύστημα μπορεί και συνήθως είναι περισσότερες από μία. Για να επιλέξουμε τις νότες που θα δέχεται σαν πρόβλεψη το μοντέλο θέτουμε μία τιμή κατωφλίου (συγκεκριμένα διαλέξαμε την τιμή 0.5). Οποιαδήποτε πρόβλεψη του μοντέλου στην οποία ένα στοιχείο της ακολουθίας εισόδου αντιστοιχίζεται σε μία κλάση με πιθανότητα μεγαλύτερη από 0.5 δεχόμαστε ότι το στοιχείο αυτό ανήκει στη συγκεκριμένη κλάση.

Η συγκεκριμένη μέθοδος δειγματοληψίας παρουσιάζει ένα πρόβλημα ή καλύτερα μία ιδιαιτερότητα που έρχεται σε αντιδιαστολή με τον σκοπό που επιτελεί το μοντέλο. Πιο συγκεκριμένα, στα δεδομένα οι παύσεις αναπαριστώνται σαν μηδενικά διανύσματα, δηλώνουν δηλαδή ότι καμία νότα δεν ηχεί τη χρονική στιγμή που αντιστοιχεί στη θέση του διανύσματος. Αυτό έχει ως αποτέλεσμα το μοντέλο να μαθαίνει να χρησιμοποιεί παύσεις στην αρμονία που προβλέπει. Αυτό θα θέλαμε να το αποφύγουμε καθώς θεωρούμε ότι το μοντέλο θα πρέπει να παράγει πάντα συγχορδίες εξόδου. Εκτός αυτού, υπάρχει περίπτωση το μοντέλο κατά την διαδικασία πρόβλεψης να δώσει σαν έξοδο μια κατανομή πιθανοτήτων για μία χρονική στιγμή στην οποία να μην υπάρχουν στοιχεία με τιμή πιθανότητας μεγαλύτερη από αυτή του κατωφλίου ή να υπάρχει ένα μόνο στοιχείο στο οποίο να αποδόθηκε πιθανότητα μεγαλύτερη της τιμής του κατωφλίου. Έτσι προκύπτει το πρόβλημα ότι η συγχορδία που προβλέπει το μοντέλο δεν είναι στην πραγματικότητα συγχορδία αλλά παύση ή μία νότα μεμονωμένη. Αυτό έρχεται σε αντίθεση με τη λειτουργία του μοντέλου.

Για τους παραπάνω λόγους εκτός από την κλασική μέθοδο δειγματοληψίας με τη χρήση κατωφλίου χρησιμοποιήσαμε και άλλη μία υβριδική μέθοδο δειγματοληψίας που αντιμετωπίζει σε κάποιο τα προβλήματα που εμφανίζονται με τη χρήση της κλασσικής μεθόδου δειγματοληψίας με χρήση κατωφλίου. Ουσιαστικά αυτό που κάνουμε στη μέθοδο αυτή είναι να ελέγχουμε πόσες κλάσεις αντιστοιχίζονται σε τιμές πιθανότητας μεγαλύτερες από αυτή του κατωφλίου. Αν οι κλάσεις, δηλαδή οι νότες, αυτές είναι 3 ή περισσότερες δεν αλλάζουμε καθόλου την πρόβλεψη του μοντέλου. Αν είναι λιγότερες από 3, τότε δεν συνιστούν μία τρίφωνη συγχορδία και άρα αποδεχόμαστε τις τρεις κλάσεις στις οποίες αποδόθηκαν οι τρεις μεγαλύτερες πιθανότητες. Αυτή η μέθοδος αντιμετωπίζει το πρόβλημα τις εμφάνισης παύσεων και ελλιπών συγχορδιών εξόδου αλλά

η πιθανότητα σφάλματος στην πρόβλεψη αυξάνεται καθώς αποδεχόμαστε κλάσεις οι οποίες έχουν χαμηλή πιθανότητα (μικρότερη της τιμής του κατωφλίου).

Σημειώνουμε ότι και οι δύο μέθοδοι δειγματοληψίας που χρησιμοποιήθηκαν για τη διεξαγωγή προβλέψεων του multi label classification μοντέλου αποτελούν ντετερμινιστικές μεθόδους.

3.Αναπαράσταση δεδομένων

3.1. Προέλευση dataset

Για την εκπόνηση της συγκεκριμένης διπλωματικής εργασίας χρειαστήκαμε πληθώρα μουσικών τραγουδιών, διαφορετικών μουσικών ειδών, σε συμβολική γραφή. Λέγοντας συμβολική γραφή εννοούμε ότι οι πληροφορίες των τραγουδιών που έχουν να κάνουν με νότες και συγχορδίες είναι κωδικοποιημένες και δεν χρειάστηκε να τις εξάγουμε εμείς με κάποιον τρόπο από κάποιο αρχείο μουσικής, όπως για παράδειγμα ένα αρχείο .mp3 ή .wav. Για το λόγο αυτό κατεβάσαμε ένα dataset 8700 τραγουδιών σε μορφή MusicXML από το HookTheory. Η αρχική μορφή των κομματιών, για να είμαστε περισσότερο ακριβείς, ήταν αρχεία κειμένου (.txt files), τα οποία μετατρέψαμε σε MusicXml για να μπορέσουμε να τα χρησιμοποιήσουμε μέσω της βιβλιοθήκης MiniDom της Python.

3.2. Στόχος προεπεξεργασίας δεδομένων

Πριν εξηγηθεί ο τρόπος με τον οποίο τα δεδομένα που συλλέχθηκαν από το HookTheory σε μορφή xml αρχείων μετατράπηκαν σε numpy arrays κρίνεται σημαντικό να τονιστεί η τελική μορφή που θα πρέπει να έχει το dataset έτσι ώστε να μπορεί να χρησιμοποιηθεί για την εκπαίδευση των μοντέλων. Κατ’ αρχήν για κάθε μοντέλο θα πρέπει να μετατρέψουμε διαφορετικά το dataset καθώς οι διαστάσεις των εισόδων κάθε μοντέλου είναι διαφορετικές, Επίσης για κάθε μοντέλο δοκιμάσαμε αρκετές μορφές διανυσμάτων εισόδου και συγκρίναμε τα αποτελέσματα του κάθε dataset. Στη συνέχεια θα περιγράψουμε τη μορφή που θα πρέπει να έχει το dataset για την εκπαίδευση του κάθε μοντέλου:

Coconet

Το Coconet δέχεται σαν είσοδο ένα διάνυσμα με τις εξής διαστάσεις: (sequence_length, 4) όπου sequence_length είναι το μήκος του κομματιού σε δέκατα έκτα και ο αριθμός 4 είναι ο αριθμός των φωνών. Το μήκος του κομματιού δεν είναι fixed όπως ο αριθμός των φωνών. Το coconet σχεδιάστηκε για την αυτόματη σύνθεση Chorales του J.S.Bach. Τα chorales είναι μουσικά κομμάτια που αποτελούνται από 4 φωνές. Κάθε φωνή είναι μία ακολουθία νοτών. Κάθε στήλη του πίνακα αντιστοιχεί σε μία φωνή. Για να μπορεί το μοντέλο να επιτελέσει το task της εύρεσης συγχορδιών για κάποια μελωδία δημιουργήσαμε το dataset για την εκπαίδευση του με τον παρακάτω τρόπο: Δημιουργήσαμε έναν πίνακα διαστάσεων (sequence_length, 1) από τις μελωδίες των τραγουδιών που είχαμε στη διάθεσή μας. Στη συνέχεια δημιουργήσαμε έναν δεύτερο διάνυσμα διαστάσεων (sequence_length, 3) από τις συγχορδίες των τραγουδιών του dataset. Τέλος ενώνοντας τα δύο αυτά διανύσματα εξάγουμε το τελικό διάνυσμα διαστάσεων (sequence_length, 4) το οποίο

τροφοδοτούμε στο μοντέλο για την εκπαίδευσή του. Αξίζει να σημειωθεί ότι με τον τρόπο αυτό οι συγχορδίες που χρησιμοποιούμε μπορούν να αποτελούνται από 3 νότες το πολύ. Επειδή στην μουσική, εκτός από τις τρίφωνες συγχορδίες, υπάρχουν και συγχορδίες οι οποίες αποτελούνται από περισσότερες από 3 νότες, αναγκαστικά κάποιες από τις νότες αυτές παραλείφθηκαν. Η επιλογή των νοτών αυτών έγινε με γνώμονα τη λειτουργία τους στη συγχορδία και τις συνέπειες που θα είχε η αφαίρεσή τους από αυτή.

Sequence to Sequence

Εκπαιδεύσαμε ένα sequence to sequence μοντέλο για την υλοποίηση δύο πειραμάτων. Το πρώτο πείραμα είναι ένα multi-class classification πρόβλημα, με σκοπό να αντιστοιχίσουμε μια ακολουθία νοτών με μία ακολουθία συγχορδιών. Για την αναπαράσταση τον νοτών Το δεύτερο πείραμα είναι ένα multi-label classification πρόβλημα όπου αντιστοιχίζουμε μια ακολουθία νοτών σε μια ακολουθία νοτών, όμως η ακολουθία εξόδου σε αντίθεση με την ακολουθία εισόδου μπορεί να περιέχει περισσότερες νότες (άρα περισσότερα από ένα labels). Και στις δύο περιπτώσεις χρησιμοποιήθηκε 1-hot encoding, δηλαδή τα διανύσματα περιέχουν μόνο 0 και 1.

Multi-class classification

Η είσοδος του sequence to sequence μοντέλου έχεις τις εξής διαστάσεις: (batch_size, sequence_length, 14) για την είσοδο του κωδικοποιητή και (batch_size, sequence_length, 38) για την είσοδο του αποκωδικοποιητή. Για το λόγο αυτό τα dataset που κατασκευάσαμε για το μοντέλο αυτό έχουν διαστάσεις (number_of_tracks, sequence_length, 14) για τον κωδικοποιητή και (number_of_tracks, sequence_length, 38) αντίστοιχα για τον αποκωδικοποιητή. Η πρώτη διάσταση (number of tracks) είναι ο συνολικός αριθμός τραγουδιών που υπάρχουν στο dataset από το HookTheory. Τα τραγούδια τροφοδοτούνται στο μοντέλο ανάλογα με το batch_size. Η δεύτερη διάσταση (sequence_length) αναφέρεται στο μήκος ενός κομματιού σε δέκατα έκτα. Στην περίπτωση του sequence to sequence μοντέλου, το sequence_length είναι σταθερός αριθμός για να μπορέσει να δημιουργηθεί ο τρισδιάστατος πίνακας που αποτελεί το dataset. Σαν sequence_length χρησιμοποιήθηκαν διάφορες τιμές, όπως το μέγιστο μήκος κομματιού του dataset, και αξιολογήθηκαν τα αποτελέσματα για κάθε τιμή ου sequence_length. Τέλος, η τρίτη διάσταση, η οποία για τον κωδικοποιητή είναι (14) και για τον αποκωδικοποιητή του μοντέλου είναι (38) προκύπτει: α) από τον αριθμό νοτών που υπάρχουν και β) από τον αριθμό διαφορετικών συγχορδιών που μπορούν να δημιουργηθούν από τους 7 φυσικούς τρόπους, δηλαδή 12 μείζονες, 12 ελάσσονες και 12 ελαττωμένες (μία για κάθε μία από τις 12 συνολικά νότες). Άρα ο αριθμός των νοτών ανέρχεται σε 12 και ο αριθμός των συγχορδιών ανέρχεται σε 36. Οι τελικοί αριθμοί 14 και 38 προκύπτουν από την προσθήκη ενός αρχικού και ενός τερματικού χαρακτήρα. Αυτό συνέβη γιατί το fixed sequence_length οδήγησε στην ανάγκη να συμπληρώσουμε τους πίνακες με μηδενικές γραμμές. Με τους χαρακτήρες αυτούς σηματοδοτείται η αρχή και το τέλος του κομματιού.

Multi-label classification

Η είσοδος του sequence to sequence μοντέλου έχεις τις εξής διαστάσεις: (batch_size, sequence_length, 14) για την είσοδο του κωδικοποιητή και (batch_size, sequence_length, 14) για την είσοδο του αποκωδικοποιητή. Η είσοδος του κωδικοποιητή είναι πανομοιότυπη με την περίπτωση του multi-class classification. Για την είσοδο του αποκωδικοποιητή κατασκευάσαμε ένα

numpy array διαστάσεων (number_of_tracks, sequence_length, 14). Κάθε θέση του διανύσματος αντιπροσωπεύει μία νότα όπως και στο διάνυσμα εισόδου με τη διαφορά ότι εδώ μπορούν περισσότερες από μία θέσεις από τις 14 να πάρουν την τιμή 1, εκτός αν πρόκειται για τους χαρακτήρες έναρξης και τερματισμού του τραγουδιού.

3.3. Προεπεξεργασία δεδομένων

Η μορφή ενός αρχείου MusicXML είναι η παρακάτω:


```
<?xml version="1.0" encoding="UTF-8"?>
<super>
  <meta>
    <artist></artist>
    <title>Melodies of life japanese -  Emiko Shiratori</title>
    <beats_in_measure>4</beats_in_measure>
    <bpm>68</bpm>
    <key>E</key>
    <youtubeid>fmShPzYD9ng</youtubeid>
    <mode>1</mode>
    <sections>
      &lt;_Verse&gt;
        <global_start>18.89</global_start>
        <section_duration>38.83</section_duration>
        <active_start>3.53</active_start>
        <active_stop>35.3</active_stop>
      <!--_Verse-->
    </sections>
  </meta>
  <sections>
    &lt;_Verse&gt;
      <segment>
        <instrument></instrument>
        <rhythm></rhythm>
        <lyrics></lyrics>
        <notes>
          <note>
            <start_beat_abs>0</start_beat_abs>
            <start_measure>1</start_measure>
            <start_beat>1</start_beat>
            <note_length>1</note_length>
            <scale_degree>3</scale_degree>
            <octave>0</octave>
            <isrest>0</isrest>
          </note>
         </notes>
      .
      .
```

.
<chords>
<chord>
<sd>1</sd>
<fb></fb>
<sec></sec>
<emb>add9</emb>
<sus></sus>
<pedal></pedal>
<alternate></alternate>
<borrowed></borrowed>
<chord_duration>1</chord_duration>
<start_measure>1</start_measure>
<start_beat>1</start_beat>
<start_beat_abs>0</start_beat_abs>
<isrest>0</isrest>
</chord>
.
.
.
</chords>
<nummeasures>4</nummeasures>

</sections>
</super>

Για να μπορέσουμε να χρησιμοποιήσουμε το παραπάνω dataset για την εκπαίδευση των δικτύων μας έπρεπε να μετατραπεί σε πίνακες αριθμών οι οποίοι να αποδίδουν με κάποιον τρόπο τις πληροφορίες που εμπεριέχονται στα MusicXML αρχεία. Επίσης, για το κάθε μοντέλο που εξετάσαμε χρησιμοποιήσαμε διαφορετικούς πίνακες. Στη συνέχεια θα παρουσιαστεί ο τρόπος με τον οποίο αντλήσαμε τις πληροφορίες από τα tags των MusicXML αρχείων και πως μετατρέψαμε το dataset σε numpy arrays.

Αρχικά είναι απαραίτητο να εξηγήσουμε τι ακριβώς δηλώνει το κάθε tag στα xml του dataset μας. Το super tag περικλύει όλα τα υπόλοιπα tags του αρχείου, οπότε δεν έχει κάποια ιδιαίτερη σημασία. Το ίδιο και το meta tag, μέσα στο οποίο βρίσκονται πληροφορίες που δεν έχουν να κάνουμε με την μουσική σημειολογία του κομματιού, όπως ο τίτλος του και ο καλλιτέχνης. Υπάρχει όμως το mode tag το οποίο είναι απαραίτητο καθώς υποδηλώνει τον τρόπο στον οποίο είναι γραμμένο το κομμάτι. Σε περίπτωση που δεν υπάρχει το συγκεκριμένο tag εννοείται ότι ο τρόπος του κομματιού είναι ο μείζον / ιωνικός τρόπος. Τα tags που αναφέρονται στον ρυθμό του κομματιού όπως το beats in measure tag δεν το χρειαστήκαμε καθώς παρακάτω θα δούμε πως υπάρχουν tags που μας δίνουν ακριβή πληροφορία για τη διάρκεια κάθε νότας. Εκτός αυτού δεν αναπαραστάθηκαν με κάποιον τρόπο τα μέτρα (bars) του κομματιού καθώς δεν επηρεάζουν το τελικό αποτέλεσμα.

Μετά το κλείσιμο του meta tag υπάρχει το sections tag το οποίο χωρίζει το κομμάτι σε τομείς όπως verse (κουπλέ), chorus (ρεφραίν), bridge (γέφυρα) και άλλα. Το κάθε section χωρίζεται σε segments από το segment tag. Τα segments είναι απλά κομμάτια του section και δεν έχουν κάποια ιδιαίτερη σημασία, ούτε δίνουν κάποια μουσική πληροφορία. Παρ’ όλα αυτά τα section και segment tags οργανώνουν τη δομή του κομματιού και του xml αρχείου γενικότερα.

Και τώρα φτάνουμε στο ουσιαστικό μέρος του xml όπου περιέχονται όλες οι πληροφορίες για τη δημιουργία των numpy arrays που θα χρησιμοποιήσαμε σαν είσοδο στα μοντέλα που θα εξετάστηκαν στα πλαίσια της συγκεκριμένης διπλωματικής εργασίας. Κάθε section περιέχει δύο tags στο επόμενο επίπεδο. Τα tags αυτά είναι τα notes και chords tags. Σε πολλά κομμάτια αντί για notes και chords υπήρχαν τα tags melody και harmony αντίστοιχα. Τα tags αυτά περιέχουν τα tags note και chord αντίστοιχα, όπου κάθε note και chord tag περιέχει όλες τις πληροφορίες της νότας ή της συγχορδίας στην οποία αναφέρεται.

Μελωδία

Στη συνέχεια φαίνονται όλα τα διάφορα tags που υπάρχουν μέσα σε ένα note tag. Ακολούθως θα γίνει σαφές πως δημιουργούμε τον πίνακα νοτών από τις πληροφορίες που περιέχουν τα xml αρχεία. Αρχικά να τονίσουμε ότι ήδη από το mode tag γνωρίζουμε το mode του κομματιού.

```
<note>
  <start_beat_abs>0</start_beat_abs>
  <start_measure>1</start_measure>
  <start_beat>1</start_beat>
  <note_length>1</note_length>
  <scale_degree>3</scale_degree>
  <octave>0</octave>
  <isrest>0</isrest>
  </note>
```

Τα τρία πρώτα tags που θα αναφερθούν παρέχουν πληροφορίες για την τονικότητα της νότας. Τα υπόλοιπα tags ορίζουν την χρονική αξία της νότας.

- scale_degree: Αναφέρεται στην βαθμίδα της νότας. Όπως είδαμε και στο κεφάλαιο 2.1. γνωρίζοντας τη βαθμίδα μιας νότας σε μία κλίμακα / τρόπο και την κλίμακα / τον τρόπο μπορούμε να βρούμε αυτή τη νότα. Αυτό το πετύχαμε δημιουργώντας python dictionaries για κάθε mode, συνολικά 21 (οι 7 φυσικοί τρόποι, 7 flat modes, 7 sharp modes). Για παράδειγμα για τον ιωνικό/μείζονα τρόπο το python dictionary θα είναι κάπως έτσι:

```
def ionian_dict(x):
return {
        'rest': NaN,
        '1': 0,
        '1s': 1, '2f': 1,
        '2': 2,
        '2s': 3, '3f': 3,
        '3': 4, '4f': 4,
        '4': 5,
        '4s': 6, '5f': 6,
        '5': 7,
        '5s': 8, '6f': 8,
        '6': 9,
```

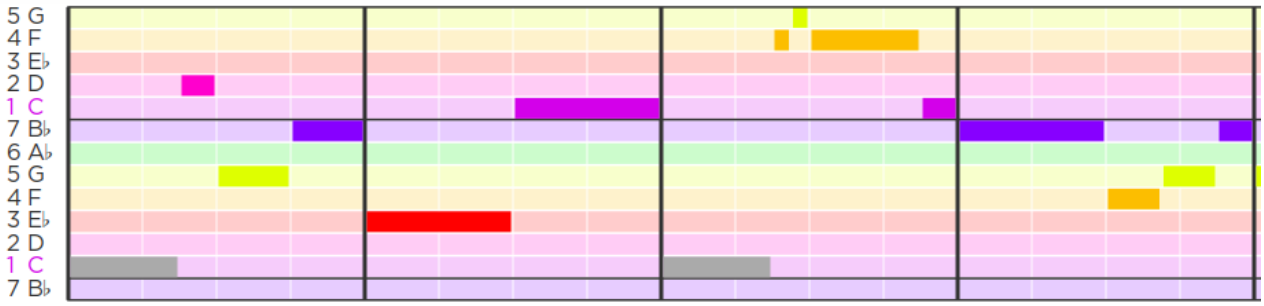
```
        '6s': 10, '7f': 10,
        '7': 11,
        '9': 14,
        '11': 17,
        '13': 21,
}[x]
```

Προφανώς οι νότες που αντιστοιχούν σε αριθμούς μεγαλύτερους του δώδεκα αναφέρονται ειτε στην νότα που προκύπτει από την πράξη ionian_dict(x) % 12 είτε στην νότα αυτή αλλά μία οκτάβα πάνω.

- octave: Αναφέρεται σε ποια οκτάβα ανήκει η νότα. Αν δύο νότες έχουν το ίδιο scale_degree αλλά η μία έχει octave 0 και η άλλη 1, τότε η δεύτερη νότα έχει απόσταση ακριβώς μία οκτάβα προς τα πάνω από την πρώτη. Να σημειωθεί σε αυτό το σημείο ότι οι κτάβες χρησιμοποιήθηκαν μόνο για την εκπαίδευση του Coconet, καθώς μόνο σε αυτό το εύρος τιμών των νοτών ήταν 256. Δεδομένου ότι κάθε οκτάβα έχει 12 νότες τότε συμπεραίνουμε ότι οι 256 νότες καλύπτουν 21 οκτάβες και 4 νότες επιπλέον. Άρα για να ορίσουμε ακριβώς σε ποια νότα αναφέρεται το note tag του xml αρχείου χρειάζεται να λάβουμε υπόψη και το scale_degree και το octave tag. Στο sequence to sequence μοντέλο χρηιμοποιήσαμε εύρος νοτών μία οκτάβα οπότε το συγκεκριμένο tag δεν παίζει κανένα ρόλο
- isrest: Δηλώνει εάν η νότα στην οποία αναφέρεται το note tag είναι στην πραγματικότητα παύση. Αν έχει την τιμή 1 δηλώνει ότι είναι παύση, η τιμή 0 δηλώνει ότι δεν είναι παύση. Αν το isrest παίρνει την τιμή ένα, τα tags note και octave αγνοούνται. Στην περίπτωση του Coconet, όταν υπάρχει παύση, η τιμή της νότας γίενται NaN ενώ στην περίπτωση του sequence to sequence μοντέλου όπου έχουμε one hot enocoding, σε περίπτωση παύσης ολόκληρο το διάνυσμα παίρνει μηδενικές τιμές.
- start_beat_abs: Το tag αυτό δηλώνει την χρονική στιγμή που αρχίζει να ηχεί η νότα. Η χρονική στιγμή 0 αναφέρεται στην αρχή του κάθε segment. Η χρονική στιγμή 1 δηλώνει ότι η νότα αρχίζει να ηχεί 1/4 μετά την αρχή του segment.
- note_length: Το tag note_length δηλώνει τη χρονική διάρκεια μίας νότας. Μαζί με το tag start_beat_abs δίνουν όλες τις πληροφορίες για τη χρονική αξία κάθε νότας και τη θέση τους χρονικά στο κομμάτι. Όπως και στην περίπτωση του start_beat_abs μία μονάδα αντιστοιχεί στην χρονική αξία του 1/4.

Έχοντας μαζέψει όλες αυτές τις πληροφορίες είμαστε σε θέση να κατασκευάσουμε τον πίνακα νοτών. Θα μιλήσουμε ξεχωριστά για το κάθε μοντέλο καθώς η είσοδος τους διαφέρει. Αρχικά πρέπει να τονίσουμε ότι η μικρότερη χρονική αξία που μπορεί να έχει μία νότα είναι αυτού του 1/16 και επίσης δεν υπάρχει κάποιο σύμβολο που να δηλώνει ότι δύο ίδιες νότες που ακούγονται η μία μετά την άλλη είναι η ίδια ή διαφορετική νότα. Για αυτό θα θεωρήσουμε ότι οποιεσδήποτε ίδιες νότες ακούγονται η μία μετά την άλλη συνιστούν την ίδια νότα με χρονική αξία το άθροισμα των αξιών των δύο νοτών. Αυτό γενικοποιείται για οποιοδήποτε αριθμό ίδιων νοτών στη σειρά. Αφού η μικρότερη δυνατή αξία είναι το 1/16 θεωρούμε ότι μία νότα που εμφανίζεται στον πίνακα και πριν και μετά από αυτή υπάρχουν διαφορετικές νότες έχει αξία 1/16. Όταν υπάρχουν στον πίνακα δύο ίδιες νότες στη σειρά, ενώ πριν από την πρώτη και μετά τη δεύτερη υπάρχουν διαφορετικές νότες, τις αντιλαμβανόμαστε ως μία νότα με αξία 1/8 και ούτω καθ’ εξής.

Παρακάτω βλέπουμε δύο παραδείγματα για το πως μετασχηματίζουμε τη μελωδία ενός κομματιού από το HookTheory σε numpy arrays για καθένα από τα μοντέλα που εξετάζουμε στη συγκεκριμένη διπλωματική εργασία:



Coconet

Διάνυσμα	**Νότα**	**Αξία**
[**nan** nan nan nan nan nan	Παύση	6/16 = 3/8
74. 74.	Ρε / D	2/16 = 1/8
67. 67. 67. 67.	Σολ / G	4/16 = 1/4
70. 70. 70. 70.	Σι ♭ / B♭	4/16 = 1/4
63. 63. 63. 63. 63. 63. 63. 63.	Μι ♭ / E♭	8/16 = 1/2
72. 72. 72. 72. 72. 72. 72. 72.	Ντο / C	8/16 = 1/2
nan nan nan nan nan nan	Παύση	6/16 = 3/8
77.	Φα / F	1/16
79.	Σολ / G	1/16
77. 77. 77. 77. 77. 77.	Φα / F	6/16 = 3/8
72. 72.	Ντο / C	2/16 = 1/8
70. 70. 70. 70. 70. 70. 70. 70.	Σι ♭/ B♭	8/16 = 1/2
65. 65. 65.	Φα / F	3/16
67. 67. 67.	Σολ / G	3/16
70. 70.]	Σι ♭ / B♭	2/16 = 1/8

Sequence to Sequence

Διάνυσμα	**Νότα**	**Αξία**
[[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]	Παύση	6/16 = 3/8
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0.]	Ρε / D	2/16 = 1/8
[0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0.]	Σολ / G	4/16 = 1/4

[0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0.]	Σι ♭ / B♭	4/16 = 1/4
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0.]		
[0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0.]	Μι ♭ / E♭	8/16 = 1/2
[0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0.]		
[1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]	Ντο / C	8/16 = 1/2
[1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]	Παύση	6/16 = 3/8
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0.]	Φα / F	1/16
[0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0.]	Σολ / G	1/16
[0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0.]	Φα / F	6/16 = 3/8
[0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0.]		
[1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]	Ντο / C	2/16 = 1/8
[1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0.]	Σι ♭/ B♭	8/16 = 1/2
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0.]		
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0.]		
[0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0.]	Φα / F	3/16
[0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0.]		

[0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0.] Σολ / G 3/16
[0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0.] Σι ♭ / B♭ 2/16 = 1/8
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0.]]

Αρμονία

Αντίστοιχη μεθοδολογία όσον αφορά τη χρονική αξία ακολουθήσαμε και στην περίπτωση των συγχορδιών που ορίζονται μέσα στο chords ή harmony tag. Όμως στις συγχορδίες υπάρχουν διαφορετικά tags για να δηλώσουν την τονικότητα και το είδος της συγχορδίας. Τα tags αναλυτικά και η μεθοδολογία κατασκευής των numpy arrays από αυτά παρουσιάζονται παρακάτω:

```
<chord>
  <sd>1</sd>
  <fb></fb>
  <sec></sec>
  <emb>add9</emb>
  <sus></sus>
  <pedal></pedal>
  <alternate></alternate>
  <borrowed></borrowed>
  <chord_duration>1</chord_duration>
  <start_measure>1</start_measure>
  <start_beat>1</start_beat>
  <start_beat_abs>0</start_beat_abs>
  <isrest>0</isrest>
</chord>
```

Όπως και στην περίπτωση της μελωδίας, έτσι και εδώ τα πρώτα 5 tags αφορούν την τονικότητα και το είδος της συγχορδίας στην οποία αναφέρεται το κάθε note tag ενώ τα τελευταία δύο δίνουν τη χρονική αξία της συγχορδίας και τη θέση της στον χρονικό άξονα του κομματιού.

- **Sd**: Το αντίστοιχο scale_degree που είδαμε και στη μελωδία. Δηλώνει σε ποια βαθμίδα της κλίμακας ανήκει η συγχορδία στην οποία αναφέρεται το chord tag. Η κλίμακα της νότας είναι γνωστή από το mode tag που είδαμε προηγουμένως στα meta info. Χρησιμοποιήθηκε αντίστοιχη μεθοδολογία με τις νότες και για τον ορισμό της κάθε συγχορδίας. Δημιουργήθηκαν python dictonaries για κάθε natural, flat και sharp mode. Για παράδειγμα ένα python dictionary για τον ορισμό μίας συγχορδίας θα έμοιαζε κάπως έτσι:

```
def ionian_chord_dict(x):
return {
        'rest': [NaN, NaN, NaN],
        '1': [7, 4, 0],
        '2': [9, 5, 2],
```

```
        '3': [11, 7, 4],
        '4': [12, 9, 5],
        '5': [14, 11, 7],
        '6': [16, 12, 9],
        '7': [17, 14, 11],
    }[x]
```

Προφανώς οι νότες που αντιστοιχούν σε αριθμούς μεγαλύτερους του δώδεκα αναφέρονται είτε στην νότα που προκύπτει από την πράξη ionian_chord_dict(x) % 12 είτε στην νότα αυτή αλλά μία οκτάβα πάνω.

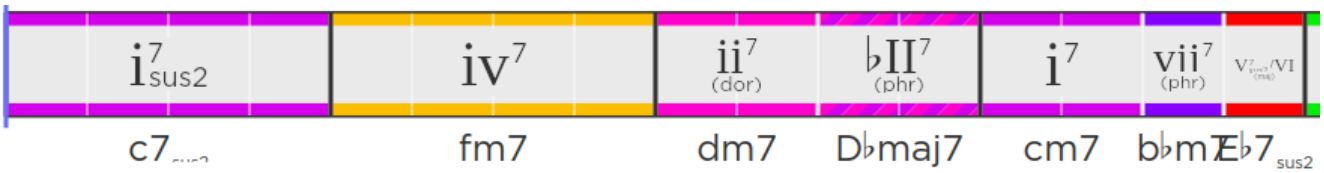
- **fb**: Το tag αυτό αναφέρεται σε επιπλέον νότες (7η, 9η, 11η, 13η) ή αναστροφές που υπάρχουν στην συγχορδία. Οι επιπλέον νότες δηλώνονται με τον αντίστοιχο αριθμό βαθμίδας (7, 9, 11, 13) ενώ ο συμβολισμός των αναστροφών είναι ο εξής:
    - 1η αναστροφή: 6
    - 2η αναστροφή: 64
    - 1η αναστροφή με έβδομη: 65
    - 2η αναστροφή με έβδομη: 43
    - 3η αναστροφή με έβδομη: 42

- **Sec**: Το συγκεκριμένο tag εκφράζει αν η συγχορδία αυτή αποτελεί δευτερεύουσα συγχορδία. Περισσότερες πληροφορίες για τις δευτερεύουσες συγχορδίες στο κεφάλαιο 2.1.. Οι πιθανές τιμές του sec tag είναι όλοι οι ακέραιοι από 1 έως 7. Όταν το sec tag δεν είναι κενό τότε υπολογίζουμε λαμβάνοντας υπόψη την τιμή του sd tag υπολογίζουμε την τονικότητα της συγχορδίας. Το sd tag παίρνει τιμές 4, 5 και 7 όπου η καθεμία δηλώνει την 4η είζονα, 5η μείζονα ή 7η ελαττωμένη της συγχορδίας της βαθμίδας δηλώνει το sec tag. Για παράδειγμα αν η κλίμακα του κομματιού είναι η Ντο μείζονα και τα sd και sec tags έχουν τις τιμές 5 ακι 5 αντίστοιχα, υπολογίζουμε την συγχορδία της πέμπτης βαθμίδας της Ντο μείζονας κλίμακας, η οποία είναι η σολ μείζονα και στη συνέχεια υπολογίζουμε τη συγχορδία της πέμπτης βαθμίδας της σολ μείζονας κλίμακας που είναι η ρε μείζονα συγχορδία.
- **sus**: Suspended chords είναι οι συγχορδίες που δεν περιέχουν την τρίτη (μείζονα ή ελάσσονα) παραλείπεται και αντικαθίσταται, συνήθως, με μία τέλεια τέταρτη ή μία μείζονα δεύτερη, ανάλογα με τον αριθμό που υπάρχει στο sus tag. Οι πιθανές τιμές του tag είναι 2,4 και 24 που δηλώνουν ότι την θέση της τρίτης παίρνουν η δεύτερη, η τέταρτη ή η δεύτερη και η τέταρτη αντίστοιχα. Σημειώνεται ότι suspended chords χρησιμοποιήθηκαν μόνο κατά την εκπαίδευση του Coconet καθώς στο sequence to sequence μοντέλο χρησιμοποιήθηκαν μόνο μείζονες, ελάσσονες και ελαττωμένες συγχορδίες στην απλή τους μορφή.
- **Emb**: Αναφέρεται σε επιπλέον νότες που μπορεί να υπάρχουν στην συγχορδία. Οι πιθανές τιμέ αυτού του tag είναι οι add9, add11, add13 που δηλώνουν ότι στην συγχορδία πρέπει να προστεθούν οι νότες της 9ης, της 11ης ή της 13ης βαθμίδας αντίστοιχα.
- **Borrowed**: Το συγκεκριμένο tag δηλώνει εάν η συγχορδία στην οποία αναφέρεται ανήκει στον τρόπο που έχει δηλωθεί στα meta info του κομματιού ή πρόκειται για δανεισμό από κάποιον άλλο παράλληλο τρόπο. Πληροφορίες για το τι είναι παράλληλοι τρόποι και τι σημαίνει δανεισμός συγχορδίας από παράλληλο τρόπο υπάρχουν στο κεφάλαιο 2.1. της διπλωματικής αυτής εργασίας. Οι πιθανές τιμές του συγκεκριμένου tag είναι όλοι οι ακέραιοι από -12 μέχρι 8, συνολικά 21. Όσοι και οι πιθανοί τρόποι από τους οποίους μπορεί να προέρχεται η συγχορδία. Ο αριθμός που αναφέρεται στο borrowed tag δηλώνει

ουσιαστικά τον αριθμό των υφέσεων που υπάρχουν στον συγκεκριμένο τρόπο, ενώ οι αρνητικοί αριθμοί αναφέρονται στον αριθμό των διέσεων.

- **Isrest**: Αντίστοιχα με τις νότες, και για τις συγχορδίες υπάρχει το isrest tag για να δηλώνει αν στο χρονικό διάστημα στο οποίο αναφέρεται το συγκεκριμένο tag ηχεί κάποια νότα ή υπάρχει παύση, δηλαδή δεν ηχεί καμία συγχορδία. Η τιμή 0 δηλώνει ότι ηχεί η συγχορδία που αναφέρεται στο sd tag ενώ η τιμή 1 σημαίνει ότι υπάρχει παύση και η τιμή στο sd tag δεν έχει κάποια πρακτική σημασία.
- **start_beat_abs**: Δηλώνει την την χρονική στιγμή που αρχίζει να ηχεί η συγχορδία στην οποία αναφέρεται το συγκεκριμένο chord tag. Η χρονική στιγμή 0 αναφέρεται στην αρχή του κάθε segment. Η χρονική στιγμή 1 δηλώνει ότι η συγχορδία αρχίζει να ηχεί 1/4 μετά την αρχή του segment.
- **chord_duration**: Το tag αυτό δηλώνει την χρονική διάρκεια για την οποία ηχεί η συγχορδία στην οποία αναφέρεται το συγκεκριμένο chord tag. Μαζί με το tag start_beat_abs δίνουν όλες τις πληροφορίες για τη χρονική διάρκεια κάθε συγχορδίας και τη χρονική της σειρά στο κομμάτι. Όπως και στην περίπτωση του start_beat_abs μία μονάδα αντιστοιχεί στην χρονική αξία του 1/4. Άρα για να ηχεί μία συγχορδία για ένα ολόκληρο μέτρο θα πρέπει να έχει chord_duration ίσο με 4 αν το μέτρο του κομματιού είναι 4/4 και 3 να το μέτρο του κομματιού είναι 3/4 και ούτω καθ' εξής.

Παρακάτω βλέπουμε δύο παραδείγματα για το πως μετασχηματίζουμε την αρμονία ενός κομματιού από το HookTheory σε numpy arrays για καθένα από τα μοντέλα που εξετάζουμε στη συγκεκριμένη διπλωματική εργασία:



Coconet

Διάνυσμα	**Νότες**	**Συγχορδία**	**Αξία**
[**[58. 50. 48.]**	48 = Ντο / C	Cmin7sus2	16/16 = 1
[58. 50. 48.]	50 = Ρε / D		
[58. 50. 48.]	58 = Σι ♭ / B♭		
[58. 50. 48.]			
[58. 50. 48.]			
[58. 50. 48.]			
[58. 50. 48.]			
[58. 50. 48.]			
[58. 50. 48.]			
[58. 50. 48.]			
[58. 50. 48.]			
[58. 50. 48.]			
[58. 50. 48.]			
[58. 50. 48.]			

[58. 50. 48.]			
[58. 50. 48.]			
[51. 44. 41.]	41 = Φα / F	Fmin7	16/16 = 1
[51. 44. 41.]	44 = Λα ♭ / A♭		
[51. 44. 41.]	51 = Μι ♭ / E♭		
[51. 44. 41.]			
[51. 44. 41.]			
[51. 44. 41.]			
[51. 44. 41.]			
[51. 44. 41.]			
[51. 44. 41.]			
[51. 44. 41.]			
[51. 44. 41.]			
[51. 44. 41.]			
[51. 44. 41.]			
[51. 44. 41.]			
[51. 44. 41.]			
[48. 41. 38.]	38 = Ρε / D	Dmin	8/16 = 1/2
[48. 41. 38.]	41 = Φα / F		
[48. 41. 38.]	48 = Ντο / C		
[48. 41. 38.]			
[48. 41. 38.]			
[48. 41. 38.]			
[48. 41. 38.]			
[48. 41. 38.]			
[48. 41. 37.]	37 = Ρε ♭ / D♭	D♭maj7	8/16 = 1/2
[48. 41. 37.]	41 = Φα / F		
[48. 41. 37.]	48 = Ντο / C		
[48. 41. 37.]			
[48. 41. 37.]			
[48. 41. 37.]			
[48. 41. 37.]			
[48. 41. 37.]			
[58. 51. 48.]	48 = Ντο / C	Cmin7	8/16 = 1/2
[58. 51. 48.]	51 = Μι ♭ / E♭		
[58. 51. 48.]	58 = Σι ♭ / B♭		
[58. 51. 48.]			
[58. 51. 48.]			
[58. 51. 48.]			
[58. 51. 48.]			
[58. 51. 48.]			
[56. 49. 46.]	46 = Σι ♭ / B♭	B♭min7	4/16 = 1/4
[56. 49. 46.]	49 = Ρε ♭ / D♭		
[56. 49. 46.]	56 = Λα ♭ / A♭		
[56. 49. 46.]			
[61. 59. 51.]	51 = Μι ♭ / E♭	E♭7sus2	4/16 = 1/4
[61. 59. 51.]	59 = Σι / B		
[61. 59. 51.]	63 = Ρε ♭ / D♭		

[61. 59. 51.]]

Sequence to Sequence

Multi-class

Λόγω του μεγέθους του πίνακα (sequence_length, 36) δεν θα τον παραθέσουμε αυτούσιο. Η μορφή του είναι αρκετά απλή. Κάθε γραμμή αποτελείται από 35 μηδενικά και 1 ένα. Η θέση του 1 δηλώνει την συγχορδία που ηχεί εκείνη τη χρονική στιγμή. Οι πρώτες 12 θέσεις δηλώνουν τις 12 ματζόρε συγχορδίες, οι επόμενες 12 τις 12 μινόρε συγχορδίες και οι 12 τελευταίες τις 12 ελαττωμένες . Η χρονική αξία κάθε συγχορδίας δηλώνεται με το ίδιο τρόπο όπως και στην περίπτωση του Coconet.

Multi-label

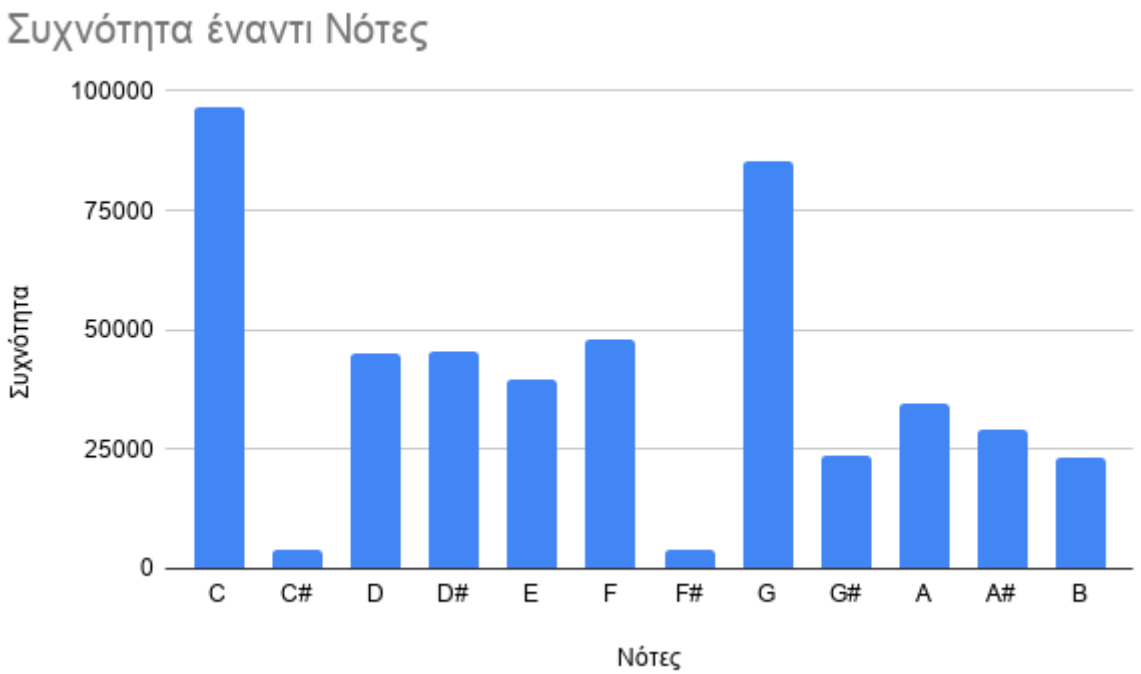
Διάνυσμα	**1-Δείκτες**	**Νότες**	**Συγχορδία**
[**[1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]**	0	Ντο / C	Cmin7sus2
[1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]	2	Ρε / D	
[1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]	7	Σολ / G	
[1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]	10	Σι ♭ / B♭	
[1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]			
[1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]			
[1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]			
[1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]			
[1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]			
[1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]			
[1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]			
[1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]			
[1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]			
[1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]			
[1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]			
[1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]			
[1. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]	0	Ντο / C	Fmin7
[1. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]	3	Μι ♭ / E♭	
[1. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]	5	Φα / F	
[1. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]	8	Λα ♭ / A♭	
[1. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]			
[1. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]			
[1. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]			
[1. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]			
[1. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]			
[1. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]			
[1. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]			
[1. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]			
[1. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]			
[1. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]			
[1. 0. 0. 1. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]			
[1. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 0.]	0	Ντο / C	Dmin

[1. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 0.]	2	Ρε / D	
[1. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 0.]	5	Φα / F	
[1. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 0.]	9	Λα / A	
[1. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 0.]			
[1. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 0.]			
[1. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 0.]			
[1. 0. 1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 0. 0.]			
[1. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]	0	Ντο / C	D♭maj7
[1. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]	1	Ρε ♭ / D♭	
[1. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]	5	Φα / F	
[1. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]	8	Λα ♭ / A♭	
[1. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]			
[1. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]			
[1. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]			
[1. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 0. 0.]			
[1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]	0	Ντο / C	Cmin7
[1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]	3	Μι ♭ / E♭	
[1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]	7	Σολ / G	
[1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]	10	Σι ♭ / B♭	
[1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]			
[1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]			
[1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]			
[1. 0. 0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0.]			
[0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 1. 0. 0. 0.]	1	Ρε ♭ / D♭	B♭min7
[0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 1. 0. 0. 0.]	5	Φα / F	
[0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 1. 0. 0. 0.]	8	Λα ♭ / A♭	
[0. 1. 0. 0. 0. 1. 0. 0. 1. 0. 1. 0. 0. 0.]	10	Σι ♭ / B♭	
[0. 1. 0. 1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 0.]	1	Ρε ♭ / D♭	E♭7sus2
[0. 1. 0. 1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 0.]	3	Μι ♭ / E♭	
[0. 1. 0. 1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 0.]	5	Φα / F	
[0. 1. 0. 1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 0.]]	10	Σι ♭ / B♭	

3.4. Πόλωση dataset

Το dataset που κατεβάσαμε από το hooktheory αντιστοιχεί σε πραγματικά τραγούδια. Αυτό έχει ως αποτέλεσμα τα δεδομένα του dataset να μην είναι εξισορροπημένα. Πιο συγκεκριμένα, ο αριθμός φορών που εμφανίζεται μια συγχορδία μπορεί να διαφέρει πολύ σε σχέση με τον αριθμό που εμφανίζεται κάποια άλλη συγχορδία. Αυτό δημιουργεί πρόβλημα στη διαδικασία εκπαίδευσης μιας και το δίκτυο πολώνεται στις πιο συνηθισμένες συγχορδίες ή νότες και όπως είναι λογικό οι προβλέψεις που θα κάνει τείνουν να κατευθύνονται προς αυτές τις συγχορδίες ή νότες. Αυτό το πρόβλημα μπορεί να αντιμετωπιστεί με τη λεγόμενη διαδικασία εξισορρόπησης κατά την οποία αφαιρούνται διαδοχικά από το dataset δεδομένα από τις ποιο συνήθεις κλάσεις με σκοπό να εξαλειφθούν οι ανομοιομορφίες που εμφανίζονται στο dataset. Η μείωση όμως των δεδομένων την στιγμή που αυτά είναι περιορισμένα επιφέρει ένα άλλο κόστος, τον κίνδυνο εμφάνισης υπερεκπαίδευσης του μοντέλου.

Στα dataset που χρησιμοποιήσαμε για την εκπαίδευση των μοντέλων εμφανίζεται το πρόβλημα της πόλωσης των δεδομένων καθώς αφενός κάποιες νότες ή συγχορδίες δεν ανήκουν στο κλειδί του κάθε κομματιού και αφετέρου τα περισσότερα κομμάτια τείνουν να χρησιμοποιούν έναν περιορισμένο και συγκεκριμένο αριθμό συγχορδιών. Τα συμπεράσματα του φαινομένου της πόλωσης των δεδομένων θα παρατεθούν στα κεφάλαια αξιολόγησης των μοντέλων.



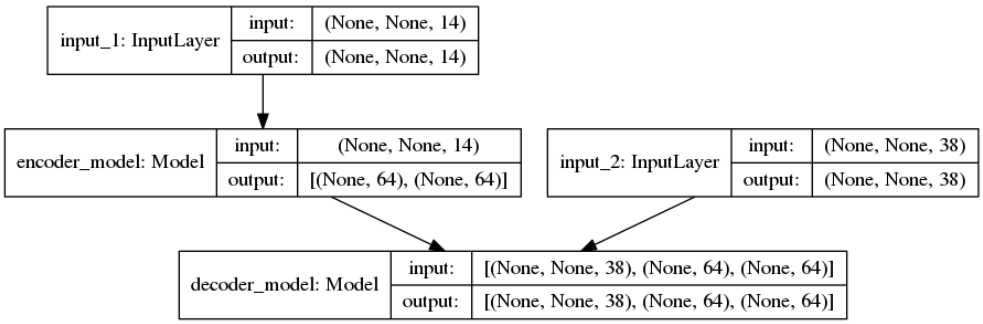
4. Εναρμόνιση μελωδίας με αναδρομικά νευρωνικά δίκτυα (RNN)

4.1. Αρχιτεκτονική δικτύου

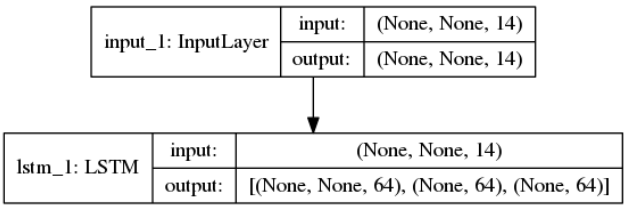
Για την υλοποίηση ενός συστήματος πρόβλεψης συγχορδιών με βάση μια μελωδία εισόδου επιλέξαμε το μονέλο Sequence To Sequence (γνωστό και ως μοντέλο κωδικοποιητή – αποκωδικοποιητή). Το μοντέλο αυτό έχει γνωρίσει σημαντική επιτυχία σε εφαρμογές επεξεργασίας φυσικής γλώσσας, αυτόματης μετάφρασης και γενικά είναι πολύ διαδεδομένο σε εφαρμογές που εμπεριέχουν την παραγωγή ακολουθιών βασισμένων σε ακολουθίες εισόδου. Θεωρώντας μια μελωδία μια αλληλουχία νοτών και την αρμονία σαν μια ακολουθία συγχορδιών, η επιλογή αυτού του μοντέλου ήταν εντελώς φυσιολογική.

4.1.1. Κωδικοποιητής – Αποκωδικοποιητής – απλά LSTM

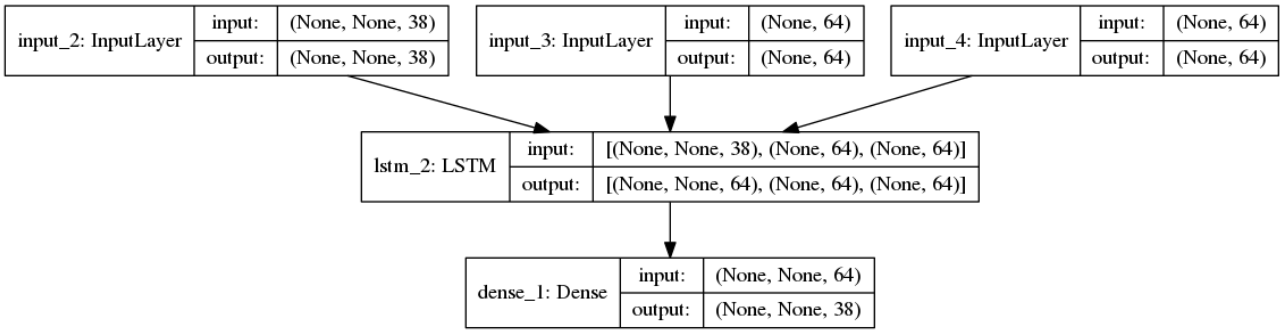
Αρχικά δοκιμάσαμε την παρακάτω αρχιτεκτονική του μοντέλου Sequence To Sequence η οποία είναι η πιο απλή που εξετάσαμε στα πλαίσια της συγκεκριμένης διπλωματικής εργασίας:



όπου τα επίπεδα ipnut_1 και input_2 είναι τα επίπεδα εισόδου του κωδικοποιητή και του αποκωδικοποιητή αντίστοιχα. Συνολικά ο κωδικοποιητής αποτελείται από ένα επίπεδο LSTM κια έχει την εξής μορφή



Το διάγραμμα του αποκωδικοποιητή είναι το παρακάτω:



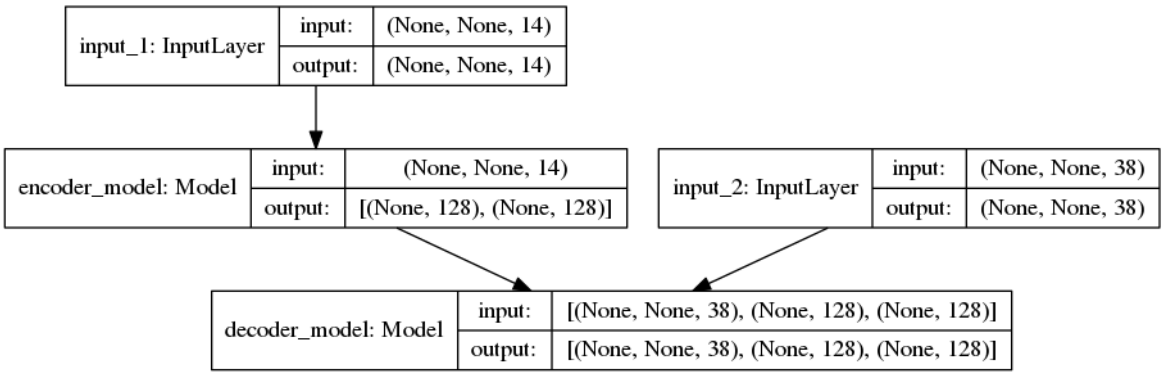
Όπως βλέπουμε ο αποκωδικοποιητής αποτελείται από ένα επίπεδο LSTM και ένα fully-connected (dense) επίπεδο. Από το συνολικό διάγραμμα του μοντέλο βλέπουμε ότι 2 από τις συνολικά 3 εξόδους του κωδικοποιητή τροφοδοτούνται στον αποκωδικοποιητή. Οι έξοδοι αυτοί είναι το hidden state και το cell state του κωδικοποιητή. Τα διανύσματα αυτά χρησιμοποιούνται σαν initial states του αποκωδικοποιητή. Η τρίτη είσοδος του αποκωδικοποιητή (με διαστάσεις None, None, 38) είναι τα διανύσματα εξόδου που πρέπει να προβλέψει σύμφωνα με την τεχνική εκπαίδευσης teacher forcing. Η έξοδος του αποκωδικοποιητή έχει διαστάσεις (None, None, 38) και είναι πανομοιότυπη με την είσοδο του, με τι διαφορά ότι τα διανύσματα είναι μετατοπισμένα ένα timestep μπροστά.

Η εκπαίδευση γίνεται ως εξής: Ο κωδικοποιητής δέχεται το διάνυσμα εισόδου και παράγει το κωδικοποιημένο διάνυσμα περιεχομένου (context vector). Τα διανύσματα κατάστασης του αποκωδικοποιητή αρχικοποιούνται με τα τελικά διανύσματα εισόδου του κωδικοποιητή. Ο αποκωδικοποιητής δέχεται σαν είσοδο τον χαρακτήρα αρχής τραγουδιού και προσπαθεί να προβλέψει την πρώτη συγχορδία. Μετά την πρόβλεψη γίνεται σύγκριση με την πραγματική συγχορδία και ξεκινά η διαδικασία του back propagation. Τέλος, ο αποκωδικοποιητής δέχεται σαν επόμενη είσοδο την πραγματική έξοδο του προηγούμενου timestep, σύμφωνα με την τεχνική teacher forcing.

4.1.2. Κωδικοποιητής – Αποκωδικοποιητής – αμφίδρομα LSTM

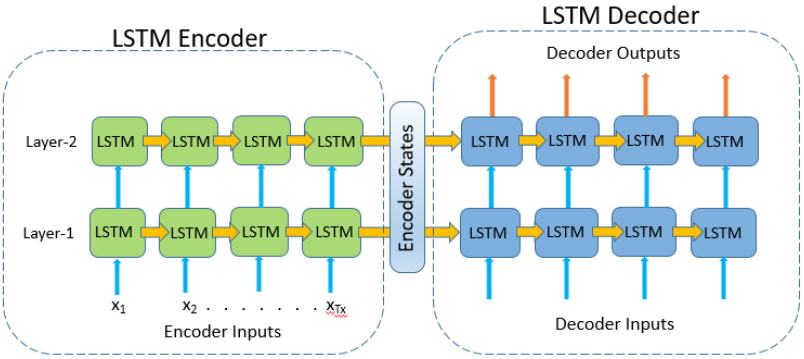
Σε αυτή την αρχιτεκτονική, οι διαφορές με την προηγούμενη που εξετάστηκε είναι ελάχιστες. Πιο συγκεκριμένα σε αυτή την αρχιτεκτονική χρησιμοποιήσαμε LSTM δίκτυα διπλής κατεύθυνσης έχοντας υπόψη ότι μια συγχορδία δε εξαρτάται αποκλειστικά από τις προηγούμενες χρονικά συγχορδίες αλλά πολύ συχνά προσδιορίζεται σε μεγάλο βαθμό από τις μετέπειτα συγχορδίες ενός μουσικού κομματιού.

Το διάγραμμα του συνολικού μοντέλου φαίνεται στην παρακάτω εικόνα. Η μόνη διαφορά με το αρχικό μοντέλο είναι η αλλαγή του αριθμού των recurrent units του LSTM από 64 σε 128, δηλαδή ο διπλασιασμός τους λόγω της προσθήκης του LSTM αντίθετης κατεύθυνσης.

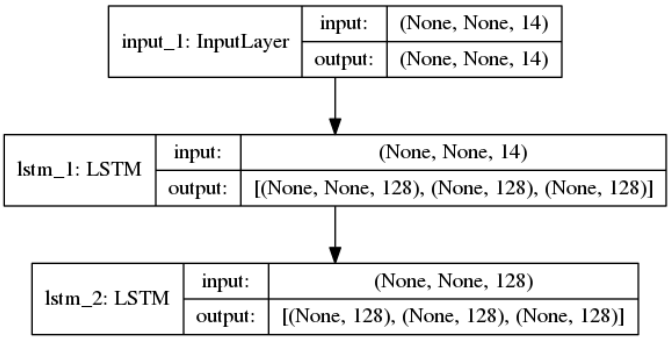


4.1.3. Κωδικοποιητής – Αποκωδικοποιητής – 2 επίπεδα απλών LSTM

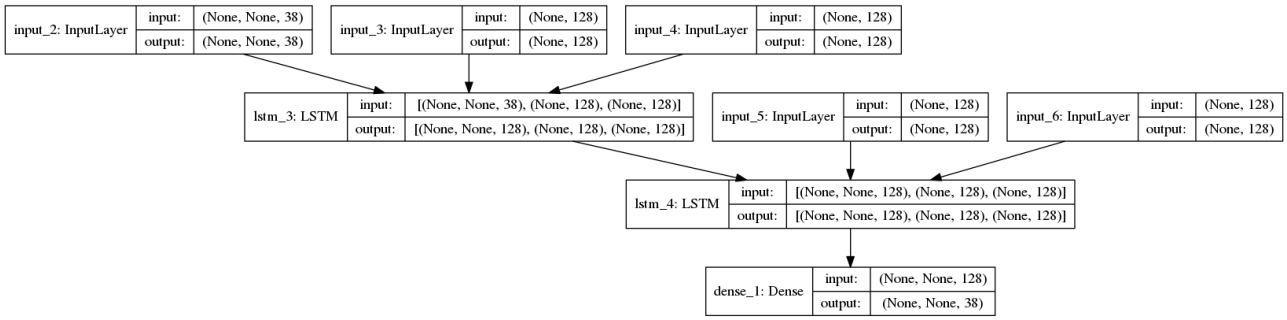
Σε αυτή την αρχιτεκτονική αυξάνουμε το βάθος του δικτύου προσθέτοντας ένα επιπέον επίπεδο LSTM. Υπάρχουν αρκετές θεωρήσεις για την συγκεκριμένη αρχιτεκτονική και για τον αριθμό επίπεδων αναδρομικών δικτύων στον κωδικοποιητή και στον αποκωδικοποιητή. Εμείς επιλέξαμε να προσθέσουμε από ένα επίπεδο αναδρομικών δικτύων σε κωδικοποιητή και αποκωδικοποιητή.



Στον κωδικοποιητή η σύνδεση των δύο επιπέδων γίνεται ως εξής: Η είσοδος τροφοδοτείται στο LSTM του πρώτου επιπέδου. Τα διανύσματα κατάστασης (hidden state και cell state) του πρώτου επιπέδου τροφοδοτείται στην είσοδο του πρώτου LSTM του ποκωδικοποιητή, ενώ η έξοδος (hidden state για κάθε timestep) τροφοδοτείται στο επόμενο επίπεδο του κωδικοποιητή. Το δεύτερο επίπεδο του κωδικοποιητή, με τη σειρά του, τροφοδοτεί τα διανύσματα κατάστασης του στο LSTM του δεύτερου επιπέδου του αποκωδικοποιητή.



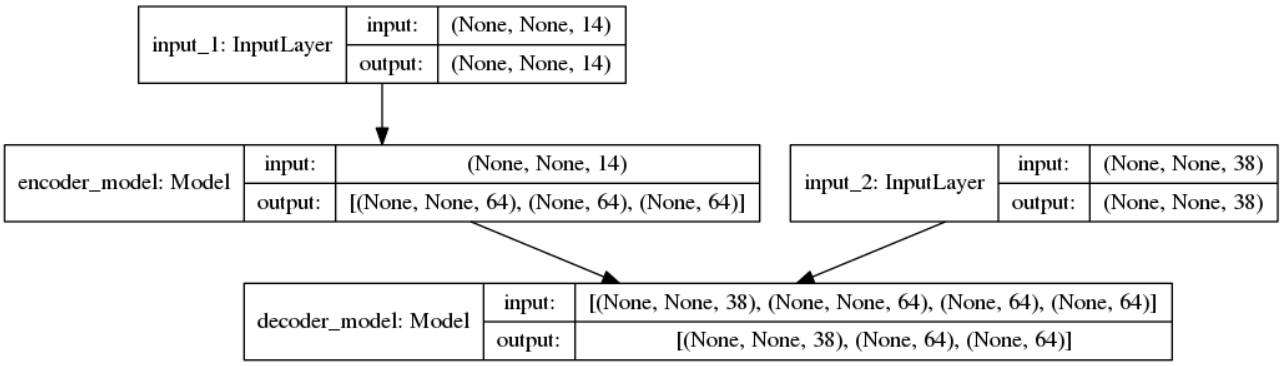
Στον αποκωδικοποιητή το πρώτο επίπεδο αρχικοποιεί τα διανύσματα κατάστασης του από τα αντίστοιχα διανύσματα εξόδου του πρώτου επιπέδου του κωδικοποιητή και τροφοδοτεί την έξοδο του στο επόμενο επίπεδο. Αυτό αρχικοποιεί αντίστοιχα τα διανύσματα κατάστασης του από τα διανύσματα κατάστασης του δεύτερου επιπέδου του κωδικοποιητή και δίνει σαν έξοδο την έξοδο του μοντέλου.



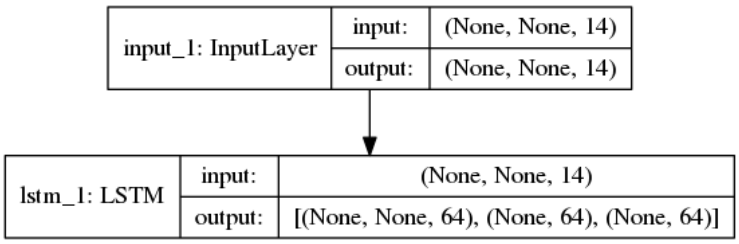
4.1.4. Κωδικοποιητής – Αποκωδικοποιητής – απλό LSTM – μηχανισμός προσοχής

Η συγκεκριμένη αρχιτεκτονική είναι πολύ διαδεδομένη τα τελευταία χρόνια καθώς έχει πετύχει αύξηση της αποτελεσματικότητας του μοντέλου κωδικοποιητή αποκωδικοποιητή. Η χρησιμότητα της είναι ακόμη μεγαλύτερη σε περιπτώσεις μεγάλων ακολουθιών εισόδου.

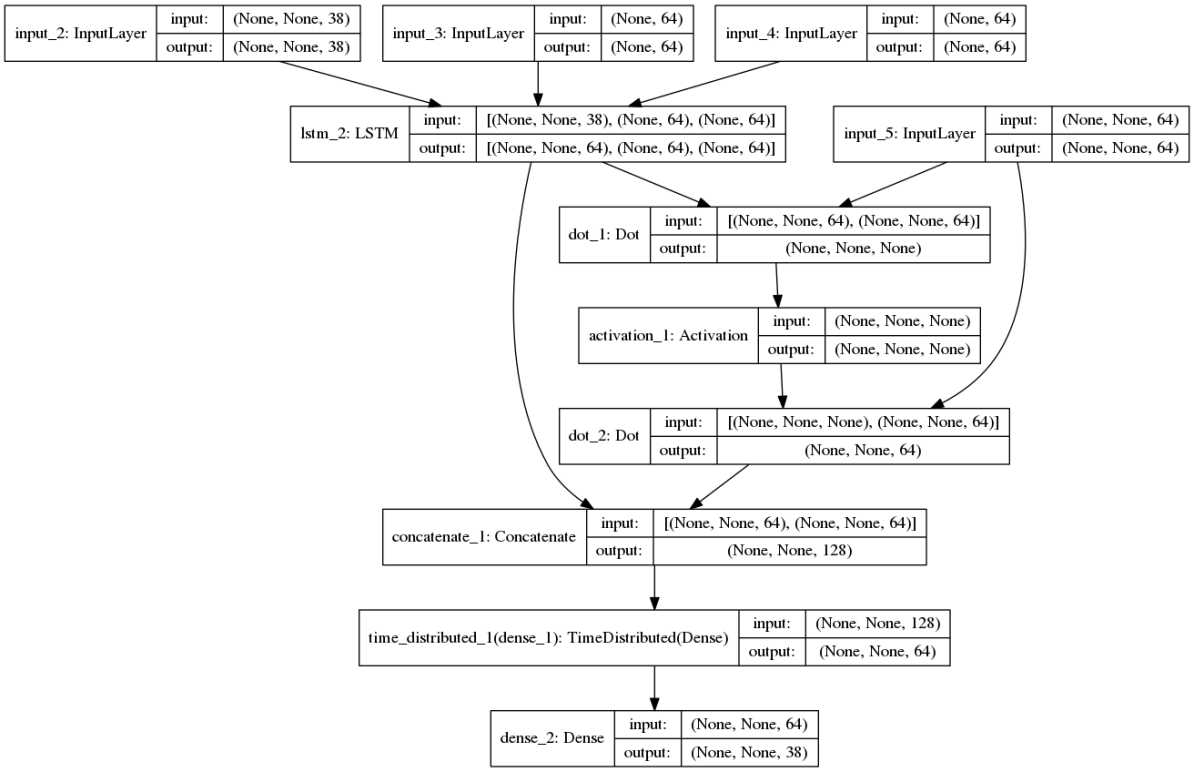
Αντίστοιχα με την αρχιτεκτονική του μοντέλου κωδικοποιητή – αποκωδικοποιητή που εξετάσαμε πρώτο, και αυτό το μοντέλο αποτελείται από ένα LSTM που λειτουργεί ως κωδικοποιητής και ένα LSTM που λειτουργεί σαν αποκωδικοποιητής.



Από το παραπάνω διάγραμμα παρατηρούμε την εξής διαφορά με το αρχικό απλό μοντέλο χωρίς τον μηχανισμό προσοχής: Ο κωδικοποιητής τροφοδοτεί τον αποκωδικοποιητή με ένα επιπλέον διάνυσμα διαστάσεων (None, None, Number_of_units). Το διάνυσμα αυτό περιέχει όλα τα διανύσματα εξόδου κάθε timestep του κωδικοποιητή. Στο απλό δίκτυο αγνοούσαμε πλήρως την έξοδο του κωδικοποιητή. Ρίχνοντας μια ματιά στο διάγραμμα του κωδικοποιητή είναι εμφανές ότι δεν παρουσιάσει κάποια διαφορά πέραν της επιπλέον εξόδου που αναφέραμε προηγουμένως.



Ο μηχανισμός προσοχής υλοποιείται στον αποκωδικοποιητή. Το διάγραμμα του αποκωδικοποιητή που περιλαμβάνει τον μηχανισμό προσοχής είναι το παρακάτω:



Ο αποκωδικοποιητής δέχεται σαν είσοδο τα 3 διανύσματα εξόδου του κωδικοποιητή (είσοδος πραγματικών συγχορδιών, hidden state, cell state). Τα διανύσματα hidden state και cell state του κωδικοποιητή χρησιμοποιούνται για την αρχικοποίηση των αντίστοιχων διανυσμάτων του αποκωδικοποιητή, όπως και στα προηγούμενα μοντέλα που εξετάσαμε. Ο μηχανισμός προσοχής περιγράφεται αναλυτικά στο κεφάλαιο “Μηχανισμός προσοχής”.

Οι παραπάνω αρχιτεκτονικές νευρωνικών δικτύων χρησιμοποιήθηκαν για την εκτέλεση των πειραμάτων για multi-class classification καθώς και για multi-label classification. Οι διαφορές που παρουσιάζουν τα μοντέλα για τα δύο αυτά πειράματα είναι οι εξής: Το επίπεδο εισόδου του αποκωδικοποιητή που τροφοδοτείται με τα δεδομένα εισόδου του αποκωδικοποιητή στην περίπτωση του multi-class classification προβλήματος έχει διαστάσεις *(None, None, 38)*. Στην περίπτωση του multi-label classification προβλήματος έχει διαστάσεις *(None, None, 14)*. Οι τιμές *None* υπάρχουν καθώς το μέγεθος του batch και του μήκους της ακολουθίας εισόδου και εξόδου δεν είναι γνωστά στο δίκτυο εκ των προτέρων.

4.2. Επιλογή υπερπαραμέτρων

4.2.1. Διαχωρισμός dataset

Εκπαιδεύσαμε το δίκτυο πάνω στα ίδια δεδομένα που κατεβάσαμε από το HookTheory.Για την εκπαίδευση των μοντέλων τα δεδομένα κάθε ενός dataset χωριστήκαν το κάθε ένα σε 3 διαφορετικά κομμάτια, ως είθισται:

- Το σύνολο εκπαίδευσης (train set) το οποίο χρησιμοποιείται για την εκπαίδευση του εκάστοτε μοντέλου. Αποτελεί το 70- με 80% του συνόλου των δεδομένων
- Το σύνολο ανάπτυξης (development set) το οποίο περιέχει δεδομένα από την ίδια κατανομή με το σύνολο εκπαίδευσης και χρησιμοποιείται για την αξιολόγηση του μοντέλου καθώς και για την διασπορά των δεδομένων. Αποτελεί το 10- 15% περίπου του συνόλου των δεδομένων
- Το σύνολο δοκιμής (test set), το οποίο περιέχει δεδομένα από διαφορετική κατανομή από το σύνολο εκπαίδευσης και χρησιμοποιείται για την τελική αξιολόγηση των μοντέλων καθώς και της πόλωσης των δεδομένων. Αποτελεί το 10- 15% περίπου του συνόλου των δεδομένων

Σε κάθε πείραμα γίνεται διαφορετικός και τυχαίος διαχωρισμός συνόλου εκπαίδευσης και συνόλου επικύρωσης για να μην προκύψουν αποτελέσματα που ρυθμίζονται σε μεγάλο βαθμό από τον αρχικό διαχωρισμό των συνόλων. Από την άλλη το σύνολο δοκιμής (test set) είναι το ίδιο σε όλες τις εκτελέσεις των αλγορίθμων εκπαίδευσης των μοντέλων. Η επιλογή αυτή έγινε για να μπορέσει να υπάρχει μια κοινή βάση σε όλα τα μοντέλα και να μπορέσει να γίνει μια δίκαιη σύγκριση μεταξύ τους.

4.2.2. Συναρτήσεις ενεργοποίησης

Η συνάρτηση ενεργοποίησης του τελικού dense layer στην περίπτωση του multi-class classification προβλήματος είναι η *softmax*. Αυτό έγινε καθώς θέλουμε να πετύχουμε μια πολυωνυμική κατανομή και το άθροισμα όλων των πιθανοτήτων που αντιστοιχούν στην κάθε κλάση να ισούται με 1. Η συνάρτησή αυτή χρησιμοποιείται σαν activation function στο τελευταίο επίπεδο των δικτύων που λύνουν προβλήματα ταξινόμησης (Logistic Regression). Η εξίσωση αυτή παρουσιάζεται παρακάτω:

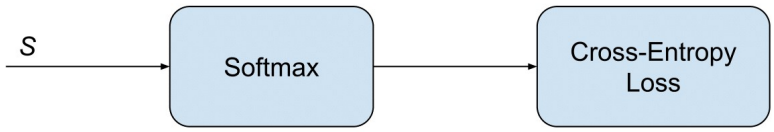
$$softmax = \frac{\left(\exp\left(y^{j}\right)\right)}{\sum_{j=1}^{N}\left(\exp\left(y^{j}\right)\right)}$$

Αντίθετα στην περίπτωση του multi-label classification προβλήματος θέλουμε η πιθανότητα να ανήκει μια νότα στη συγχορδία να μην επηρεάζει τις πιθανότητες των υπόλοιπων νοτών να ανήκουν και αυτές στην συγχορδία. Για αυτό το λόγο επιλέχθηκε η σιγμοειδής *(sigmoid, σ)* συνάρτηση ενεργοποίησης. Η συνάρτηση αυτή έχει αρκετά καλές ιδιότητες όπως ότι είναι μη-γραμμική, διαφορίσιμη καθώς και το πεδίο τιμών τις είναι (0,1) και έτσι μπορούν να εκφραστούν τα αποτελέσματα της ως πιθανότητες. Παρόλα αυτά έχει το μειονέκτημα ότι η μέγιστή τιμή της παραγώγου της *sigmoid* $\left(\frac{d\sigma}{dx}\right)$ είναι πολύ μικρή, μόλις 0.25. Αυτό δημιουργεί πρόβλημα κατά την εκπαίδευση βαθιών αρχιτεκτονικών, μιας και στον back propagation περνιέται μόνο ένα μικρό μέρος του σφάλματος στα πίσω επίπεδα, με αποτέλεσμα το δίκτυο τα πρώτα επίπεδά των δικτύων να εκπαιδεύονται πολύ πιο αργά (Vanishing Gradient) .

4.2.3. Συναρτήσεις κόστους

Χρησιμοποιήσαμε διαφορετικές συναρτήσεις κόστους για τα δύο πειράματα. Η συνάρτηση κόστους στο πλαίσιο των νευρωνικών δικτύων είναι η συνάρτηση που επιθυμούμε να ελαχιστοποιήσουμε

και ουσιαστικά εκφράζει την απόσταση των προβλεπόμενων τιμών από τις πραγματικές. Στο πρόβλημα του multi-class classification χρησιμοποιήσαμε τη συνάρτηση *categorical crossentropy.* Η επιλογή της συνάρτησης αυτής είναι στενά συνδεδεμένη με την επιλογή της συνάρτησης ενεργοποίησης. Με τη συνάρτηση αυτή εκπαιδεύουμε το δίκτυο ώστε για κάθε έξοδο να δημιουργεί μία κατανομή πιθανοτήτων για τις πιθανές κλάσεις τέτοια ώστε το άθροισμα των πιθανοτήτων αυτών να ισούται με 1.



$$f(s)_i = \frac{e^{s_i}}{\sum_j^C e^{s_j}} \quad CE = -\sum_i^C t_i log(f(s)_i)$$

Στην περίπτωση μας, όπου έχουμε χρησιμοποιήσει one-hot κωδικοποίηση για τα δεδομένα μας, μόνο μία κλάση συμμετέχει στον υπολογισμό της συνάρτησης απώλειας, καθώς μόνο για μία κλάση ο συντελεστής t_i δεν ισούται με μηδέν. Έτσι η συνάρτηση μπορεί να μετασχηματιστεί στην παρακάτω μορφή:

$$CE = -\log\left(\frac{e^{s_p}}{\sum_j^C e^{s_j}}\right)$$

Στην περίπτωση του προβλήματος multi-label classification σαν συνάρτηση απωλειών χρησιμοποιήσαμε την binary crossentropy. Η επιλογή αυτή βασίστηκε στο γεγονός ότι θέλουμε να εκπαιδεύσουμε το δίκτυο ώστε να παράγει τιμές πιθανοτήτων για κάθε κλάση που να μην επηρεάζει η μία την άλλη. Για κάθε κλάση η συνάρτηση υπολογίζεται ως εξής:

$$CE = -\sum_{i=1}^{C'=2} t_i \log(f(s_i)) = -t_1 \log(f(s_1)) - (1-t_1)\log(1-f(s_1))$$

Προφανώς και σε αυτήν την περίπτωση η επιλογή της συνάρτησης απωλειών έχει άμεση σχέση με την επιλογή συνάρτησης ενεργοποίησης καθώς για προβλήματα multi-label classification είναι πολύ συνήθης ο συνδυασμός αυτών των συναρτήσεων.

$$CE = -t_1 log(f(s_1)) - (1 - t_1) log(1 - f(s_1))$$

$$f(s_i) = \frac{1}{1 + e^{-s_i}}$$

4.2.4. Βελτιστοποιήτες

Η επιλογή του κατάλληλου βελτιστοποιήτη είναι μια πολύ σημαντική υπερπαράμετρος η οποίο επηρεάζει σημαντικά τον χρόνο εκπαίδευσης των δικτύων. Η επιλογή αυτού εξαρτάται περισσότερο από το είδος και την δομή του δικτύου και λιγότερο από τα δεδομένα εκπαίδευσης. Για την εκπαίδευση των μοντέλων επιλέξαμε τον βελτιστοποιητή Adam. Αυτή η επιλογή είναι και η πιο συνήθης επιλογή στην βιβλιογραφία, μιας και το μοντέλο κωδικοποιητή αποκωδικοποιητή έχει επιδείξει ταχύτατα αποτελέσματα εκπαίδευσης.

4.3. Ανάλυση αποτελεσμάτων

Multi-class Single-label Classification

Κωδικοποιητής – Αποκωδικοποιητής – απλά LSTM

Στην πρώτη φάση των πειραμάτων εκπαιδεύσαμε ένα απλό μοντέλο κωδικοποιητή αποκωδικοποιητή και συγκρίναμε τα αποτελέσματα της εκπαίδευσης για διαφορετικές τιμές του αριθμού αναδρομικών μονάδων των LSTM. Συγκεκριμένα οι τιμές που εξετάσαμε είναι οι 64, 128 και 256.

train					
		macro			micro
Αριθμός μονάδων	Dropout	f1	precision	recall	f1
64	0	0.177144874	0.302097627	0.162615208	0.442373427
128	0	0.380753606	0.493974021	0.339024619	0.504050548
128	0.2	0.358923043	0.479123943	0.310923498	0.478910942
128	0.4	0.338923402	0.460193289	0.309823983	0.457892011
256	0	0.676033161	0.729511840	0.637021463	0.706316667
256	0.2	0.610063509	0.739180520	0.544401808	0.683410711
256	0.4	0.438036607	0.664509821	0.378008114	0.563623457

test					
		macro			micro
Αριθμός μονάδων	Dropout	f1	precision	recall	f1

64	0	0.178744119	0.213459314	0.181554060	0.3663483538
128	0	0.193135238	0.193135238	0.195200276	0.3790146763
128	0.2	0.194492772	0.196335764	0.196334028	0.3876604459
128	0.4	0.193288930	0.192560958	0.194928116	0.3734112616
256	0	0.271806018	0.304047609	0.260519362	0.4207874827
256	0.2	0.237499292	0.275438795	0.230674964	0.4500284464
256	0.4	0.229696573	0.260287827	0.222793547	0.4383707237

Πίνακες σύγχυσης

64 μονάδες:

	Cmaj	C#maj	Dmaj	D#maj	Emaj	Fmaj	F#maj	Gmaj	G#maj	Amaj	A#maj	Bmaj	Cmin	C#min	Dmin	D#min	Emin
Cmaj	0.41	0.03	0.11	0.02	0.24	0.16	0.00	0.18	0.02	0.16	0.03	0.04	0.03	0.00	0.11	0.00	0.08
C#maj	0.00	0.14	0.01	0.01	0.00	0.00	0.08	0.00	0.01	0.01	0.01	0.00	0.01	0.16	0.00	0.10	0.00
Dmaj	0.01	0.01	0.06	0.01	0.00	0.01	0.00	0.02	0.00	0.07	0.00	0.00	0.00	0.03	0.05	0.04	0.07
D#maj	0.01	0.02	0.04	0.14	0.00	0.01	0.12	0.01	0.07	0.00	0.08	0.00	0.06	0.02	0.01	0.03	0.00
Emaj	0.01	0.00	0.03	0.00	0.00	0.01	0.00	0.01	0.00	0.02	0.00	0.00	0.00	0.00	0.01	0.00	0.02
Fmaj	0.12	0.01	0.03	0.03	0.05	0.33	0.08	0.12	0.02	0.07	0.04	0.00	0.03	0.00	0.10	0.00	0.14
F#maj	0.00	0.03	0.01	0.00	0.00	0.00	0.24	0.00	0.00	0.01	0.00	0.04	0.00	0.00	0.00	0.00	0.00
Gmaj	0.11	0.06	0.12	0.04	0.01	0.12	0.00	0.34	0.03	0.16	0.03	0.08	0.04	0.04	0.07	0.02	0.18
G#maj	0.02	0.09	0.02	0.13	0.00	0.01	0.11	0.01	0.25	0.00	0.12	0.04	0.12	0.14	0.01	0.09	0.00
Amaj	0.01	0.01	0.05	0.00	0.01	0.01	0.00	0.01	0.00	0.04	0.00	0.00	0.00	0.04	0.04	0.01	0.01
A#maj	0.02	0.02	0.02	0.12	0.00	0.02	0.01	0.02	0.10	0.02	0.23	0.54	0.07	0.00	0.00	0.02	0.00
Bmaj	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.07	0.00	0.00	0.00	0.03	0.01
Cmin	0.03	0.16	0.10	0.17	0.00	0.01	0.00	0.01	0.23	0.01	0.17	0.00	0.45	0.12	0.00	0.07	0.00
C#min	0.00	0.01	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.17	0.00	0.02	0.01
Dmin	0.04	0.00	0.01	0.01	0.00	0.06	0.04	0.05	0.01	0.02	0.01	0.00	0.01	0.00	0.36	0.00	0.04
D#min	0.00	0.03	0.00	0.01	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.20	0.00
Emin	0.03	0.00	0.02	0.01	0.00	0.02	0.00	0.04	0.00	0.03	0.00	0.00	0.00	0.00	0.03	0.00	0.15
Fmin	0.01	0.08	0.04	0.04	0.03	0.01	0.00	0.01	0.07	0.02	0.05	0.00	0.06	0.00	0.02	0.04	0.00
F#min	0.00	0.00	0.01	0.00	0.21	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.01	0.00	0.00
Gmin	0.01	0.02	0.03	0.06	0.00	0.01	0.00	0.00	0.05	0.04	0.06	0.00	0.04	0.00	0.01	0.07	0.00
G#min	0.00	0.01	0.00	0.00	0.00	0.00	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Amin	0.08	0.00	0.01	0.01	0.23	0.08	0.00	0.06	0.00	0.04	0.00	0.00	0.00	0.00	0.03	0.00	0.10
A#min	0.00	0.02	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.01	0.00	0.00	0.03	0.00	0.03	0.00
Bmin	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.07	0.00	0.01	0.00
Cdim	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
C#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Ddim	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.00	0.01	0.00	0.02	0.00	0.00
D#dim	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Edim	0.00	0.02	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.00
Fdim	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00
F#dim	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.00
Gdim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
G#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Adim	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.01	0.03	0.00	0.01	0.00
A#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00
Bdim	0.00	0.01	0.02	0.01	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.00
start	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
stop	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

	Fmin	F#min	Gmin	G#min	Amin	A#min	Bmin	Cdim	C#dim	Ddim	D#dim	Edim	Fdim	F#dim	Gdim	G#dim	Adim	A#dim	Bdim	start	stop
Cmaj	0.00	0.06	0.02	0.00	0.21	0.01	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.12	0.00	0.00	0.00	0.00	0.26	0.00	0.00
C#maj	0.01	0.00	0.00	0.17	0.00	0.22	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.12	0.00	0.00	0.00	0.00	0.51	0.00	0.00
Dmaj	0.00	0.00	0.02	0.00	0.02	0.00	0.22	0.22	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
D#maj	0.04	0.09	0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.11	0.00	0.00	0.00	0.00
Emaj	0.00	0.01	0.00	0.00	0.01	0.09	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Fmaj	0.04	0.00	0.02	0.00	0.12	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.17	0.00	0.00	0.10	0.00	0.00	0.00	0.00
F#maj	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Gmaj	0.03	0.10	0.06	0.00	0.12	0.01	0.20	0.22	0.00	0.00	0.00	0.00	0.00	0.16	0.00	0.00	0.11	0.00	0.00	0.00	0.00
G#maj	0.13	0.01	0.12	0.56	0.00	0.00	0.00	0.19	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00
Amaj	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
A#maj	0.11	0.00	0.11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.10	0.00	0.00	0.00	0.00
Bmaj	0.00	0.00	0.00	0.00	0.00	0.18	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Cmin	0.15	0.09	0.18	0.00	0.00	0.07	0.01	0.01	0.00	0.26	0.00	0.00	0.00	0.02	0.00	0.00	0.09	0.00	0.00	0.00	0.00
C#min	0.00	0.00	0.00	0.00	0.00	0.18	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.06	0.00	0.00	0.00	0.00
Dmin	0.00	0.00	0.03	0.03	0.06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
D#min	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Emin	0.00	0.00	0.00	0.00	0.05	0.00	0.20	0.00	0.00	0.00	0.00	0.00	0.00	0.09	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Fmin	0.33	0.24	0.04	0.00	0.01	0.00	0.00	0.01	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.15	0.00	0.00	0.00	0.00
F#min	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Gmin	0.04	0.03	0.08	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
G#min	0.00	0.00	0.00	0.00	0.00	0.00	0.09	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Amin	0.00	0.10	0.00	0.00	0.24	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00
A#min	0.00	0.03	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Bmin	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Cdim	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
C#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Ddim	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.72	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
D#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Edim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Fdim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
F#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Gdim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
G#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00
Adim	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
A#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Bdim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.11	0.77	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
start	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00
stop	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.98

128 μονάδες:

	Cmaj	C#maj	Dmaj	D#maj	Emaj	Fmaj	F#maj	Gmaj	G#maj	Amaj	A#maj	Bmaj	Cmin	C#min	Dmin	D#min	Emin
Cmaj	0.44	0.01	0.12	0.02	0.16	0.16	0.05	0.12	0.02	0.17	0.03	0.05	0.02	0.00	0.10	0.00	0.14
C#maj	0.00	0.21	0.02	0.00	0.00	0.00	0.07	0.00	0.01	0.02	0.01	0.02	0.01	0.04	0.00	0.00	0.00
Dmaj	0.01	0.00	0.13	0.00	0.05	0.01	0.02	0.02	0.01	0.04	0.00	0.06	0.00	0.08	0.01	0.03	0.01
D#maj	0.01	0.03	0.01	0.21	0.00	0.01	0.06	0.01	0.05	0.00	0.07	0.01	0.05	0.06	0.00	0.10	0.00
Emaj	0.01	0.00	0.03	0.00	0.10	0.01	0.02	0.01	0.00	0.02	0.00	0.00	0.00	0.00	0.01	0.00	0.02
Fmaj	0.11	0.04	0.06	0.02	0.06	0.32	0.02	0.08	0.02	0.05	0.03	0.01	0.02	0.00	0.13	0.00	0.10
F#maj	0.00	0.01	0.02	0.00	0.00	0.00	0.05	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.05	0.00
Gmaj	0.09	0.03	0.07	0.03	0.18	0.10	0.03	0.39	0.02	0.08	0.03	0.22	0.03	0.13	0.11	0.05	0.15
G#maj	0.02	0.11	0.02	0.11	0.00	0.01	0.06	0.02	0.34	0.00	0.07	0.03	0.10	0.21	0.01	0.12	0.01
Amaj	0.00	0.01	0.04	0.00	0.00	0.01	0.00	0.01	0.00	0.17	0.00	0.00	0.00	0.08	0.01	0.00	0.02
A#maj	0.02	0.02	0.02	0.10	0.01	0.02	0.03	0.02	0.07	0.00	0.28	0.00	0.06	0.00	0.03	0.03	0.01
Bmaj	0.00	0.00	0.00	0.00	0.01	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Cmin	0.03	0.10	0.01	0.16	0.00	0.02	0.07	0.04	0.19	0.03	0.15	0.10	0.56	0.05	0.02	0.10	0.02
C#min	0.00	0.02	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.09	0.00	0.00	0.00
Dmin	0.04	0.01	0.04	0.00	0.03	0.05	0.02	0.03	0.00	0.03	0.01	0.01	0.00	0.00	0.22	0.00	0.04
D#min	0.00	0.02	0.01	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.07	0.00
Emin	0.03	0.00	0.05	0.00	0.04	0.01	0.03	0.02	0.00	0.00	0.00	0.02	0.00	0.00	0.03	0.01	0.16
Fmin	0.01	0.04	0.02	0.06	0.01	0.01	0.03	0.02	0.07	0.00	0.05	0.00	0.05	0.00	0.01	0.02	0.00
F#min	0.00	0.01	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.04	0.00	0.01	0.00	0.00	0.00	0.00	0.00
Gmin	0.01	0.01	0.01	0.05	0.01	0.01	0.00	0.01	0.04	0.01	0.06	0.05	0.03	0.00	0.01	0.05	0.00
G#min	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Amin	0.07	0.00	0.02	0.00	0.05	0.08	0.00	0.05	0.00	0.04	0.00	0.00	0.00	0.00	0.07	0.00	0.07
A#min	0.00	0.03	0.00	0.00	0.01	0.00	0.01	0.00	0.00	0.03	0.00	0.03	0.00	0.00	0.00	0.02	0.00
Bmin	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Cdim	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
C#dim	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Ddim	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.00	0.01	0.00	0.01	0.02	0.00
D#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Edim	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Fdim	0.00	0.02	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00
F#dim	0.00	0.01	0.00	0.00	0.01	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01
Gdim	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
G#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Adim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.01	0.00	0.00	0.00	0.00	0.00
A#dim	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Bdim	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.04	0.00
start	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
stop	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00

	Fmin	F#min	Gmin	G#min	Amin	A#min	Bmin	Cdim	C#dim	Ddim	D#dim	Edim	Fdim	F#dim	Gdim	G#dim	Adim	A#dim	Bdim	start	stop
Cmaj	0.03	0.00	0.03	0.02	0.15	0.06	0.07	0.17	0.35	0.05	0.00	0.22	0.00	0.08	0.29	0.21	0.00	0.00	0.09	0.00	0.02
C#maj	0.02	0.00	0.01	0.00	0.00	0.07	0.00	0.00	0.05	0.03	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.01
Dmaj	0.00	0.00	0.00	0.00	0.01	0.01	0.14	0.01	0.00	0.00	0.00	0.00	0.70	0.07	0.00	0.06	0.00	0.00	0.01	0.00	0.00
D#maj	0.06	0.00	0.10	0.02	0.01	0.00	0.00	0.00	0.00	0.10	0.00	0.00	0.00	0.04	0.07	0.00	0.09	0.00	0.05	0.00	0.01
Emaj	0.00	0.00	0.00	0.00	0.01	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.56	0.00	0.00	0.00
Fmaj	0.02	0.09	0.03	0.00	0.15	0.00	0.00	0.01	0.11	0.11	0.04	0.00	0.00	0.02	0.00	0.18	0.08	0.00	0.07	0.00	0.02
F#maj	0.00	0.00	0.00	0.11	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00
Gmaj	0.03	0.00	0.02	0.15	0.11	0.08	0.23	0.03	0.11	0.01	0.00	0.00	0.00	0.09	0.07	0.10	0.03	0.00	0.16	0.00	0.03
G#maj	0.14	0.00	0.07	0.13	0.01	0.05	0.00	0.13	0.00	0.16	0.00	0.07	0.00	0.05	0.07	0.05	0.05	0.00	0.04	0.00	0.01
Amaj	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
A#maj	0.09	0.00	0.12	0.06	0.01	0.03	0.04	0.00	0.05	0.05	0.33	0.17	0.00	0.00	0.00	0.00	0.06	0.00	0.00	0.00	0.01
Bmaj	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Cmin	0.15	0.00	0.14	0.07	0.01	0.06	0.04	0.21	0.00	0.04	0.00	0.05	0.00	0.11	0.00	0.00	0.17	0.00	0.09	0.00	0.02
C#min	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Dmin	0.02	0.00	0.01	0.00	0.05	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.03	0.00	0.01	0.01	0.00	0.04	0.00	0.00
D#min	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Emin	0.00	0.00	0.00	0.00	0.04	0.01	0.03	0.05	0.00	0.00	0.00	0.07	0.00	0.00	0.00	0.06	0.00	0.00	0.00	0.00	0.00
Fmin	0.17	0.00	0.06	0.00	0.00	0.06	0.00	0.05	0.00	0.00	0.00	0.11	0.00	0.01	0.00	0.00	0.09	0.00	0.01	0.00	0.01
F#min	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Gmin	0.03	0.00	0.13	0.00	0.01	0.01	0.00	0.01	0.00	0.06	0.33	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.06	0.00	0.01
G#min	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.06	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Amin	0.00	0.00	0.01	0.00	0.24	0.00	0.06	0.01	0.01	0.00	0.00	0.02	0.00	0.00	0.05	0.00	0.00	0.00	0.04	0.00	0.00
A#min	0.00	0.00	0.00	0.00	0.00	0.14	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.15	0.00	0.00	0.00	0.00	0.00	0.00
Bmin	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00
Cdim	0.00	0.00	0.00	0.00	0.00	0.07	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.01	0.00	0.00	0.00	0.00
C#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00
Ddim	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.02	0.00	0.00	0.03	0.00	0.00	0.00	0.00
D#dim	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00
Edim	0.00	0.61	0.00	0.08	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Fdim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
F#dim	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.01	0.00	0.00	0.00	0.00
Gdim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00
G#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Adim	0.01	0.00	0.01	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.02	0.00	0.00	0.00	0.00
A#dim	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Bdim	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
start	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00
stop	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.14	0.00	0.00	0.98

256 μονάδες:

	Cmaj	C#maj	Dmaj	D#maj	Emaj	Fmaj	F#maj	Gmaj	G#maj	Amaj	A#maj	Bmaj	Cmin	C#min	Dmin	D#min	Emin	Fmin	F#min	Gmin	G#min	Amin	A#min	Bmin	Cdim	C#dim	Ddim	D#dim	Edim	Fdim	F#dim	Gdim	G#dim	Adim	A#dim	Bdim	start	stop
Cmaj	0.46	0.03	0.12	0.02	0.16	0.15	0.05	0.11	0.02	0.07	0.02	0.03	0.02	0.00	0.12	0.01	0.15	0.03	0.00	0.05	0.00	0.17	0.07	0.15	0.21	0.12	0.03	0.29	0.17	0.00	0.12	0.19	0.16	0.05	0.00	0.06	0.00	0.00
C#maj	0.00	0.32	0.00	0.01	0.01	0.00	0.06	0.00	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.00	0.00	0.07	0.00	0.00	0.00	0.02	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Dmaj	0.01	0.01	0.23	0.00	0.03	0.01	0.03	0.01	0.00	0.04	0.00	0.10	0.00	0.00	0.01	0.01	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.08	0.00	0.04	0.00	0.00
D#maj	0.01	0.04	0.00	0.30	0.00	0.01	0.02	0.01	0.05	0.00	0.07	0.01	0.05	0.08	0.01	0.01	0.00	0.05	0.00	0.07	0.15	0.00	0.00	0.03	0.00	0.00	0.02	0.00	0.00	0.00	0.03	0.00	0.00	0.03	0.00	0.00	0.00	0.00
Emaj	0.01	0.00	0.01	0.00	0.03	0.01	0.00	0.01	0.00	0.06	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.00	0.00	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00
Fmaj	0.10	0.04	0.06	0.02	0.06	0.38	0.06	0.08	0.02	0.06	0.04	0.02	0.02	0.00	0.13	0.06	0.07	0.03	0.01	0.03	0.00	0.12	0.00	0.03	0.00	0.00	0.03	0.00	0.11	0.00	0.06	0.02	0.00	0.10	0.08	0.03	0.00	0.00
F#maj	0.00	0.00	0.00	0.00	0.00	0.00	0.21	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.41	0.00	0.03	0.00	0.01	0.00	0.00
Gmaj	0.08	0.02	0.10	0.03	0.19	0.08	0.05	0.40	0.02	0.08	0.03	0.07	0.03	0.00	0.08	0.04	0.17	0.04	0.00	0.04	0.16	0.11	0.00	0.26	0.04	0.32	0.10	0.41	0.01	0.00	0.11	0.00	0.27	0.12	0.00	0.15	0.00	0.00
G#maj	0.02	0.04	0.00	0.09	0.00	0.01	0.05	0.02	0.39	0.01	0.07	0.06	0.11	0.00	0.01	0.01	0.00	0.12	0.00	0.06	0.05	0.00	0.05	0.01	0.00	0.00	0.13	0.00	0.00	0.00	0.02	0.02	0.00	0.11	0.00	0.00	0.00	0.00
Amaj	0.00	0.00	0.01	0.00	0.01	0.00	0.02	0.01	0.00	0.30	0.00	0.03	0.00	0.13	0.01	0.02	0.01	0.00	0.17	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.08	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
A#maj	0.02	0.03	0.03	0.09	0.00	0.02	0.01	0.03	0.07	0.01	0.35	0.00	0.06	0.00	0.02	0.03	0.01	0.06	0.00	0.09	0.11	0.01	0.04	0.01	0.03	0.02	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.09	0.00	0.00
Bmaj	0.00	0.00	0.00	0.00	0.01	0.00	0.03	0.00	0.00	0.00	0.00	0.07	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.15	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Cmin	0.03	0.15	0.02	0.15	0.00	0.03	0.03	0.04	0.18	0.04	0.13	0.06	0.59	0.03	0.02	0.11	0.01	0.13	0.00	0.13	0.01	0.01	0.04	0.01	0.20	0.00	0.10	0.00	0.09	0.14	0.14	0.00	0.00	0.07	0.08	0.02	0.00	0.00
C#min	0.00	0.02	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.29	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Dmin	0.04	0.00	0.03	0.01	0.04	0.05	0.01	0.03	0.00	0.02	0.01	0.06	0.00	0.00	0.27	0.00	0.04	0.01	0.08	0.01	0.00	0.05	0.00	0.09	0.16	0.12	0.02	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.03	0.00	0.00
D#min	0.00	0.01	0.00	0.01	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.06	0.00	0.07	0.00	0.26	0.00	0.00	0.02	0.00	0.00	0.00	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Emin	0.03	0.00	0.03	0.00	0.05	0.01	0.01	0.03	0.00	0.03	0.00	0.00	0.00	0.00	0.02	0.00	0.21	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.04	0.00	0.04	0.00	0.00	0.02	0.00	0.00
Fmin	0.01	0.06	0.01	0.06	0.01	0.01	0.05	0.02	0.07	0.03	0.04	0.01	0.04	0.03	0.02	0.01	0.00	0.27	0.00	0.04	0.06	0.01	0.12	0.00	0.00	0.00	0.07	0.00	0.01	0.00	0.00	0.05	0.00	0.00	0.00	0.02	0.00	0.00
F#min	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.13	0.00	0.00	0.00	0.00	0.61	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Gmin	0.01	0.02	0.04	0.03	0.01	0.01	0.00	0.01	0.03	0.01	0.06	0.00	0.03	0.00	0.01	0.07	0.00	0.04	0.00	0.23	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.05	0.00	0.00
G#min	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.06	0.00	0.00	0.00	0.00	0.00
Amin	0.06	0.00	0.07	0.00	0.05	0.07	0.00	0.06	0.00	0.03	0.00	0.03	0.00	0.00	0.06	0.00	0.09	0.01	0.08	0.01	0.00	0.26	0.00	0.03	0.00	0.00	0.00	0.00	0.04	0.00	0.08	0.00	0.00	0.00	0.00	0.00	0.00	0.00
A#min	0.00	0.01	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.23	0.01	0.00	0.00	0.00	0.00	0.10	0.00	0.00	0.00	0.11	0.00	0.00	0.00	0.00	0.00
Bmin	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Cdim	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
C#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Ddim	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.01	0.01	0.00	0.01	0.00	0.01	0.00	0.00	0.01	0.00	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.04	0.00	0.00	0.37	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00
D#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Edim	0.00	0.02	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Fdim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.06	0.00	0.00	0.00	0.00	0.00
F#dim	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Gdim	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.06	0.00	0.00	0.00	0.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
G#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Adim	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00
A#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.35	0.00	0.00	0.00
Bdim	0.00	0.00	0.01	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.06	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.06	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.00	0.21	0.00	0.00
start	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00
stop	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.97

Κωδικοποιητής – Αποκωδικοποιητής – αμφίδρομα LSTM

Μετά το βασικό μοντέλο κωδικοποιητή – αποκωδικοποιητή, δοκιμάσαμε να αντικαταστήσουμε τα LSTM με LSTM διπλής κατεύθυνσης. Οι τιμές του αριθμού αναδρομικών μονάδων που εξετάσαμε είναι 64 και 128, μιας και ουσιαστικά με τη χρήση LSTM διπλής κατεύθυνσης διπλασιάζεται ο αριθμός των παραμέτρων που εισάγουν στο δίκτυο τα LSTM.

train				
	macro			micro
Αριθμός μονάδων	f1	precision	recall	f1
64	0.394234092	0.500234932	0.346193244	0.524050548
128	0.678076161	0.729511840	0.643021343	0.714514462

test				
	macro			micro
Αριθμός μονάδων	f1	precision	recall	f1
64	0.201354985	0.220859624	0.195763700	0.4034395807
128	0.243864025	0.257660690	0.240700717	0.4327730004

Πίνακες σύγχυσης

64 μονάδες:

	Cmaj	C#maj	Dmaj	D#maj	Emaj	Fmaj	F#maj	Gmaj	G#maj	Amaj	A#maj	Bmaj	Cmin	C#min	Dmin	D#min	Emin	Fmin	F#min	Gmin	G#min	Amin	A#min	Bmin	Cdim	C#dim	Ddim	D#dim	Edim	Fdim	F#dim	Gdim	G#dim	Adim	A#dim	Bdim	start	stop
Cmaj	0.46	0.06	0.12	0.01	0.07	0.14	0.02	0.13	0.01	0.15	0.01	0.00	0.02	0.00	0.14	0.00	0.30	0.02	0.05	0.04	0.00	0.32	0.01	0.01	0.03	0.12	0.04	0.00	0.09	0.00	0.06	0.00	0.00	0.12	0.00	0.04	0.00	0.00
C#maj	0.00	0.25	0.01	0.01	0.01	0.00	0.06	0.00	0.01	0.01	0.01	0.06	0.01	0.01	0.00	0.04	0.00	0.02	0.00	0.01	0.06	0.00	0.18	0.00	0.09	0.00	0.01	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.41	0.00	0.00	0.00
Dmaj	0.01	0.00	0.13	0.00	0.04	0.01	0.04	0.02	0.00	0.06	0.00	0.01	0.00	0.04	0.02	0.00	0.02	0.00	0.13	0.01	0.00	0.01	0.00	0.12	0.05	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.05	0.00	0.00
D#maj	0.01	0.02	0.02	0.21	0.00	0.01	0.00	0.01	0.07	0.00	0.06	0.00	0.06	0.00	0.01	0.06	0.00	0.05	0.00	0.08	0.00	0.01	0.04	0.00	0.06	0.00	0.06	0.28	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00
Emaj	0.01	0.00	0.01	0.00	0.08	0.01	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.07	0.01	0.00	0.01	0.00	0.01	0.00	0.11	0.01	0.00	0.11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.64	0.00	0.00	0.00	0.00	0.00
Fmaj	0.10	0.00	0.05	0.02	0.06	0.38	0.03	0.09	0.01	0.00	0.03	0.01	0.02	0.07	0.39	0.03	0.10	0.04	0.00	0.04	0.00	0.13	0.00	0.02	0.00	0.00	0.04	0.00	0.00	0.00	0.04	0.00	0.00	0.15	0.00	0.02	0.00	0.00
F#maj	0.00	0.04	0.01	0.00	0.00	0.00	0.05	0.00	0.00	0.03	0.00	0.03	0.00	0.00	0.00	0.05	0.01	0.00	0.13	0.00	0.06	0.00	0.00	0.04	0.08	0.00	0.03	0.00	0.00	0.00	0.01	0.00	0.06	0.06	0.00	0.00	0.00	0.00
Gmaj	0.08	0.01	0.13	0.01	0.13	0.09	0.01	0.39	0.02	0.06	0.02	0.02	0.04	0.00	0.13	0.10	0.15	0.02	0.01	0.02	0.19	0.09	0.00	0.14	0.00	0.00	0.07	0.00	0.00	0.00	0.10	0.00	0.00	0.04	0.00	0.15	0.00	0.00
G#maj	0.02	0.08	0.01	0.11	0.00	0.01	0.13	0.01	0.33	0.05	0.09	0.16	0.11	0.04	0.01	0.03	0.01	0.11	0.01	0.08	0.11	0.00	0.02	0.06	0.00	0.00	0.12	0.00	0.00	0.00	0.02	0.00	0.00	0.02	0.29	0.13	0.00	0.00
Amaj	0.01	0.00	0.04	0.00	0.00	0.01	0.00	0.01	0.00	0.21	0.00	0.04	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.04	0.00	0.04	0.09	0.00	0.14	0.00	0.00	0.06	0.05	0.00	0.00	0.00	0.00	0.00	0.00
A#maj	0.02	0.02	0.01	0.12	0.01	0.03	0.02	0.02	0.08	0.01	0.34	0.00	0.05	0.00	0.01	0.01	0.00	0.07	0.00	0.13	0.11	0.01	0.08	0.03	0.00	0.00	0.07	0.00	0.02	0.00	0.02	0.00	0.00	0.05	0.00	0.03	0.00	0.00
Bmaj	0.00	0.00	0.01	0.00	0.02	0.00	0.06	0.00	0.00	0.00	0.00	0.21	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Cmin	0.03	0.16	0.04	0.15	0.06	0.01	0.16	0.03	0.18	0.00	0.13	0.00	0.53	0.08	0.01	0.13	0.01	0.17	0.00	0.15	0.00	0.00	0.08	0.02	0.01	0.00	0.10	0.00	0.00	0.70	0.03	0.00	0.00	0.09	0.00	0.08	0.00	0.00
C#min	0.00	0.01	0.00	0.00	0.00	0.00	0.06	0.00	0.00	0.00	0.00	0.00	0.00	0.17	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.27	0.00	0.00	0.00	0.00	0.00	0.65	0.00	0.06	0.00	0.00	0.00	0.00
Dmin	0.04	0.00	0.04	0.00	0.06	0.29	0.00	0.04	0.01	0.00	0.01	0.00	0.01	0.00	0.20	0.00	0.02	0.01	0.08	0.01	0.00	0.06	0.03	0.00	0.00	0.00	0.03	0.00	0.02	0.00	0.07	0.00	0.00	0.00	0.00	0.00	0.00	0.00
D#min	0.00	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.00	0.22	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.06	0.00	0.00	0.00	0.01	0.28	0.00	0.00	0.09	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Emin	0.02	0.00	0.05	0.00	0.06	0.02	0.00	0.04	0.00	0.01	0.00	0.00	0.00	0.04	0.01	0.00	0.16	0.00	0.04	0.00	0.00	0.04	0.00	0.08	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00
Fmin	0.01	0.06	0.01	0.07	0.01	0.01	0.04	0.01	0.06	0.00	0.04	0.00	0.05	0.00	0.01	0.05	0.01	0.22	0.00	0.05	0.04	0.00	0.04	0.01	0.00	0.00	0.10	0.00	0.00	0.00	0.00	0.00	0.00	0.07	0.00	0.04	0.00	0.00
F#min	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.09	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.49	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Gmin	0.01	0.01	0.02	0.05	0.00	0.01	0.01	0.01	0.03	0.04	0.07	0.00	0.03	0.00	0.02	0.00	0.00	0.03	0.00	0.16	0.00	0.00	0.01	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00
G#min	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Amin	0.06	0.01	0.05	0.00	0.09	0.08	0.00	0.05	0.00	0.06	0.00	0.00	0.00	0.00	0.07	0.00	0.09	0.00	0.00	0.01	0.00	0.25	0.00	0.04	0.00	0.00	0.00	0.00	0.01	0.00	0.07	0.00	0.00	0.00	0.00	0.00	0.00	0.00
A#min	0.00	0.03	0.01	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.09	0.00	0.00	0.00	0.01	0.02	0.00	0.14	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Bmin	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Cdim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.34	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
C#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Ddim	0.00	0.01	0.01	0.00	0.00	0.01	0.00	0.01	0.01	0.00	0.01	0.00	0.01	0.00	0.00	0.04	0.01	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.05	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.08	0.00	0.00
D#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Edim	0.00	0.02	0.01	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.31	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Fdim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
F#dim	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Gdim	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
G#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Adim	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.01	0.01	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
A#dim	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Bdim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.04	0.00	0.01	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.37	0.00	0.00
start	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00
stop	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.98

128 μονάδες:

	Cmaj	C#maj	Dmaj	D#maj	Emaj	Fmaj	F#maj	Gmaj	G#maj	Amaj	A#maj	Bmaj	Cmin	C#min	Dmin	D#min	Emin	Fmin	F#min	Gmin	G#min	Amin	A#min	Bmin	Cdim	C#dim	Ddim	D#dim	Edim	Fdim	F#dim	Gdim	G#dim	Adim	A#dim	Bdim	start	stop
Cmaj	0.61	0.05	0.08	0.06	0.13	0.13	0.03	0.13	0.02	0.13	0.04	0.11	0.04	0.00	0.14	0.00	0.12	0.02	0.02	0.01	0.00	0.18	0.02	0.11	0.00	0.02	0.02	0.00	0.15	#VALUE!	0.09	0.00	0.12	0.01	#VALUE!	0.01	0.00	0.01
C#maj	0.00	0.20	0.00	0.01	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.00	0.01	0.00	0.00	0.05	0.00	0.01	0.00	0.01	0.00	0.00	0.13	0.01	0.00	0.00	0.01	0.00	0.00	#VALUE!	0.00	0.01	0.00	0.00	#VALUE!	0.00	0.00	0.00
Dmaj	0.01	0.00	0.23	0.00	0.04	0.01	0.01	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.05	0.01	0.00	0.00	0.01	0.01	0.01	0.00	0.18	0.04	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00
D#maj	0.01	0.05	0.01	0.25	0.00	0.01	0.04	0.02	0.06	0.00	0.06	0.00	0.05	0.00	0.01	0.02	0.00	0.04	0.00	0.04	0.06	0.00	0.02	0.03	0.00	0.00	0.04	0.00	0.04	#VALUE!	0.00	0.00	0.00	0.01	#VALUE!	0.01	0.00	0.00
Emaj	0.01	0.00	0.01	0.00	0.17	0.01	0.03	0.01	0.00	0.00	0.00	0.00	0.00	0.19	0.01	0.00	0.01	0.00	0.00	0.00	0.08	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00
Fmaj	0.10	0.01	0.06	0.04	0.08	0.39	0.06	0.08	0.03	0.08	0.05	0.00	0.02	0.00	0.13	0.00	0.08	0.02	0.00	0.04	0.01	0.13	0.04	0.04	0.01	0.18	0.01	0.53	0.00	#VALUE!	0.08	0.00	0.00	0.04	#VALUE!	0.10	0.00	0.00
F#maj	0.00	0.01	0.00	0.00	0.00	0.00	0.13	0.00	0.00	0.00	0.00	0.10	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.01	0.00	0.02	0.03	0.00	0.00	0.01	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00
Gmaj	0.07	0.04	0.15	0.04	0.09	0.08	0.04	0.40	0.04	0.05	0.04	0.06	0.03	0.01	0.11	0.09	0.15	0.03	0.07	0.04	0.00	0.10	0.04	0.09	0.10	0.18	0.13	0.00	0.00	#VALUE!	0.16	0.08	0.17	0.12	#VALUE!	0.10	0.00	0.01
G#maj	0.01	0.06	0.01	0.08	0.00	0.01	0.08	0.01	0.36	0.00	0.06	0.00	0.10	0.00	0.00	0.11	0.00	0.11	0.00	0.08	0.06	0.00	0.11	0.01	0.09	0.00	0.06	0.00	0.08	#VALUE!	0.04	0.01	0.00	0.08	#VALUE!	0.00	0.00	0.00
Amaj	0.01	0.01	0.02	0.00	0.00	0.01	0.00	0.01	0.00	0.27	0.00	0.05	0.00	0.00	0.01	0.09	0.02	0.00	0.46	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.13	#VALUE!	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00
A#maj	0.01	0.03	0.01	0.07	0.01	0.02	0.04	0.02	0.05	0.01	0.38	0.05	0.05	0.00	0.02	0.04	0.00	0.07	0.00	0.10	0.00	0.01	0.04	0.03	0.04	0.00	0.09	0.04	0.05	#VALUE!	0.00	0.04	0.00	0.04	#VALUE!	0.06	0.00	0.00
Bmaj	0.00	0.02	0.01	0.00	0.01	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.11	0.00	#VALUE!	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00
Cmin	0.01	0.11	0.05	0.13	0.00	0.03	0.04	0.04	0.18	0.00	0.10	0.00	0.57	0.00	0.01	0.06	0.01	0.16	0.00	0.14	0.30	0.01	0.04	0.08	0.06	0.00	0.18	0.00	0.03	#VALUE!	0.09	0.09	0.07	0.08	#VALUE!	0.20	0.00	0.00
C#min	0.00	0.04	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.44	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.13	0.00	#VALUE!	0.01	0.00	0.00
Dmin	0.03	0.01	0.04	0.01	0.01	0.06	0.01	0.04	0.01	0.10	0.01	0.00	0.01	0.00	0.27	0.00	0.05	0.01	0.00	0.01	0.00	0.04	0.01	0.00	0.01	0.00	0.01	0.00	0.02	#VALUE!	0.00	0.00	0.06	0.00	#VALUE!	0.00	0.00	0.00
D#min	0.00	0.00	0.01	0.01	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.09	0.00	0.00	0.00	0.01	#VALUE!	0.03	0.24	0.00	0.00	#VALUE!	0.01	0.00	0.00
Emin	0.02	0.01	0.02	0.01	0.05	0.02	0.00	0.03	0.00	0.06	0.01	0.00	0.00	0.00	0.01	0.00	0.14	0.01	0.00	0.00	0.00	0.03	0.01	0.04	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.02	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00
Fmin	0.01	0.08	0.01	0.04	0.01	0.01	0.05	0.01	0.06	0.00	0.04	0.08	0.04	0.19	0.01	0.04	0.01	0.15	0.00	0.04	0.00	0.00	0.02	0.03	0.01	0.00	0.06	0.00	0.08	#VALUE!	0.02	0.06	0.04	0.00	#VALUE!	0.01	0.00	0.00
F#min	0.00	0.01	0.00	0.00	0.02	0.00	0.03	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.01	0.00	0.01	0.00	0.04	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00
Gmin	0.01	0.03	0.01	0.04	0.00	0.01	0.01	0.01	0.03	0.02	0.05	0.00	0.03	0.00	0.01	0.00	0.00	0.04	0.16	0.17	0.03	0.00	0.01	0.01	0.01	0.33	0.02	0.00	0.03	#VALUE!	0.05	0.15	0.03	0.04	#VALUE!	0.05	0.00	0.00
G#min	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.06	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00
Amin	0.06	0.01	0.02	0.01	0.11	0.08	0.00	0.05	0.01	0.06	0.01	0.00	0.01	0.00	0.06	0.00	0.07	0.01	0.00	0.00	0.02	0.15	0.01	0.02	0.00	0.00	0.01	0.00	0.01	#VALUE!	0.03	0.00	0.07	0.00	#VALUE!	0.06	0.00	0.01
A#min	0.00	0.01	0.00	0.01	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.10	0.00	0.00	0.00	0.00	0.04	0.01	#VALUE!	0.00	0.00	0.01	0.00	#VALUE!	0.00	0.00	0.00
Bmin	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00
Cdim	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.11	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00
C#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00
Ddim	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.01	0.01	0.00	0.01	0.00	0.01	0.00	0.01	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.01	#VALUE!	0.00	0.00	0.00	0.03	#VALUE!	0.00	0.00	0.00
D#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00
Edim	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.07	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.06	#VALUE!	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00
Fdim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00
F#dim	0.00	0.00	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.05	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.03	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00
Gdim	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00
G#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00
Adim	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.13	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.05	0.00	0.00	0.26	#VALUE!	0.00	0.00	0.00
A#dim	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00
Bdim	0.00	0.00	0.00	0.00	0.02	0.00	0.02	0.01	0.00	0.01	0.01	0.08	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.01	#VALUE!	0.08	0.00	0.00
start	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	#VALUE!	0.00	1.00	0.00
stop	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.01	0.01	0.00	#VALUE!	0.00	0.00	0.98

Κωδικοποιητής – Αποκωδικοποιητής – 2 επίπεδα απλών LSTM

Στην επόμενη φάση των πειραμάτων εξετάσαμε ένα πιο βαθύ μοντέλο κωδικοποιητή – αποκωδικοποιητή προσθέτοντας από ένα επίπεδο LSTM σε κωδικοποιητή και αποκωδικοποιητή αντίστοιχα θέλοντας να εξετάσουμε αν η αύξηση των παραμέτρων του συστήματος προσθέτοντας επίπεδα αντί για επιπλέον αναδρομικές μονάδες στο κάθε LSTM θα έχει καλύτερα αποτελέσματα.

train				
	macro			micro
Αριθμός μονάδων	f1	precision	recall	f1
128	0.130429657	0.190863886	0.143303180	0.3935981674
256	0.191350790	0.288212366	0.181178088	0.3846558599

test				
	macro			micro
Αριθμός μονάδων	f1	precision	recall	f1
128	0.243864025	0.153678099	0.121103476	0.392924937
256	0.156007937	0.179531856	0.156806167	0.3722402534

Κωδικοποιητής – Αποκωδικοποιητής – απλό LSTM – μηχανισμός προσοχής

Το τελευταία μοντέλο που δοκιμάσαμε είναι αυτό του απλού κωδικοποιητή - αποκωδικοποιητή με μηχανισμό προσοχής, θέλοντας να εξετάσουμε την επίδραση του μηχανισμού προσοχής σε ένα μοντέλο πρόβλεψης συγχορδιών.

train					
		macro			micro
Αριθμός μονάδων	Dropout	f1	precision	recall	f1
64	0	0.240447077	0.361248134	0.218152824	0.39474728073655
128	0	0.546746286	0.593623244	0.439509787	0.63659402579133
128	0.2	0.503298234	0.572390232	0.420432982	0.60423912838432
128	0.4	0.480239432	0.550324954	0.410293484	0.55329042290299
256	0	0.806041529	0.851106608	0.771515168	0.81610371248122
256	0.2	0.649023429	0.710235923	0.539283993	0.73013290284320
256	0.4	0.518753289	0.650294829	0.490829022	0.61928324895342

test					
		macro			micro
Αριθμός μονάδων	Dropout	f1	precision	recall	f1

64	0	0.16344172	0.202197463	0.165251017	0.4327561906
128	0	0.217853753	0.249913706	0.212841092	0.4517934958
128	0.2	0.209682592	0.245756279	0.210592626	0.4522582251
128	0.4	0.214799639	0.237053767	0.208706201	0.4399627823
256	0	0.293506781	0.329296791	0.291355126	0.4937596172
256	0.2	0.324148537	0.355276973	0.311456641	0.5083427961
256	0.4	0.316576513	0.349736051	0.299163711	0.5034777127

Πίνακες σύγχυσης

64 μονάδες:

	Cmaj	C#maj	Dmaj	D#maj	Emaj	Fmaj	F#maj	Gmaj	G#maj	Amaj	A#maj	Bmaj	Cmin	C#min	Dmin	D#min	Emin	Fmin	F#min	Gmin	G#min	Amin	A#min	Bmin	Cdim	C#dim	Ddim	D#dim	Edim	Fdim	F#dim	Gdim	G#dim	Adim	A#dim	Bdim	start	stop
Cmaj	0.47	0.01	0.07	0.02	0.02	0.14	0.02	0.10	0.00	0.08	0.03	0.05	0.02	0.00	0.09	0.00	0.16	0.03	0.00	0.03	0.00	0.15	0.03	0.39	0.16	#VALUE!	0.00	0.00	0.07	0.00	0.13	0.00	#VALUE!	0.24	#VALUE!	0.13	0.00	0.00
C#maj	0.00	0.46	0.00	0.01	0.03	0.00	0.08	0.00	0.01	0.00	0.01	0.00	0.01	0.00	0.00	0.02	0.00	0.03	0.00	0.00	0.00	0.00	0.09	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.09	0.00	0.00
Dmaj	0.01	0.00	0.29	0.00	0.00	0.01	0.14	0.02	0.00	0.01	0.00	0.02	0.00	0.03	0.01	0.00	0.02	0.00	0.00	0.01	0.00	0.02	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
D#maj	0.01	0.03	0.00	0.28	0.00	0.01	0.02	0.01	0.08	0.02	0.08	0.02	0.06	0.05	0.01	0.07	0.00	0.02	0.00	0.07	0.47	0.00	0.04	0.00	0.06	#VALUE!	0.00	0.00	0.04	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.03	0.00	0.00
Emaj	0.01	0.00	0.01	0.00	0.32	0.00	0.05	0.01	0.00	0.01	0.00	0.02	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.02	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
Fmaj	0.10	0.00	0.03	0.02	0.04	0.39	0.00	0.06	0.01	0.05	0.03	0.04	0.02	0.00	0.11	0.00	0.07	0.02	0.00	0.04	0.00	0.13	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.07	0.00	0.00	0.00	#VALUE!	0.14	#VALUE!	0.05	0.00	0.00
F#maj	0.00	0.00	0.00	0.01	0.00	0.00	0.23	0.00	0.00	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.05	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.03	#VALUE!	0.00	0.00	0.00
Gmaj	0.08	0.12	0.16	0.02	0.17	0.08	0.01	0.47	0.03	0.06	0.03	0.18	0.03	0.12	0.19	0.05	0.28	0.03	0.00	0.03	0.00	0.09	0.00	0.31	0.00	#VALUE!	0.00	0.00	0.06	0.25	0.14	0.00	#VALUE!	0.09	#VALUE!	0.16	0.00	0.00
G#maj	0.01	0.09	0.02	0.09	0.12	0.01	0.01	0.01	0.33	0.00	0.07	0.00	0.12	0.00	0.01	0.07	0.00	0.14	0.00	0.06	0.23	0.00	0.10	0.00	0.11	#VALUE!	0.00	0.00	0.04	0.20	0.29	0.35	#VALUE!	0.03	#VALUE!	0.01	0.00	0.00
Amaj	0.00	0.00	0.03	0.00	0.00	0.01	0.00	0.01	0.00	0.23	0.01	0.00	0.00	0.05	0.01	0.00	0.00	0.00	0.37	0.00	0.00	0.01	0.02	0.00	0.06	#VALUE!	0.00	0.00	0.04	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
A#maj	0.02	0.00	0.03	0.07	0.04	0.02	0.00	0.02	0.08	0.00	0.35	0.00	0.07	0.02	0.02	0.00	0.00	0.10	0.00	0.13	0.00	0.00	0.05	0.00	0.06	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
Bmaj	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.70	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
Cmin	0.03	0.06	0.00	0.16	0.00	0.01	0.05	0.03	0.16	0.00	0.10	0.05	0.50	0.01	0.01	0.09	0.00	0.09	0.00	0.13	0.00	0.00	0.09	0.00	0.00	#VALUE!	0.00	0.00	0.04	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
C#min	0.00	0.02	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.05	0.00	0.00	0.00	0.26	0.00	0.02	0.00	0.00	0.07	0.00	0.00	0.00	0.00	0.00	0.02	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
Dmin	0.04	0.00	0.02	0.01	0.02	0.07	0.01	0.04	0.00	0.09	0.01	0.00	0.00	0.00	0.28	0.00	0.02	0.01	0.00	0.02	0.00	0.06	0.05	0.00	0.00	#VALUE!	0.00	0.00	0.06	0.00	0.00	0.00	#VALUE!	0.03	#VALUE!	0.00	0.00	0.00
D#min	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.09	0.00	0.00	0.00	0.35	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.06	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
Emin	0.03	0.00	0.01	0.00	0.01	0.02	0.05	0.03	0.00	0.02	0.00	0.07	0.00	0.04	0.03	0.00	0.20	0.00	0.27	0.00	0.00	0.04	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.05	0.00	0.00
Fmin	0.01	0.06	0.01	0.06	0.00	0.02	0.00	0.01	0.06	0.00	0.04	0.00	0.05	0.04	0.01	0.05	0.00	0.28	0.00	0.02	0.00	0.00	0.04	0.00	0.00	#VALUE!	0.70	0.00	0.00	0.00	0.00	0.22	#VALUE!	0.01	#VALUE!	0.02	0.00	0.00
F#min	0.00	0.00	0.01	0.00	0.00	0.00	0.07	0.00	0.00	0.00	0.00	0.00	0.00	0.11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.02	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
Gmin	0.01	0.01	0.01	0.06	0.00	0.01	0.00	0.01	0.03	0.06	0.07	0.00	0.04	0.01	0.02	0.00	0.00	0.02	0.00	0.23	0.00	0.00	0.02	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.13	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
G#min	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
Amin	0.07	0.02	0.04	0.00	0.06	0.07	0.00	0.05	0.00	0.09	0.00	0.00	0.00	0.00	0.04	0.00	0.02	0.01	0.00	0.00	0.00	0.31	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.03	#VALUE!	0.00	0.00	0.00
A#min	0.00	0.01	0.02	0.00	0.02	0.00	0.03	0.00	0.00	0.00	0.00	0.01	0.00	0.06	0.00	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.11	0.00	0.13	#VALUE!	0.00	0.00	0.15	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.02	0.00	0.00
Bmin	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
Cdim	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.02	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
C#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
Ddim	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.02	0.03	0.01	0.00	0.00	0.00	0.00	0.02	0.00	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.03	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.02	0.00	0.00
D#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
Edim	0.00	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.00	#VALUE!	0.00	0.00	0.14	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
Fdim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
F#dim	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.06	#VALUE!	0.00	0.00	0.00
Gdim	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
G#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
Adim	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.25	0.14	0.00	#VALUE!	0.04	#VALUE!	0.02	0.00	0.00
A#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.04	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.00
Bdim	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.01	0.00	0.01	0.00	0.01	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.17	0.00	0.00
start	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	1.00	0.00
stop	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	0.00	0.00	0.00	0.00	0.00	#VALUE!	0.00	#VALUE!	0.00	0.00	0.99

256 μονάδες:

	Cmaj	C#maj	Dmaj	D#maj	Emaj	Fmaj	F#maj	Gmaj	G#maj	Amaj	A#maj	Bmaj	Cmin	C#min	Dmin	D#min	Emin	Fmin	F#min	Gmin	G#min	Amin	A#min	Bmin	Cdim	C#dim	Ddim	D#dim	Edim	Fdim	F#dim	Gdim	G#dim	Adim	A#dim	Bdim	start	stop
Cmaj	0.52	0.03	0.07	0.03	0.12	0.12	0.03	0.11	0.02	0.07	0.02	0.00	0.03	0.00	0.09	0.00	0.17	0.03	0.00	0.03	0.12	0.15	0.00	0.00	0.00	0.05	0.05	0.12	0.11	0.22	0.04	0.02	0.00	0.03	0.00	0.02	0.00	0.00
C#maj	0.00	0.32	0.00	0.00	0.03	0.00	0.06	0.00	0.01	0.00	0.01	0.00	0.01	0.28	0.00	0.00	0.00	0.02	0.00	0.01	0.00	0.00	0.16	0.00	0.00	0.00	0.02	0.02	0.00	0.18	0.03	0.03	0.00	0.01	0.00	0.02	0.00	0.00
Dmaj	0.01	0.01	0.31	0.00	0.02	0.01	0.01	0.02	0.00	0.06	0.00	0.05	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.12	0.01	0.00	0.17	0.00	0.09	0.01	0.00	0.00	0.00	0.04	0.03	0.00	0.06	0.00	0.03	0.00	0.00
D#maj	0.01	0.03	0.01	0.43	0.01	0.01	0.07	0.01	0.04	0.01	0.05	0.00	0.04	0.00	0.02	0.06	0.01	0.04	0.00	0.07	0.03	0.00	0.00	0.00	0.28	0.02	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.17	0.00	0.02	0.00	0.00
Emaj	0.01	0.00	0.01	0.00	0.20	0.01	0.02	0.01	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.04	0.02	0.00	0.00	0.00	0.12	0.01	0.00	0.00	0.00	0.07	0.00	0.08	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00
Fmaj	0.09	0.01	0.07	0.01	0.02	0.45	0.01	0.06	0.01	0.10	0.03	0.00	0.02	0.00	0.11	0.00	0.06	0.02	0.00	0.04	0.03	0.13	0.04	0.00	0.00	0.06	0.03	0.08	0.00	0.00	0.09		0.15	0.06	0.00	0.01	0.00	0.00
F#maj	0.00	0.01	0.00	0.00	0.01	0.00	0.21	0.00	0.00	0.00	0.00	0.08	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.06	0.00	0.02	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.11	0.00	0.01	0.00	0.00	0.00	0.00
Gmaj	0.07	0.03	0.15	0.02	0.14	0.07	0.01	0.51	0.01	0.04	0.04	0.19	0.03	0.00	0.09	0.00	0.14	0.02	0.00	0.04	0.02	0.08	0.00	0.26	0.00	0.08	0.13	0.14	0.06	0.18	0.12		0.22	0.00	0.00	0.13	0.00	0.00
G#maj	0.02	0.05	0.00	0.09	0.01	0.01	0.04	0.01	0.47	0.00	0.06	0.02	0.10	0.00	0.02	0.15	0.01	0.11	0.00	0.08	0.01	0.00	0.03	0.09	0.00	0.00	0.15	0.04	0.03	0.13	0.03	0.05	0.00	0.06	0.00	0.08	0.00	0.00
Amaj	0.00	0.02	0.03	0.00	0.04	0.01	0.03	0.01	0.00	0.38	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.70	0.00	0.01	0.01	0.00	0.02	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.58	0.02	0.00	0.00
A#maj	0.02	0.03	0.02	0.07	0.01	0.02	0.03	0.02	0.05	0.03	0.45	0.07	0.05	0.00	0.02	0.11	0.01	0.06	0.00	0.11	0.01	0.01	0.02	0.03	0.00	0.04	0.08	0.02	0.17	0.00	0.00	0.00	0.00	0.00	0.00	0.07	0.00	0.00
Bmaj	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.21	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.01	0.09	0.03	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Cmin	0.03	0.09	0.02	0.14	0.01	0.03	0.09	0.03	0.18	0.00	0.12	0.05	0.60	0.00	0.04	0.00	0.02	0.14	0.00	0.09	0.00	0.01	0.16	0.01	0.00	0.20	0.05	0.03	0.11	0.00	0.00	0.54	0.00	0.07	0.00	0.09	0.00	0.00
C#min	0.00	0.03	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.72	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.04	0.01	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00
Dmin	0.03	0.00	0.04	0.00	0.03	0.04	0.00	0.03	0.00	0.06	0.02	0.00	0.00	0.00	0.37	0.00	0.01	0.01	0.00	0.01	0.01	0.06	0.00	0.02	0.00	0.02	0.01	0.01	0.00	0.00	0.00	0.01	0.12	0.00	0.00	0.00	0.00	0.00
D#min	0.00	0.01	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.41	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.04	0.02	0.00	0.02	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00
Emin	0.03	0.01	0.04	0.00	0.04	0.02	0.00	0.03	0.00	0.01	0.00	0.02	0.00	0.00	0.01	0.00	0.26	0.00	0.00	0.00	0.03	0.03	0.00	0.01	0.04	0.03	0.03	0.08	0.11	0.00	0.03	0.01	0.06	0.00	0.00	0.03	0.00	0.00
Fmin	0.01	0.06	0.01	0.05	0.02	0.02	0.06	0.01	0.06	0.01	0.04	0.02	0.04	0.00	0.00	0.00	0.00	0.35	0.00	0.03	0.00	0.01	0.01	0.01	0.01	0.00	0.08	0.02	0.00	0.00	0.10	0.01	0.00	0.00	0.00	0.06	0.00	0.00
F#min	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.25	0.00	0.12	0.00	0.00	0.09	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00
Gmin	0.01	0.00	0.00	0.04	0.00	0.01	0.00	0.01	0.03	0.00	0.05	0.01	0.03	0.00	0.01	0.00	0.00	0.03	0.00	0.31	0.00	0.00	0.00	0.02	0.01	0.01	0.04	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00
G#min	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.02	0.01	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Amin	0.06	0.01	0.04	0.00	0.03	0.07	0.00	0.04	0.00	0.03	0.00	0.00	0.00	0.00	0.04	0.00	0.07	0.00	0.00	0.01	0.00	0.37	0.05	0.02	0.01	0.01	0.03	0.00	0.00	0.00	0.00	0.00	0.05	0.00	0.00	0.01	0.00	0.00
A#min	0.00	0.01	0.00	0.00	0.00	0.00	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.29	0.09	0.01	0.01	0.00	0.07	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Bmin	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.02	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00
Cdim	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.45	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
C#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Ddim	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.01	0.00	0.01	0.00	0.01	0.00	0.01	0.00	0.00	0.01	0.00	0.01	0.00	0.00	0.02	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.02	0.00	0.01	0.00	0.01	0.00	0.00
D#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.06	0.00	0.00	0.00	0.02	0.00	0.01	0.00	0.00	0.00	0.00
Edim	0.00	0.02	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.21	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00
Fdim	0.00	0.01	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.12	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00
F#dim	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.40	0.00	0.00	0.06	0.00	0.00	0.00	0.00
Gdim	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00
G#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.18	0.00	0.00	0.00	0.00	0.00
Adim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.25	0.00	0.00	0.00	0.00
A#dim	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00
Bdim	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.03	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.17	0.00	0.00
start	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00
stop	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.12	0.00	0.00	0.99

Multi-class Multi-label Classification

Κωδικοποιητής – Αποκωδικοποιητής – απλά LSTM

Όπως και στην περίπτωση του Multi-class classification, έτσι και εδώ το πρώτο μοντέλο που εξετάσαμε είναι το απλούστερο μοντέλο κωδικοποιητή – αποκωδικοποιητή. Εκπαιδεύσαμε το μοντέλο για διαφορετικές τιμές μεγέθους κελιού ώστε να αξιολογήσουμε την επίδραση του αριθμού εκπαιδεύσιμων παραμέτρων του μοντέλου στη διαδικασία εκπαίδευσης.

Train						
	macro			micro		
Αριθμός μονάδων	f1	precision	recall	f1	precision	recall
64	0.487210895	0.668163452	0.456054377	0.644943209	0.68967146	0.554825922
128	0.582089092	0.705985565	0.538685187	0.652869878	0.70840303	0.605410515

test						
	macro			micro		
Αριθμός μονάδων	f1	precision	recall	f1	precision	recall
64	0.487210894	0.668163452	0.456054376	0.616691628	0.66724467	0.533300678
128	0.537774482	0.638215897	0.499716143	0.620197488	0.67093607	0.554031889

Κωδικοποιητής – Αποκωδικοποιητής – αμφίδρομα LSTM

train						
	macro			micro		
Αριθμός μονάδων	f1	precision	recall	f1	precision	recall
64	0.582089092	0.705985564	0.538685186	0.652869878	0.70840303	0.605410514
128	0.606838366	0.720965887	0.560823057	0.672612096	0.70212453	0.659253047

test						
	macro			micro		
Αριθμός μονάδων	f1	precision	recall	f1	precision	recall

64	0,64866667	0,77283648	0,59967475	0.62858744	0.67466712	0.547007057
128	0.596014366	0.654667127	0.547007057	0.646523164	0.68633443	0.617009847

Κωδικοποιητής – Αποκωδικοποιητής – 2 επίπεδα απλών LSTM

test						
	macro			micro		
Αριθμός μονάδων	f1	precision	recall	f1	precision	recall
128	0.447833129	0.657304596	0.445460195	0.569353885	0.63273883	0.517511857

Κωδικοποιητής – Αποκωδικοποιητής – απλό LSTM – μηχανισμός προσοχής

dropout = 0

train						
	macro			micro		
Αριθμός μονάδων	f1	precision	recall	f1	precision	recall
64	0.461234483	0.657304596	0.445460195	0.628027657	0.69280611	0.549857099
128	0.582089092	0.705985564	0.538685186	0.670157504	0.71415825	0.618638396
256	0.606838366	0.720965887	0.560823057	0.695579855	0.71260631	0.665392578

test						
	macro			micro		
Αριθμός μονάδων	f1	precision	recall	f1	precision	recall
64	0.461234483	0.657304596	0.445460195	0.598271588	0.67686124	0.536033315
128	0.540555558	0.644030400	0.499728958	0.605024059	0.65987595	0.561263827
256	0.596014366	0.654667127	0.547007057	0.632773814	0.64991007	0.625815754

dropout = 0.4

train						
	macro			micro		
Αριθμός μονάδων	f1	precision	recall	f1	precision	recall
64	0.449304076	0.607549633	0.404963814	0.592304125	0.64454383	0.534205035
128	0.456520114	0.632121320	0.421754993	0.646794195	0.68010539	0.545656590
256	0.481126262	0.567071943	0.456485598	0.668353369	0.70253053	0.633059254

test						
	macro			micro		
Αριθμός μονάδων	f1	precision	recall	f1	precision	recall
64	0.419304076	0.597549633	0.404963814	0.573883463	0.63532842	0.487303743
128	0.456520114	0.632121320	0.421754993	0.594555356	0.64010575	0.535416423
256	0.481126262	0.567071943	0.456485598	0.642883623	0.67634242	0.633095765

4.4. Αξιολόγηση

4.4.1. Μετρικές

Για την αξιολόγηση των μοντέλων και της επίδρασης διαφόρων παραμέτρων των μοντέλων αυτών στα αποτελέσματα χρησιμοποιήθηκαν δύο ειδών μετρικές. Οι πρώτες είναι ενδεικτικές του πόσο καλά το μοντέλο κάνει το classification. Συγκεκριμένα είναι οι f1, precision και recall. Ο υπολογισμός τους γίνεται ως εξής:

$$precision = \frac{True\ positives}{True\ Positives + False\ Positives}$$

Η μετρική precision ουσιαστικά υπολογίζει το ποσοστό των σωστών κατηγοριοποιήσεων ως σωστών προς το σύνολο των κατηγοριοποιήσεων ως σωστών. Ουσιαστικά υπολογίζει την ακρίβεια του μοντέλου στο να κατηγοριοποιεί. Στην περίπτωσή μας, η συνάρτηση αυτή εκφράζει την ακρίβεια του μοντέλου, όταν αυτό προβλέπει μία νότα ή μία συγχορδία, κατά πόσο η πρόβλεψη αυτή ήταν σωστή.

$$recall = \frac{True\ positives}{True\ Positives + False\ Negatives}$$

Η μετρική recall υπολογίζει το ποσοστό των σωστών κατηγοριοποιήσεων ως σωστών σε σχέση με τον συνολικό αριθμό σωστών. Εκφράζει δηλαδή, από τις συνολικές φορές που εμφανίζεται μία νότα η μία συγχορδία στα δεδομένα, πόσες από αυτές προέβλεψε το μοντέλο.

$$F1 = 2 \times \left(\frac{Precision * Recall}{Precision + Recall} \right)$$

Εκτός από τις παραπάνω συναρτήσεις που εκφράζουν την ακρίβεια του κάθε μοντέλου, για την αξιολόγηση των μοντέλων χρησιμοποιήσαμε επιπλέον κάποιες μετρικές συναρτήσεις που αξιολογούν την ποιότητα του παραγόμενου αποτελέσματος από μουσική σκοπιά. Οι συναρτήσεις αυτές είναι οι παρακάτω:

- Empty bar rate – ποσοστό άδειων μέτρων
- Pitches used – χρησιμοποιημένοι τόνοι
- Pitch class used – χρησιμοποιημένες κλάσεις τόνων
- Qualified notes rate – ποσοστό ενδεδειγμένων νοτών
- Tonal distance – τονική απόσταση μελωδίας - αρμονίας

Για να αξιολογήσουμε τα αποτελέσματα των μοντέλων με αυτές τις μετρικές, συγκρίνουμε τις τιμές που προκύπτουν από την εφαρμογή αυτών των μετρικών στις προβλέψεις των μοντέλων με τις αντίστοιχες τιμές όταν αυτές οι μετρικές εφαρμόζονται στα αληθινά δεδομένα και ελέγχεται η διαφορά των δύο τιμών.

Η μετρική empty bar rate υπολογίζει το ποσοστό των μέτρων των παραγόμενων τραγουδιών που δεν ηχεί κάποια νότα / συγχορδία προς τον συνολικό αριθμό μέτρων. Δεδομένου ότι το μέτρο όλων των τραγουδιών είναι 4/4 και κάθε στοιχείο της ακολουθίας αντιστοιχεί σε αξία 1/16, το μήκος ακολουθίας που αντιστοιχεί σε ένα μέτρο είναι 16.

Η μετρική pitches used αποδίδει τον αριθμό διαφορετικών τόνων που χρησιμοποιούνται σε ένα κομμάτι.

Η μετρική pitch classes used αποδίδει τον αριθμό διαφορετικών νοτών που χρησιμοποιούνται σε ένα κομμάτι, χωρίς να διαχωρίζονται οι νότες που απέχουν διάστημα μίας οκτάβας μεταξύ τους

Τέλος, η μετρική qualified note rate υπολογίζει το ποσοστό των νοτών με χρονική αξία μεγαλύτερη ή ίση του 1/8 προς το σύνολο των νοτών. Η μετρική αυτή είναι χρήσιμη για να έχουμε μία αίσθηση σχετικά με το πόσο οι παραγόμενες συγχορδίες είναι κατακερματισμένες, δηλαδή διαρκούν για πολύ μικρο χρονικό διάστημα (1/16).

Η μετρική tonal distance δίνει μία αίσθηση του πόσο συμφωνεί/διαφωνεί η μελωδία με την αρμονία. Ο υπολογισμός της μετρικής αυτής γίνεται ως εξής:

Αρχικά υπολογίζεται ένα 6-διάστατο διάνυσμα ζ_n για κάθε χρονική στιγμή n ως εξής:

$$\zeta_n(d) = \frac{1}{\|c_n\|} \sum_{l=0}^{11} \Phi(d,l)\, c_n(l)$$

όπου c_n είναι το chromagram της μελωδίας/αρμονίας και Φ ένας πίνακας, ο οποίος αποτελεί βάση του 6-διάστατου διανυσματικού χώρου στον οποίο υπολογίζεται η τελική απόσταση. Η διαίρεση με την frobenius norm γίνεται για κανονικοποίηση των τιμών. Στη συνέχεια υπολογίζεται για κάθε χρονική στιγμή n η τονική απόσταση μεταξύ μελωδίας και αρμονίας ως εξής:

$$\xi_n = \sqrt{\sum_{d=0}^{5} \left[\zeta_n^{\mu\varepsilon\lambda\omega\delta\acute{\iota}\alpha} - \zeta_n^{\alpha\rho\mu o\nu\acute{\iota}\alpha} \right]^2}$$

Η τελική τιμή υπολογίζεται ως η μέση απόσταση για κάθε χρονική στιγμή n.

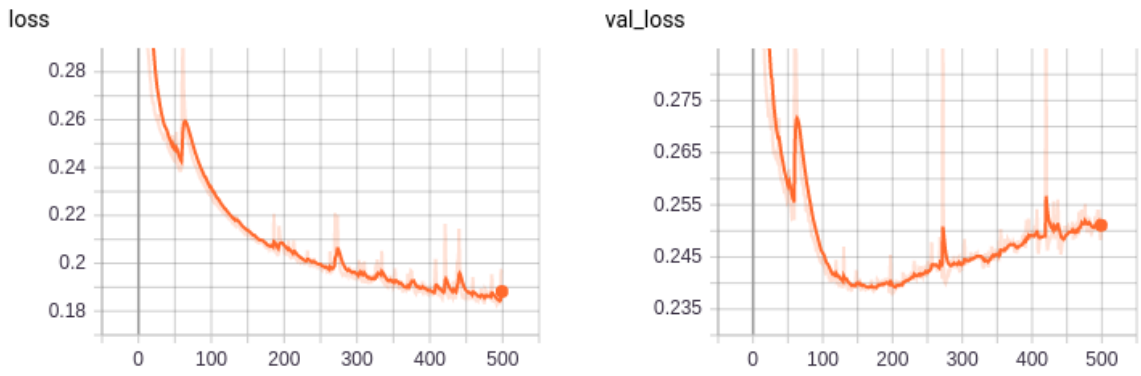
4.4.2. Επίδραση πόλωσης δεδομένων

Η ανισορροπία των δεδομένων του dataset είναι εμφανής στα αποτελέσματα των πειραμάτων. Πιο συγκεκριμένα, στην οι macro τιμές των μετρικών συναρτήσεων f1, precision και recall είναι σημαντικά χειρότερες σε σχέση με τις micro τιμές. Αυτό οφείλεται στο γεγονός ότι στον υπολογισμό των macro τιμών των συναρτήσεων αυτών δεν λαμβάνεται υπόψη ο αριθμός δειγμάτων που ανήκει στην κάθε κλάση αλλά αντίθετα υπολογίζεται ένας μέσος όρος μιας τιμής ακρίβειας. Αντίθετα, οι micro τιμές υπολογίζονται από τον σταθμισμένο μέσο όρο αυτών των τιμών ακρίβειας, όπου το βάρος της κάθε κλάσης εξαρτάται από τον αριθμό των δειγμάτων που ανήκουν σε αυτή. Άρα οι πιο συνηθισμένες κλάσεις συνεισφέρουν περισσότερο στο τελικό αποτέλεσμα.

Εκτός από τις macro και micro τιμές των μετρικών που δίνουν μία πρώτη εικόνα για τις επιπτώσεις της πόλωσης των δεδομένων στην επίδοση του μοντέλου, άλλη μία συνέπεια της ανισορροπίας αυτής είναι η πολύ μικρή ακρίβεια προβλέψεων στις κλάσεις οι οποίες δεν διαθέτουν ικανοποιητικό αριθμό δειγμάτων στο dataset. Με μία ματιά στον πίνακα σύγχυσης αντιλαμβάνεται κάποιος ότι στο κάτω δεξί μέρος του πίνακα η διαγώνιος σωστών προβλέψεων του μοντέλου εξαφανίζεται. Το μοντέλο αντί για αυτές τις συγχορδίες προβλέπει κάποιες άλλες που εμφανίζονται πολύ συχνότερα. Αντίστοιχα, κάποιες φορές που προβλέπει κάποιες λιγότερο συνηθισμένες συγχορδίες κάνει λάθος καθώς οι νευρώνες που οδηγούν στην επιλογή των συγχορδιών αυτών δεν έχουν εκπαιδευτεί επαρκώς ώστε να δίνουν σωστά αποτελέσματα.

4.4.3. Επίδραση μεγέθους κελιού

Η αύξηση του μεγέθους του κελιού του LSTM του μοντέλου κωδικοποιητή - αποκωδικοποιητή οδηγεί σε αύξηση του αριθμού των εκπαιδεύσιμων παραμέτρων του μοντέλου. Αυτό έχει ως συνέπεια το νευρωνικό δίκτυο να μπορεί να προσεγγίσει καλύτερα πιο περίπλοκες συναρτήσεις και να αντιληφθεί περισσότερα χαρακτηριστικά των δεδομένων. Η έλλειψη όμως επαρκούς αριθμού δεδομένων μπορεί να οδηγήσει σε υπερεκπαίδευση (overfitting) του μοντέλου, δηλαδή την εμφανώς καλύτερη επίδοση του μοντέλου στα δεδομένα εκπαίδευσης σε σχέση με τα δεδομένα ελέγχου.



Παρατηρώντας τα γραφήματα των συναρτήσεων απώλειας του συνόλου εκπαίδευσης και του συνόλου επικύρωσης διαπιστώνει κάποιος ότι παρά τη χρήση κανονικοποιητών όπως την τεχνική dropout, η συνάρτηση απωλειών του συνόλου επικύρωσης αρχίζει να γίνεται γνησίως αύξουσα, ενώ αντίθετα η συνάρτηση απώλειας του συνόλου εκπαίδευσης συνεχίζει να έχει αρνητική κλίση.

Παρόλα αυτά οι τιμές των μετρικών f1, precision και recall δείχνουν να βελτιώνονται όσο αυξάνουμε το μέγεθος κελιού του LSTM ανεξάρτητα από την αρχιτεκτονική που έχουμε χρησιμοποιήσει.

Multi-class (απλό μοντέλο κωδικοποιητή - αποκωδικοποιητή)

test				
	macro			micro
Αριθμός μονάδων	f1	precision	recall	f1
64	0.127674371	0.152470939	0.129681472	0.3663483538
128	0.137953742	0.137953742	0.139428769	0.3876604459
256	0.194147156	0.217176864	0.186085259	0.4500284464

Multi-label (απλό μοντέλο κωδικοποιητή – αποκωδικοποιητή)

test						
	macro			micro		
Αριθμός μονάδων	f1	precision	recall	f1	precision	recall
64	0.442918995	0.607421320	0.414594888	0.538909692	0.60658607	0.484818799
128	0.488885893	0.580196270	0.454287403	0.541831242	0.59176007	0.503665354

Αν εξετάσουμε τον πίνακα σύγχυσης 2 μοντέλων, ενός με μικρό αριθμό εκπαιδεύσιμων παραμέτρων και ενός μεγαλύτερου μοντέλου θα παρατηρήσουμε μερικές σημαντικές λεπτομέρειες.

Συγχορδία	Μικρό μέγεθος κελιού Σωστές προβλέψεις	Μεγάλο μέγεθος κελιού Σωστές προβλέψεις	Support
Cmaj	**11430**	8953	23641
C#maj	163	**171**	1466
Dmaj	94	**369**	1855
D#maj	439	**1586**	7157
Emaj	0	**120**	1064
Fmaj	3412	**4768**	15946
F#maj	39	**57**	390
Gmaj	3778	**6201**	16798
G#maj	2254	**3860**	12409

Amaj	39	**155**	1016
A#maj	2618	**3347**	10337
Bmaj	**1**	0	309
Cmin	**11669**	10929	22760
C#min	**23**	20	219
Dmin	254	**1133**	5930
D#min	**62**	39	335
Emin	156	**639**	3507
Fmin	484	**1210**	6615
F#min	0	0	153
Gmin	67	**1075**	5000
G#min	0	**16**	182
Amin	1785	**1839**	8727
A#min	1	**91**	471
Bmin	0	0	136
Cdim	0	**48**	202
C#dim	0	0	38
Ddim	**10**	3	1149
D#dim	0	0	32
Edim	0	**14**	291
Fdim	0	0	100
F#dim	0	**9**	348
Gdim	0	0	122
G#dim	0	0	43
Adim	0	**85**	437
A#dim	0	0	31
Bdim	0	**25**	584

Από τον παραπάνω πίνακα βλέπουμε ότι το υπερεκπαιδευμένο μοντέλο τα πηγαίνει αισθητά καλύτερα στην πρόβλεψη τον περισσότερων συγχορδιών σε σχέση με το μη υπερεκπαιδευμένο μοντέλο. Αντίθετα, το μη υπερεκπαιδευμένο μοντέλο τα πηγαίνει καλύτερα στις συγχορδίες με μεγάλο αριθμό δειγμάτων στο dataset. Αυτό σημαίνει ότι όσο περισσότερο εκπαιδεύεται το μοντέλο και όσο περισσότερες εκπαιδεύσιμες παραμέτρους προσθέτουμε σε αυτό μπορεί από τη μία να το οδηγεί στο overfitting, όμως το κάνει να μην διαλέγει τις πιο συχνές απαντήσεις που πιθανόν να οδηγούσαν σε αύξηση της απόδοσης του. Αυτό είναι συνέπεια και της ανισορροπίας του συνόλου δεδομένων καθώς ένα μοντέλο το οποίο δεν έχει εκπαιδευτεί επαρκώς και προβλέπει 3 – 4 συγχορδίες μόνο μπορεί να εμφανίσει καλή ακρίβεια στις προβλέψεις του, αν αυτές οι συγχορδίες καλύπτουν το μεγαλύτερο μέρος του dataset.

Παρόμοια αποτελέσματα παρατηρούνται και στην περίπτωση του Multi-label classification. Πάλι το μοντέλο που παρουσιάζει υπερεκπαίδευση σημειώνει καλύτερες προβλέψεις στην πλειοψηφία των νοτών εκτός από τις νότες που έχουν τη μεγαλύτερη συχνότητα εμφάνισης στο dataset.

Νότες	Μικρό μέγεθος κελιού True Positives	Μεγάλο μέγεθος κελιού True Positives	Support
C	**84481**	72547	96487
C#	351	**805**	4171
D	8668	**20494**	45200
D#	**33122**	29484	45542
E	15909	**21526**	39738
F	7421	**20235**	48147
F#	226	**666**	4030
G	**60860**	56522	85418
G#	1700	**8055**	23553
A	9315	**15205**	34643
A#	4869	**12964**	28940
B	2865	**8042**	23116

Τέλος, ελέγχουμε την επίδραση του μεγέθους κελιού με τις μετρικές αξιολόγησης των δειγματοληπτούμενων τραγουδιών. Διαπιστώνουμε σε γενικές γραμμές ότι η αύξηση του μεγέθους κελιού οδηγεί σε τιμές των μετρικών πιο κοντινές στις πραγματικές τιμές.

Multi-class

Μέγεθος κελιού	Empty bar rate	Pitches used	Pitch classes used	Qualified Notes	Tonal Distance
64	**0.9848042230**	5.7322673226	5.3927839278	0.9104932595	1.7268203278
128	0.9921330463	6.3456334563	5.9007790077	0.9285003925	1.6872805621
256	0.9926711767	**6.4723247232**	**6.0176301763**	**0.9372423234**	**1.5703302682**
Αληθινά δεδομένα	0.9642015170	6.5518655186	6.0754407544	0.9383285923	1.41395361105

Multi-label

Μέγεθος κελιού	Empty bar rate	Pitches used	Pitch classes used	Qualified Notes	Tonal Distance
64	0.9692753177	3.5666256662	3.4473144731	0.9190523894	1.6415749617
128	**0.9684040590**	**4.8343583435**	**4.5990159901**	**0.9224238932**	**1.6125686829**

Αληθινά δεδομένα	0.9642015170	6.7080770807	6.1832718327	0.9372693726	1.4243345421

4.4.4. Επίδραση LSTM διπλής κατεύθυνσης

Η επιλογή των LSTM διπλής κατεύθυνσης έγινε με τη λογική ότι οι νότες και οι συγχορδίες μίας μελωδίας εξαρτώνται όχι μόνο από τις νότες ή τις συγχορδίες που προηγούνται αυτών, αλλά πιθανόν και από αυτές που έπονται. Έτσι με τη χρήση αναδρομικών δικτύων διπλής κατεύθυνσης θα ήταν δυνατόν να εντοπιστεί αυτή η σχέση και να οδηγήσει σε καλύτερη απόδοση του δικτύου. Τα αποτελέσματα των πειραμάτων ήταν τα εξής:

Multi-class:

	F1 micro	Empty beat rate	Pitches used	Pitch classes used	Tonal Distance
64 αμφίδρομο	**0.403439**	**0.988314883148**	6.15703157031	5.77941779417	**1.64157496176**
128 απλό	0.379014	0.992133046330	**6.34563345633**	**5.90077900779**	1.68728056214

	F1 micro	Empty beat rate	Pitches used	Pitch classes used	Tonal Distance
128 αμφίδρομο	**0.4327737**	**0.98831488314**	**6.84583845838**	**6.34194341943**	**1.61256868299**
256 απλό	0.4207874	0.992671176711	6.47232472324	6.01763017630	1.64206492034

Multi-label:

	F1 micro	Empty beat rate	Pitches used	Pitch classes used	Tonal Distance
128 αμφίδρομο	**0.6630044**	**0.98831488314**	**5.84583845838**	**5.34194341943**	**1.64228764748**
256 απλό	0.6501974	0.99267117671	5.47232472324	5.01763017630	1.67492481067

Σε γενικές γραμμές, αν και οι αποκλίσεις είναι μικρές, τα LSTM δύο κατευθύνσεων δείχνουν να έχουν μεγαλύτερη ακρίβεια στο σύνολο ελέγχου από τα αντίστοιχα LSTM μίας κατεύθυνσης, αλλά δεν είναι απολύτως ασφαλές να βγάλουμε κάποιο συμπέρασμα λόγω των μικρών αποκλίσεων των τιμών των δύο μετρικών συναρτήσεων. Στο σύνολο εκπαίδευσης και ελέγχου που χρησιμοποιήσαμε οι ακολουθίες είχαν fixed μήκος ίσο με 66 το οποίο είναι σχετικά μικρό καθώς τα LSTM μπορούν να απομνημονεύσουν πληροφορίες από ακολουθίες πολύ μεγαλύτερες από αυτές που χρησιμοποιήσαμε στα πειράματά μας. Σε τέτοιες περιπτώσεις η επιλογή LSTM διπλής κατεύθυνσης θα ήταν σίγουρα καλύτερη καθώς παρά την ικανότητα των LSTM να απομνημονεύουν τις πιο σημαντικές πληροφορίες και να “ξεχνούν” τις λιγότερο σημαντικές, η χρήση LSTM διπλής κατεύθυνσης συμβάλει στο να μην χάνονται πληροφορίες από τα αρχικά στοιχεία της ακολουθίας.

4.4.5. Επίδραση μηχανισμού προσοχής

Ο μηχανισμός προσοχής παρέχει μια πιο πλούσια αναπαράσταση της εισόδου στον decoder ενώ παράλληλα του προσφέρει και έναν μηχανισμό εκμάθησης, για το που να εστιάζει την προσοχή του. Ενδιαφέρον παρουσιάζει όχι μόνο το κατά πόσο βελτιώνει την απόδοση του δικτύου ο μηχανισμός αυτός, αλλά και πως το πετυχαίνει αυτό. Για το λόγο αυτό δημιουργήσαμε ένα γράφημα που δείχνει τις τιμές που παίρνει ο πίνακας που περιέχει τα βάρη του μηχανισμού προσοχής, δηλαδή που σε ποια σημεία του διανύσματος περιεχομένου εστιάζει την προσοχή του το δίκτυο.

Συγκρίνουμε τα αποτελέσματα των μετρικών ακρίβειας και ποιότητας τραγουδιού μεταξύ του μοντέλου κωδικοποιητή – αποκωδικοποιητή με μηχανισμό προσοχής και χωρίς.

Multi-class

Μέγεθος κελιού: 64

	Χωρίς μηχανισμό προσοχής	Με μηχανισμό προσοχής
f1	0.366348354	**0.4327561906**
Empty bar rate	0.9848042230422305	**0.9833435834358344**
Pitches used	**5.732267322673227**	4.829028290282903
Pitch class used	**5.392783927839278**	4.63509635096351
Qualified notes	0.945786249878036	**0.939319393193932**
Tonal Distance	1.7268203278927112	**1.644934482553617**

Μέγεθος κελιού: 128

	Χωρίς μηχανισμό προσοχής	Με μηχανισμό προσοχής
f1	0.379014676	**0.451793496**
Empty bar rate	0.9921330463304633	**0.9862074620746207**
Pitches used	**6.345633456334563**	4.775727757277573
Pitch class used	**5.900779007790078**	4.605576055760557
Qualified notes	0.943738976786745	**0.939319393193932**
Tonal Distance	1.6872805621479765	**1.5410423878521197**

Μέγεθος κελιού: 256

	Χωρίς μηχανισμό προσοχής	Με μηχανισμό προσοχής
f1	0.420787483	**0.493759617**
Empty bar rate	0.9926711767117671	**0.9895448954489545**
Pitches used	**6.472324723247232**	5.992209922099221
Pitch class used	**6.017630176301763**	5.605986059860599

Qualified notes	0.943579845673456	**0.939319393193932**
Tonal Distance	**1.570330268244935**	**1.5282129681807603**

Multi-label

Μέγεθος κελιού: 64

	Χωρίς μηχανισμό προσοχής	Με μηχανισμό προσοχής
f1	0.64669163092894	**0.662659915**
precision	0.72790329058239	**0.748394087423489**
recall	0.58178255882394	**0.584763617242920**
Empty bar rate	**0.9692753177531775**	0.9784747847478474
Pitches used	**3.5666256662566624**	3.5243952439524393
Pitch class used	3.4473144731447314	**3.4792947929479294**
Qualified notes	0.9372693726937269	**0.939319393193932**
Tonal Distance	1.7352639248544042	**1.7062307562182988**

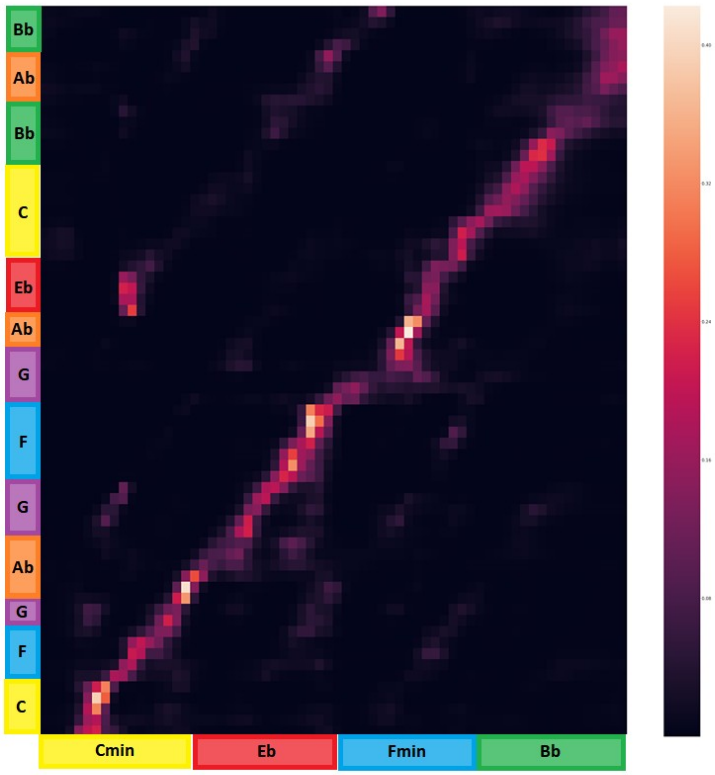
Μέγεθος κελιού: 128

	Χωρίς μηχανισμό προσοχής	Με μηχανισμό προσοχής
f1	0.65300448043458	**0.653466958234985**
precision	0.59176007226388	**0.620105079733868**
recall	**0.503665354290025**	0.485416085397485
Empty bar rate	0.9684040590405905	**0.9349118491184912**
Pitches used	**4.834358343583435**	4.451004510045101
Pitch class used	**4.599015990159901**	4.268142681426815
Qualified notes	0.938909389093891	**0.939319393193932**
Tonal Distance	1.69918336673662	**1.6232213698604266**

Παρατηρούμε ότι οι μετρικές ακρίβειας των μοντέλων με μηχανισμό προσοχής είναι λίγο πιο βελτιωμένες σε σχέση με τα μοντέλα χωρίς μηχανισμό προσοχής. Όσον αφορά της μετρικές υπολογισμού ποιότητας του μουσικού κομματιού δεν υπάρχει ξεκάθαρο συμπέρασμα καθώς οι αποκλίσεις εκατέρωθεν είναι πολύ μικρές.

Πέρα από την συμβολή του μηχανισμού προσοχής στην απόδοση του δικτύου, ενδιαφέρον παρουσιάζει και το που εστιάζει ο μηχανισμός αυτός. Για το λόγο αυτό δημιουργήσαμε ένα heatmap που απεικονίζει ακριβώς αυτή τη λειτουργία του μηχανισμού. Διαλέξαμε τυχαία μερικά τραγούδια και συγκρίναμε τα αποτελέσματα. Σε γενικές γραμμές τα αποτελέσματα και για το multi-class κια για το multi-label classification ήταν παρόμοια στα περισσότερα τραγούδια. Παρκάτω φαίνεται το heatmap του μηχανισμού προσοχής, η πιο ανοιχτόχρωμες περιοχές δηλώνουν τα

σημεία του context vector που εστιάζει την προσοχή του ο μηχανισμός ενώ οι σκουρόχρωμες δεν συμμετέχουν τόσο στη δημιουργία του attention hidden state.



Στο παραπάνω γράφημα φαίνεται σχετικά καθαρά η ύπαρξη μιας διαγώνιας γραμμής. Η αρχή των αξόνων είναι η στιγμή *t=0* , δηλαδή η αρχή του κομματιού. Η διαγώνιος αυτή δηλώνει ότι, τη χρονική στιγμή *t* το μοντέλο εστιάζει την προσοχή του κατά κύριο λόγο στην είσοδο που αντιστοιχεί στην ίδια χρονική στιγμή *t*. Ή επιλογή αυτή διαισθητικά φαίνεται δικαιολογημένη, αν σκεφτεί κανείς ότι μια συγχορδία επιλέγεται κυρίως με βάση τις νότες που ηχούν εκείνη τη στιγμή, λίγο νωρίτερα ή λίγο αργότερα. Πέρα από αυτό όμως, παρατηρούμε ότι υπάρχουν φωτεινά pixel κοντά στην διαγώνιο αυτή, σχηματίζοντας κάτι σαν τρίγωνα με υποτείνουσα την διαγώνιο, τα οποία σταματούν όταν ξεκινάει η επόμενη συγχορδία. Αυτό δηλώνει ότι ένα μέρος της προσοχής του αποκωδικοποιητή δίνεται στις προηγούμενες νότες που ηχούν όσο διαρκεί η συγχορδία. Και αυτό διαισθητικά φαίνεται σαν σωστή επιλογή καθώς για να αποφασίσουμε αν θα αλλάξουμε συγχορδία ή θα παραμείνουμε στην ίδια λαμβάνουμε κατά κύριο λόγο τις νότες που ηχούν στο συγκεκριμένο μέτρο. Τέλος, παρατηρούμε ότι το μοντέλο εστιάζει την προσοχή του και σε διάφορες νότες που ανήκουν εν τέλη στην προβλεπόμενη συγχορδία με κάπως πιο ακανόνιστο τρόπο αλλά και αυτό είναι δικαιολογημένο.

4.4.6. Επίδραση επιλογής μεθόδου δειγματοληψίας

Multi-class

Για την διεξαγωγή προβλέψεων από τα εκπαιδευμένα μοντέλα χρησιμοποιήσαμε μία ντετερμινιστική και μία μη ντετερμινιστική μέθοδο δειγματοληψίας. Η βασική διαφορά τους έγκειται στο γεγονός ότι στην ντετερμινιστική μέθοδο οι προβλέψεις του δικτύου είναι οι ίδιες πάντα ενώ στη μη ντετερμινιστική γίνεται τυχαία επιλογή από κάποια κατανομή πιθανοτήτων αλλάζοντας την έκβαση της πρόβλεψης κάθε φορά. Επίσης χρησιμοποιώντας την έννοια της θερμοκρασίας δοκιμάσαμε να κάνουμε το δίκτυο να κάνει πιο συντηρητικές ή τολμηρές προβλέψεις.

Θα συγκρίνουμε τα αποτελέσματα του απλού μοντέλου κωδικοποιητή – αποκωδικοποιητή που προκύπτουν από τις δύο μεθόδους δειγματοληψίας, της ντετερμινιστικής και της μη ντετερμινιστικής. Επιπλέον θα συγκρίνουμε τα αποτελέσματα που προκύπτουν όταν η θερμοκρασία παίρνει την τιμή 1 και την τιμή 2.

	Ντετερμινιστικό	Μη Ντετερμινιστικό, T=1.0	Μη Ντετερμινιστικό, T=2.0
Accuracy	0.493759617	0.293469968319648	0.236257839270705
Empy bar rate	0.989544895448	0.9939011890118902	0.9956437064370643
Pitches used	5.992209922099	6.519065190651906	7.813858138581386
Pitch classes used	5.605986059860	6.07380073800738	7.113161131611316
Qualified notes	0.939319393193	0.939319393193932	0.939319393193932
Tonal Distance	1.528212968180	1.673502963092383	1.840239589430082

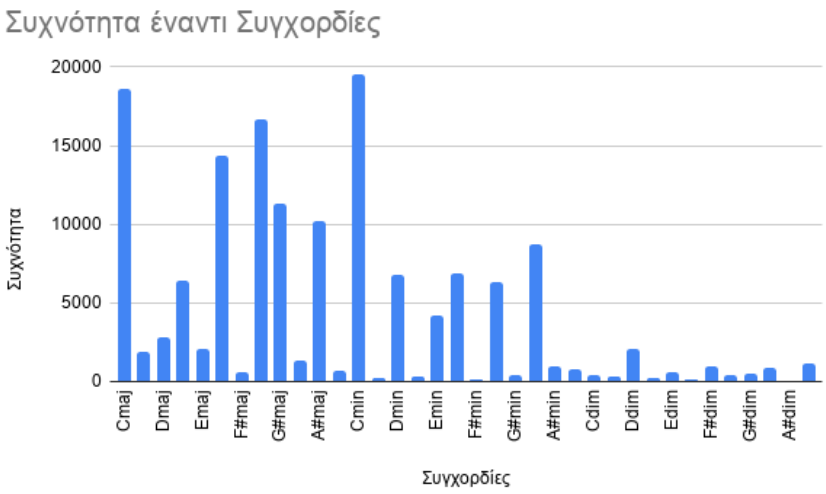
Ο παραπάνω πίνακας μετρικών επιβεβαιώνει τις προβλέψεις μας. Παρατηρούμε αρχικά ότι η ακρίβεια προβλέψεων είναι μικρότερη σε σχέση με τη ακρίβεια της ντετερμινιστικής επιλογής δειγμάτων. Αυτό είναι λογικό καθώς το μοντέλο έχει εκπαιδευτεί προσπαθώντας να μεγιστοποιήσει την ακρίβεια των προβλέψεων του. Η μη ντετερμινιστική μέθοδος δειγματοληψίας θα μπορούσε να παράγει αποτελέσματα μεγαλύτερης ακρίβειας αλλά αυτό το ενδεχόμενο συγκεντρώνει πολύ μικρή πιθανότητα. Έτσι, εξετάζοντας ένα σύνολο 2,5 χιλιάδων δειγμάτων το αποτέλεσμα που προέκυψε ήταν μείωση της ακρίβειας προβλέψεων. Επίσης παρατηρούμε ότι η αύξηση της τιμής της θερμοκρασίας της συνάρτησης *softmax* οδήγησε σε περαιτέρω μείωση της ακρίβειας των προβλέψεων, κάτι που είναι επίσης δικαιολογημένο καθώς η αύξηση της θερμοκρασίας οδηγεί σε αύξηση της πιθανότητας επιλογής μιας λιγότερο πιθανής και άρα λογικά λανθασμένης συγχορδίας. Παρ’ όλο το γεγονός ότι η μη ντετερμινιστική μέθοδος δειγματοληψίας οδήγησε σε μείωση της ακρίβειας των προβλέψεων η χρησιμότητα της δεν μειώνεται. Σε εφαρμογές όπως η εναρμόνιση μουσικής μελωδίας ή κάποια άλλη εφαρμογή που σχετίζεται με την αυτοματοποιημένη σύνθεση μουσικής η μη ντετερμινιστική επιλογή προβλέψεων είναι η πιο συνηθισμένη επιλογή. Αυτό συμβαίνει γιατί ο χαρακτηρισμός μίας πρόβλεψης (μια συγχορδίας) σε σωστή ή λανθασμένη δεν είναι τόσο απόλυτος όπως θα ήταν στην περίπτωση της αναγνώρισης μια εικόνας για παράδειγμα, όπου η ακρίβεια προβλέψεων είναι ένα αντικειμενικό κριτήριο για την επίδοση του μοντέλου. Στην περίπτωση της αυτόματης σύνθεσης μουσικής, η δυνατότητα το μοντέλο να προβλέπει λιγότερο “σωστές” συγχορδίες παρουσιάζει ενδιαφέρον λόγω της υποκειμενικότητας ως ένα βαθμό της

ορθότητας μία πρόβλεψης. Αυτό αποτυπώνεται και στα αποτελέσματα των μετρικών ποιότητας μουσικών τραγουδιών. Βλέπουμε ότι το μοντέλο επιλέγει μεγαλύτερη ποικιλία συγχορδιών και μάλιστα για T=2 ο αριθμός των διαφορετικών νοτών στις συγχορδίες που επιλέχθηκαν ξεπερνάει το 7. Αυτό σημαίνει ότι το μοντέλο συχνά επιλέγει συγχορδίες των οποίων οι νότες δεν βρίσκονται όλες στην βασική κλίμακα του κομματιού, καθώς μία κλίμακα (των βασικών τρόπων) αποτελείται από 7 νότες συνολικά. Αυτό αποτυπώνεται ακόμα καλύτερα στα παρακάτω γραφήματα. Πιο συγκεκριμένα, βλέπουμε ότι η κατανομή συχνότητας κάθε συγχορδίας αρχίζει να γίνεται πιο ομοιόμορφη, καθώς οι μεγαλύτερες συχνότητες μειώνονται ενώ ταυτόχρονα αυξάνονται οι μικρότερες συχνότητες.

Ντετερμινιστική επιλογή

Μη Ντετερμινιστική επιλογή, T=1

Μη Ντετερμινιστική επιλογή, T=2



Multi label

Για το multi label classification πρόβλημα, όπως προαναφέρθηκε, χρησιμοποιήσαμε δύο μεθόδους διεξαγωγής προβλέψεων, ντετερμινιστικές. Στην πρώτη χρησιμοποιούμε ένα κατώφλι με τιμή 0.5 και αποδεχόμαστε μία νότα αν η πιθανότητα που αντιστοιχεί στη νότα αυτή είναι μεγαλύτερη από την τιμή του κατωφλίου. Στην δεύτερη, χρησιμοποιούμε την παραπάνω τεχνική με τη διαφορά ότι αν η νότες με πιθανότητα μεγαλύτερη του κατωφλίου είναι λιγότερες από 3, αποδεχόμαστε τις 3 νότες με τις μεγαλύτερες πιθανότητες.

Παρακάτω παρουσιάζουμε τα αποτελέσματα των δύο μεθόδων για διάφορα μοντέλα που εξετάστηκαν στα πλαίσια της συγκεκριμένης διπλωματικής εργασίας.

Μοντέλο	Μέγεθος κελιού	Κατώφλι 0.5			Κατώφλι 0.5 + 3 μεγαλύτερα		
		F1	Precision	Recall	F1	Precision	Recall
Αμφίδρομα LSTM	64	0.541831242822078	0.595151934897412	0.49727914354507	0.530704079545502	0.535998937580762	0.52551280858935
	128	0.569566513049099	0.578485850273147	0.560918043628773	0.569073438530434	0.561423020610429	0.576935238862882
Απλό enc - dec	64	0.538909692388982	0.60658607537656	0.484818799020368	0.526490269247439	0.538874522575549	0.514662450527534
	128	0.544170400333155	0.59176007226388	0.503665354290025	0.535936795196636	0.54287574049809	0.529172996042203

Τα αποτελέσματα που φαίνονται στον παραπάνω πίνακα είναι αναμενόμενα και ξεκάθαρα. Χρησιμοποιώντας την υβριδική μέθοδο (συνδυασμός κατωφλίου και argmax(3)) η ακρίβεια (precision) των προβλέψεων μειώνεται ενώ αυξάνεται η μετρική recall. Αυτό συμβαίνει γιατί με

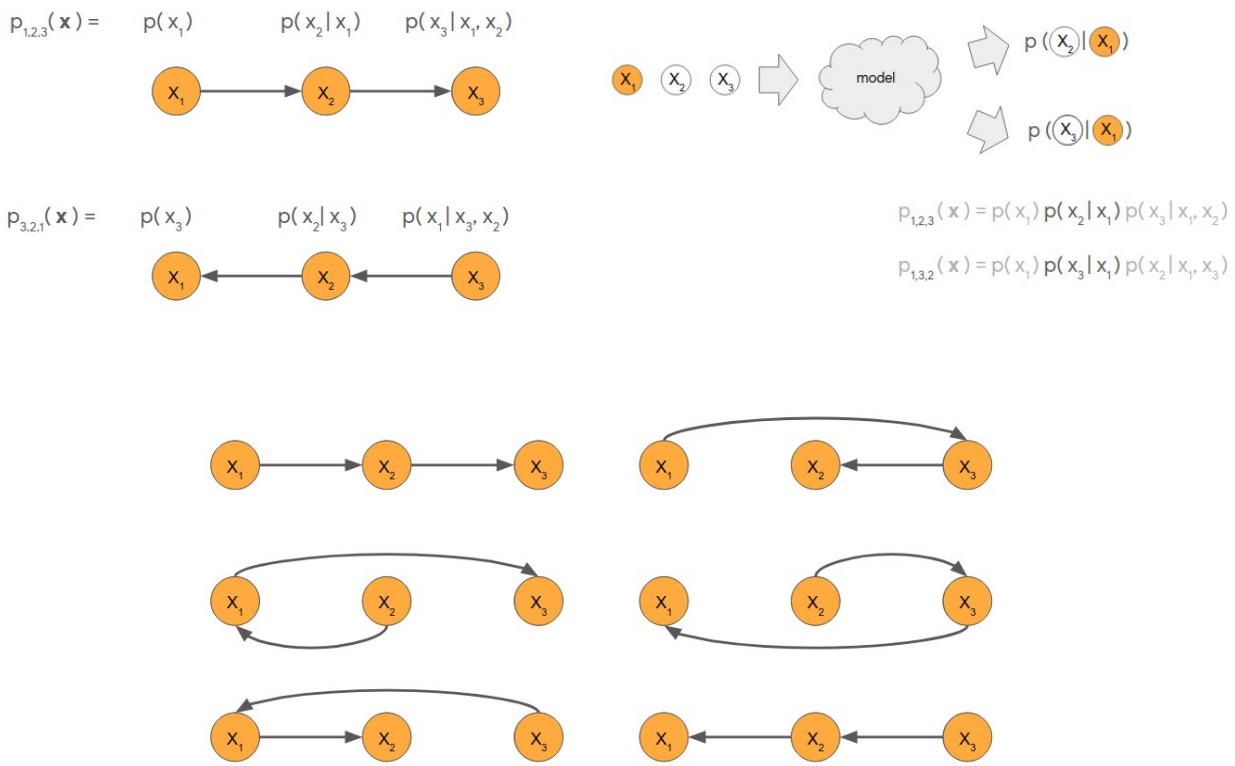
αυτή τη στρατηγική δειγματοληψίας επιλέγουμε σαν νότες που δεν είναι τόσο σίγουρες, ή για να είμαστε ακριβείς απλά είναι λίγο πιο πιθανές να ανήκουν στην συγχορδία που παράγεται απ' ότι οι υπόλοιπες νότες. Από τα αποτελέσματα προκύπτει ότι οι νότες που επιλέγονται με αυτόν τον τρόπο πολύ συχνά είναι λανθασμένες και αυτό έχει ως αποτέλεσμα να μειώνεται η τιμή της μετρικής precision. Πέρα π' αυτό, όμως, οι σωστές προβλέψεις, αν και σπάνιες) αυξάνουν την τιμή του recall, αλλά όχι σε τέτοιο βαθμό ώστε να αυξηθεί και η τιμή του αρμονικού μέσου, f1.

Η επιλογή της καλύτερης μεθόδου έχει να κάνει με το αποτέλεσμα που θέλουμε να πετύχουμε. Τυπικά , αν εξετάσουμε την τιμή f1, η απλή μέθοδος με χρήση κατωφλίου είναι καλύτερη. Αλλά πιο σημαντικό στην επιλογή μεθόδου δειγματοληψίας είναι η αξιολόγηση του τι προσφέρει η κάθε μέθοδος σε σύγκριση με την άλλη και η αξιολόγηση των πλεονεκτημάτων και μειονεκτημάτων τις κάθε μίας μεθόδου. Στην συγκεκριμένη περίπτωση μπορεί η απλή μέθοδος να παρουσιάζει μεγαλύτερη ακρίβεια αλλά πολλές φορές το παραγόμενο αποτέλεσμα δεν συνιστά συγχορδία καθώς αποτελείται από μία η και καμία νότα. Αξιολογώντας αυτό σαν σημαντικό μειονέκτημα θα λέγαμε πως η υβριδική μέθοδος, αν και με μικρότερη ακρίβεια στις προβλέψεις πλεονεκτεί στο γεγονός ότι το παραγόμενο αποτέλεσμα είναι πιο αληθοφανές.

5.Εναρμόνιση μελωδίας με συνελικτικά νευρωνικά δίκτυα (CNN)

5.1. Σχετικά με το Coconet

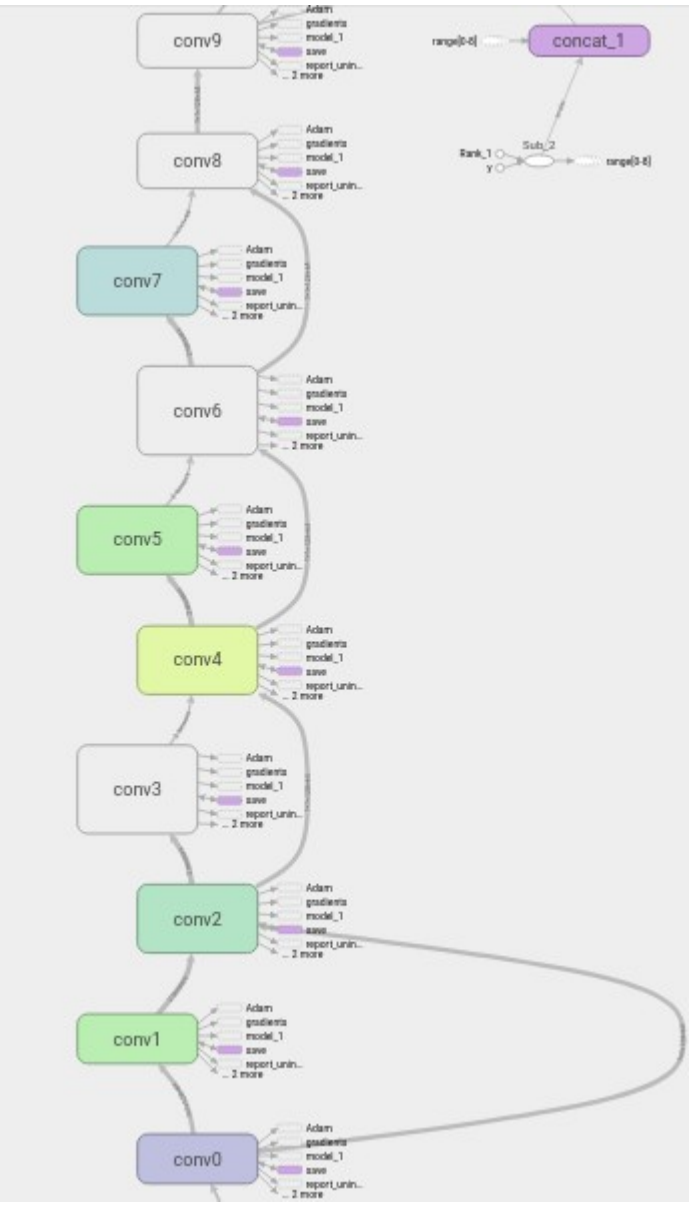
Το δεύτερο μοντέλο που εξετάσαμε στα πλαίσια της συγκεκριμένης διπλωματικής εργασίας είναι ένα συνελικτικό νευρωνικό δίκτυο, το Coconet. Το Coconet είναι ένα βαθύ συνελικτικό νευρωνικό δίκτυο που εκπαιδεύεται ώστε να μπορεί να ανακατασκευάζει ελλειπή μουσικά κομμάτια. Εφόσον εκπαιδευτεί, το Coconet μπορεί να υπολογίζει τις δεσμευμένες πιθανότητες της μορφής $p(x_i|x_C)$ όπου το C είναι ένα μέρος του μουσικού κομματιού *x* και το κάθε *i* ανήκει στο συμπλήρωμα του C. Ουσιαστικά το σύνολο C είναι οι δεδομένες νότες πάνω στις οποίες πρέπει να βασιστεί το Coconet για να κάνει τις προβλέψεις του για το κάθε *i*. Το Coconet είναι ένα παράδειγμα ενός Νευρωνικού αυτοαναδρομικού εκτιμητής κατανομής (orderless Neural Autoregressive Distribution Estimator, orderless NADE). Οι εκτιμητές αυτοί ειδικεύονται στο να μαθαίνουν ένα σύνολο παραγοντοποιήσεων για τον υπολογισμό μίας δεσμευμένης πιθανότητας p(x) για διαφορετικές διατάξεις των επιμέρους πιθανοτήτων. Για το λόγο αυτό, το Coconet δεχόμενο σαν είσοδο ένα μέρος ενός μουσικού κομματιού (στην περίπτωση της εργασίας μας μία φωνή που αντιστοιχεί στη μελωδία του κομματιού) συμπληρώνει το υπόλοιπο κομμάτι βάση μια κατανομής πιθανοτήτων που υπολογίζει λαμβάνοντας υπόψη διαφορετικές διατάξεις επιλεγομένων νοτών. Στις παρακάτω εικόνες φαίνονται οι διαφορετικές διατάξεις για την πρόβλεψη μίας μελωδίας τριών νοτών.



Για την διαδικασία δειγματοληψίας από το εκπαιδευμένο μοντέλο, το Coconet χρησιμοποιεί την μέθοδο Gibbs sampling, μία Markov Chain Monte Carlo μέθοδο που μπορεί να κάνει δειγματοληψία από μία κατανομή δεσμευμένων πιθανοτήτων κάνοντας επαναλαμβανόμενες επιλογές από τυχαίες μεταβλητές σύμφωνα με τις κατανομές τους. Η μέθοδος αυτή έχει αναλυθεί εκτενώς στο κεφάλαιο 2.

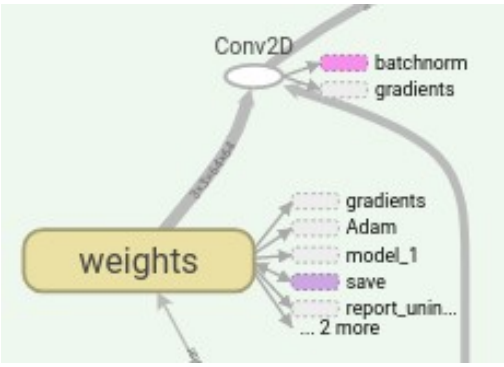
5.2. Αρχιτεκτονική νευρωνικού δικτύου

Από άποψη αρχιτεκτονικής, το Coconet είναι ένα συνελικτικό νευρωνικό δίκτυο με υπολειμματικές συνδέσεις (residual connections), επίπεδα διαχωριζομένων συνελίξεων (separable convolutions) και διασταλμένες συνελίξεις (dilated convolutions).

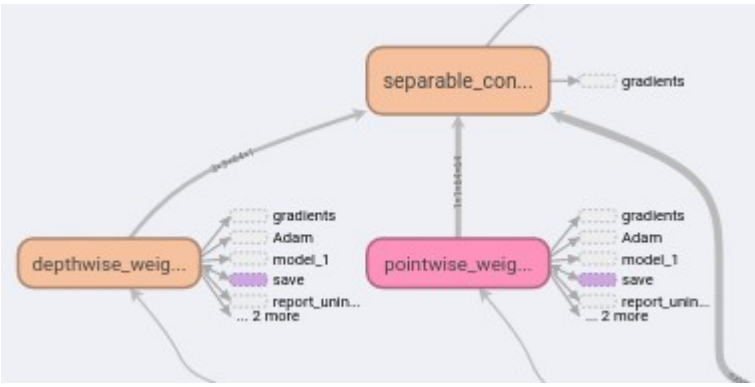


Το δίκτυο δέχεται σαν είσοδο διανύσματα διαστάσεων (batch_size, song_duration, 4), όπου η τιμή 4 αναφέρεται στα κανάλια της εισόδου που αντιστοιχούν στις 4 φωνές που αποτελούν το κάθε μουσικό κομμάτι. Η είσοδος αυτή μετατρέπεται σε pianoroll, δηλαδή ένα διάνυσμα διαστάσεων (batch_size, song_duration, pitch_range, 4), όπου η τιμή pitch_range αναφέρεται στο εύρος νοτών. Για την εκπαίδευση του δικτύου επιλέχθηκε η τιμή 128. Έτσι η είσοδος έχει πάρει τη μορφή ενός 1-hot encoded vector, όπου η τιμή 1 στο timestep t με τιμή pitch 60 δηλώνει ότι τη χρονική στιγμή t ηχεί η νότα με τιμή pitch 60. Εκτός από την μετατροπή σε pianoroll, πριν το πρώτο συνελικτικό επίπεδο υπάρχει ακόμα ένα concatenation layer στο οποίο η είσοδος του δικτύου γίνεται concatenate με 4 επιπλέον κανάλια που λειτουργούν ως μάσκες. Ο ρόλος αυτών των καναλιών είναι να υποδεικνύουν αν η νότα την χρονική στιγμή όπου η μάσκα έχει την τιμή 1 ή 0 έχει γνωστή τιμή ή όχι αντίστοιχα, άρα θα πρέπει το μοντέλο να κάνει κάποια πρόβλεψη για την τιμή της. Αυτό συμβαίνει για να γνωρίζει το μοντέλο τις ήδη καθορισμένες νότες έτσι ώστε να της συμπεριλάβει οπωσδήποτε στο τελικό αποτέλεσμα κατά τη διάρκεια της δειγματοληψίας.

Το δίκτυο αποτελείται συνολικά από 10 συνελικτικά επίπεδα. Κάθε συνελικτικό επίπεδο τροφοδοτεί το επόμενο επίπεδο αλλά και το μεθεπόμενο με μία υπολειμματική σύνδεση (residual connection). Στο δίκτυο χρησιμοποιούνται συνολικά απλές αλλά και διαχωριζόμενες συνελίξεις. Πιο συγκεκριμένα, στα δύο πρώτα συνελικτικά επίπεδα χρησιμοποιούνται απλές συνελίξεις ενώ στα υπόλοιπα επίπεδα γίνεται χρήση διαχωριζομένων συνελίξεων (separable convolutions).



Σε κάθε συνελικτικό επίπεδο απλής συνέλιξης πραγματοποιείται μία συνελικτική πράξη με πυρήνα συνέλιξης διαστάσεων (3, 3, αριθμός καναλιών, 64). Επίσης χρησιμοποιείται padding τέτοιο ώστε να μην μεταβάλλονται οι χωρικές διαστάσεις της εισόδου του συνελικτικού φίλτρου. Χρησιμοποιήσαμε dilated convolutions σε κάθε συνελικτικό επίπεδο με βήμα διαστολής 1. Το βήμα συνέλιξης που χρησιμοποιήσαμε είναι επίσης 1.



Όπως προαναφέρθηκε, μόνο στα δύο πρώτα συνελικτικά επίπεδα χρησιμοποιήθηκαν απλές συνελίξεις. Στα υπόλοιπα οκτώ συνελικτικά επίπεδα χρησιμοποιήθηκαν διαχωριζόμενες εις βάθος

συνελίξεις (depthwise separable convolutions) με σκοπό την επιτάχυνση των υπολογισμών και της εκπαίδευσης του δικτύου κατ' επέκταση. Όπως φαίνεται και στο παραπάνω διάγραμμα που απεικονίζει την αρχιτεκτονική του συνελικτικού επιπέδου, η εις βάθος συνέλιξη πραγματοποιείται μεταξύ της εισόδου του φίλτρου και του συνελικτικού πυρήνα διαστάσεων (3, 3, 1). Χρησιμοποιούνται συνολικά τόσα εις βάθος συνελικτικά φίλτρα όσος και ο αριθμός καναλιών της εισόδου του συνελικτικού επιπέδου, στην περίπτωσή μας 64, δημιουργώντας ένα φίλτρο διαστάσεων (3, 3, 64, 1). Αντίστοιχα, η σημειακή συνέλιξη γίνεται μεταξύ της εισόδου του φίλτρου και του συνελικτικού πυρήνα διαστάσεων (1, 1, 64), όπου ο αριθμός 64 αντιστοιχεί στον αριθμό καναλιών της εισόδου. Χρησιμοποιούμε συνολικά τόσα τέτοια φίλτρα όσο ο επιθυμητός αριθμός καναλιών της εξόδου. Εμείς επιθυμούμε να διατηρούμε τον αριθμό καναλιών σταθερό και ίσο με 64, για το λόγο αυτό χρησιμοποιούνται 64 τέτοια φίλτρα. Για το λόγο αυτό προκύπτουν και οι τελικές διαστάσεις του φίλτρου που φαίνεται στο σχήμα οι οποίες είναι (1, 1, 64, 64).

Σε κάθε συνελικτικό επίπεδο, εκτός από το συνελικτικό φίλτρο υπάρχει επιπλέον και ένα pooling φίλτρο, συγκεκριμένα maximum pooling φίλτρο, μεταβλητών διαστάσεων για κάθε επίπεδο. Πριν το pooling layer υπάρχει ένα activation layer, στο οποίο εφαρμόζεται η μη – γραμμική συνάρτηση ενεργοποίησης *ReLU* (Rectified Linear Unit) στην έξοδο του συνελικτικού φίλτρου. Επίσης, σε κάθε layer του δικτύου εφαρμόζεται η τεχνική batch normalization.

Το τελευταίο επίπεδο του δικτύου είναι ένα επίπεδο ενεργοποίησης που εφαρμόζει την συνάρτηση ενεργοποίησης *softmax* για την παραγωγή των προβλέψεων του δικτύου. Η συνάρτηση απωλειών που υπολογίζεται είναι η softmax cross entropy (γνωστή και ως categorical cross entropy). Ο βελτιστοποιητής που χρησιμοποιείται είναι ο Adam Optimizer.

5.3. Εκπαίδευση δικτύου

Εκπαιδεύσαμε το δίκτυο πάνω στα ίδια δεδομένα που εκπαιδεύσαμε και το αναδρομικό μοντέλο. Η κατανομή του συνόλου των δεδομένων σε σύνολο εκπαίδευσης, επικύρωσης και ελέγχου είναι 70% - 15% -15%.

Η παραπάνω αρχιτεκτονική είναι η μοναδική αρχιτεκτονική που χρησιμοποιήσαμε για την εκπαίδευση του δικτύου. Δεν δοκιμάσαμε διαφορετικές αρχιτεκτονικές ή διαφορετικές τιμές υπερπαραμέτρων καθώς ο σκοπός της εκπαίδευσης του συγκεκριμένου δικτύου ήταν η παραγωγή αποτελεσμάτων που θα χρησιμοποιηθούν για σύγκριση με τα αποτελέσματα που προέκυψαν από το αναδρομικό μοντέλο.

5.4. Σύγκριση με αναδρομικό μοντέλο

Για να συγκρίνουμε τα δύο μοντέλα χρησιμοποιήσαμε τις ίδιες μετρικές ποιότητας μουσικού κομματιού που χρησιμοποιήσαμε και για την αξιολόγηση των διαφόρων αναδρομικών νευρωνικών δικτύων που εξετάσαμε στα πλαίσια της διπλωματικής αυτής εργασίας. Για τη σύγκριση χρησιμοποιήσαμε το αναδρομικό μοντέλο που είχε τις καλύτερες τιμές στις συγκεκριμένες μετρικές ανεξαρτήτως ακρίβειας, συγκεκριμένα το μοντέλο κωδικοποιητή αποκωδικοποιητή με αμφίδρομα LSTM και μέγεθος κελιού 128. Η σύγκριση έγινε μεταξύ του μοντέλου αυτού, του μοντέλου Coconet που εκπαιδεύσαμε πάνω στα δεδομένα που συλλέξαμε από το HookTheory και ένα ήδη εκπαιδευμένο μοντέλο Coconet πάνω σε ένα σύνολο δεδομένων από 306 Chorales του J.S. Bach. Παρακάτω παρουσιάζονται τα αποτελέσματα των μετρικών που προκύπτουν για τα 3 μοντέλα για 3 διαφορετικές τιμές θερμοκρασίας δειγματοληψίας.

Για temperature = 0.1:

	Coconet (Hooktheory)	Coconet (JSB)	Seq2Seq (Hooktheory)	Αληθινά δεδομένα (Hooktheory)
EBR	0.9889204992010	0.9823049529423	0.9926711767117	0.9642015170151
PU	6.4829394258902	6.7423901993209	6.4723247232472	6.5518655186551
PCU	6.0385290234959	6.1252932439400	6.0176301763017	6.0754407544075
QNR	9.3458929348592	0.9305230643923	0.9393193931939	0.9425384395029
TD	1.6038912059302	1.5442952367103	1.5312129681807	1.4243345421489

Για temperature = 1.0:

	Coconet (Hooktheory)	Coconet (JSB)	Seq2Seq (Hooktheory)	Αληθινά δεδομένα (Hooktheory)
EBR	0.9893235892323	0.9845238295837	0.9939011890118	0.9642015170151
PU	6.6532693240259	6.9582934195290	6.5190651906519	6.5518655186551
PCU	6.1245200953258	6.3298293480529	6.0738007380073	6.0754407544075
QNR	0.9303240492358	0.9256630934523	0.9283527848928	0.9425384395029
TD	1.7052398410384	1.6624043689439	1.6798527834983	1.4243345421489

Για temperature = 2.0:

	Coconet (Hooktheory)	Coconet (JSB)	Seq2Seq (Hooktheory)	Αληθινά δεδομένα (Hooktheory)
EBR	0.9912452458392	0.9882349845923	0.9956437064370	0.9642015170151
PU	7.9823203986492	8.1843942382893	7.8138581385813	6.5518655186551
PCU	7.1758239823498	7.2582348582393	7.1131611316113	6.0754407544075
QNR	0.9239589594934	0.9218593284392	0.9393193931939	0.9425384395029
TD	1.7852385209583	1.7206394810642	1.8492039893249	1.4243345421489

6. Συμπεράσματα

Στην συγκεκριμένη διπλωματική εργασία προσεγγίσαμε το έργο της παραγωγής συγχορδιών για μια δοθείσα μελωδία δίνοντας βάρος στην ακολουθιακή φύση της μουσικής. Για το λόγο αυτό επικεντρωθήκαμε στην εξέταση μοντέλων αναδρομικών νευρωνικών δικτύων που χαρακτηρίζονται από την ικανότητα τους να διαχειρίζονται ακολουθιακά δεδομένα. Σε γενικές γραμμές, δεν συνηθίζεται να χρησιμοποιούνται οι αρχιτεκτονικές που χρησιμοποιήθηκαν για την πρόβλεψη, αλλά για την εξαγωγή χαρακτηριστικών. Όταν επιχειρείται το έργο με χαρακτηριστικά υψηλού επιπέδου της μορφής της εργασίας, στη βιβλιογραφία χρησιμοποιούνται Κρυφά Μοντέλα Μαρκόβ (Hidden Markov Models), Βαθιά Δίκτυα Εμπιστοσύνης (Deep Belief Networks) και άλλα μοντέλα που δεν εξετάστηκαν στην παρούσα εργασία.

Υπήρχε σημαντικός περιορισμός ως προς τα δεδομένα και από άποψη ποσότητας αλλά και ποιότητας. Από το μία, η έλλειψη μεγάλου όγκου δεδομένων δυσχέρανε το κομμάτι της εκπαίδευσης των μοντέλων που εξετάσαμε καθώς συχνά εμφανιζόταν φαινόμενα υπερεκπαίδευσης των δικτύων, ιδιαίτερα αυτών με μεγαλύτερο αριθμό εκπαιδεύσιμων παραμέτρων. Από την άλλη, η πόλωση των δεδομένων καθιστά δύσκολη την εκπαίδευση των μοντέλων υπό την έννοια ότι από τη μία δεν υπάρχουν αρκετά δεδομένα από κάποιες κλάσεις έτσι ώστε να μάθει το δίκτυο να κάνει σωστές προβλέψεις για τις κλάσεις αυτές ενώ από την άλλη η υπερβολική συσσώρευση δεδομένων σε κάποιες κλάσεις κάνει τα δίκτυα να περιορίζουν τις προβλέψεις τους στις κλάσεις αυτές. Για το λόγο αυτό, το μοντέλο που χρησιμοποιήσαμε για το multi label classification, στο dataset του οποίου η πόλωση των δεδομένων ήταν σχετικά μικρότερη, επέδειξε καλύτερα αποτελέσματα και από πλευράς ακρίβειας προβλέψεων και από πλευράς αποφυγής του overfitting. Θεωρούμε ότι στην περίπτωση του multi class single label classification προβλήματος θα ήταν καλύτερο να παραληφθούν οι ελαττωμένες συγχορδίες λόγω τις πολύ μικρής συμμετοχής τους στο dataset και να παραλειφθούν τελείως είτε να αντικατασταθούν από κάποιες συγχορδίες που επιτελούν τον ίδιο σκοπό από πλευράς λειτουργικής αρμονίας.

Επίσης, από πλευράς αρχιτεκτονικής των δικτύων, από τα αποτελέσματα των πειραμάτων προκύπτει ότι η χρήση αμφίδρομων LSTM βελτιώνει την επίδοση των μοντέλων, έστω και σε μικρό βαθμό. Η χρήση πολλαπλών επιπέδων LTSM από την άλλη, όχι μόνο δε βελτίωσε την επίδοση των μοντέλων αλλά μάλλον λειτούργησε ανασταλτικά για την βελτίωση της ακρίβειας των προβλέψεων των δικτύων. Στην εργασία χρησιμοποιήθηκαν 2 επίπεδα LTSM για κωδικοποιητή και αποκωδικοποιητή, κάποιος άλλος συνδυασμός ίσως να μπορούσε να αποφέρει καλύτερα αποτελέσματα. Τέλος, ο μηχανισμός προσοχής βελτίωσε την επίδοση των δικτύων αλλά περισσότερο για μικρά και μέτρια μεγέθη κελιών LSTM. Επίσης βοήθησε στην αποφυγή του overfitting.

Ως προς τις μεθόδους δειγματοληψίας μου εξετάσαμε, στις εφαρμογές όπως η αυτόματη εναρμόνιση μελωδίας η χρήση μη ντετερμινιστικών μεθόδων είναι σαφώς προτιμότερη και σίγουρα πιο διαδεδομένη καθώς για το έργο αυτό θέλουμε τα μοντέλα να μην κάνουν πάντα τις ίδιες προβλέψεις σε κάθε εκτέλεση. Στην περίπτωση του multi label classification, θεωρούμε ότι είναι προτιμότερη η υβριδική μέθοδος που χρησιμοποιήσαμε για τη διεξαγωγή προβλέψεων καθώς προβλέπει πάντα έναν ελάχιστο αριθμό νοτών ώστε οι αρμονία των κομματιών να είναι “γεμάτη” παρόλο που η ακρίβεια των προβλέψεων μειώνεται.

Συγκριτικά με το Coconet, το sequence to sequence μοντέλο είχε αρκετά καλά αποτελέσματα, πιο κοντινά στα αληθινά δεδομένα αλλά λίγο περισσότερο συντηρητικό στις προβλέψεις του. Γενικά οι μετρικές ποιότητας μουσικού κομματιού δίνουν μία αίσθηση για το πόσο κοντά στα αληθινά δεδομένα είναι οι προβλέψεις κάποιου μοντέλου αλλά δεν μπορούν να απεικονίσουν με ακρίβεια την ποιότητα του παραγόμενου αποτελέσματος. Ιδανικός τρόπος αξιολόγησης θα ήταν η

αξιολόγηση από ανθρώπους, κάτι που χρησιμοποιείται ευρέως για την αξιολόγηση σύνθετων δικτύων πάνω σε εφαρμογές μουσικής παραγωγής.

7. Υλοποίηση

Όλα τα πειράματα και τα σενάρια (scripts) για την συλλογή των δεδομένων, την προ επεξεργασία τους, την κατασκευή την εκπαίδευση και την πρόβλεψη των μοντέλων, αλλά και την εξαγωγή των αποτελεσμάτων υλοποιήθηκαν στην γλώσσα προγραμματισμού Python.

Συγκεκριμένα για την προεπεξεργασία των δεδομένων χρησιμοποιήθηκε η βιβλιοθήκη numpy και η MiniDom για την εξαγωγή δεδομένων από XML αρχεία. Για τη δημιουργία των συνόλων δεδομένων πάνω στα οποία εκπαιδεύτηκαν τα διάφορα μοντέλα χρησιμοποιήθηκε η βιβλιοθήκη numpy της Python. Συγκεκριμένα για την κατασκευή και την εκπαίδευση των μοντέλων χρησιμοποιήθηκε το framework που έχει αναπτύξει η Google ειδικά για την μηχανική μάθηση με νευρωνικά δίκτυα, το Tensorflow. Για το μοντέλο κωδικοποιητή – αποκωδικοποιητή χρησιμοποιήθηκε το framework Keras το όποια στηρίζεται στο Tensorflow και κάνει την συγγραφή συνθέτων αρχιτεκτονικών πιο εύκολη. Τα παραπάνω εργαλεία χρησιμοποιούνται ευρέως στον χώρο της μηχανικής μάθησης παρέχοντας βοηθητικές συναρτήσεις για την γρήγορη εκπαίδευση, τον υπολογισμό μετρικών καθώς και την επεξεργασίας μεγάλου όγκου δεδομένων με την χρήση καρτών γραφικών (GPU’s), αν αυτές υπάρχουν. Έτσι ο χρηστής δεν σπαταλά χρόνο στην σχεδίαση και την συγγραφή κώδικά για την εκμετάλλευση των πόρων του συστήματος του, μιας και την δουλεία αυτή την αναλαμβάνουν τα παραπάνω Frameworks. Συγκεκριμένα η βιβλιοθήκη Tensorflow χρησιμοποιεί την διεπαφή Cuda, η όποια έχει αναπτυχθεί από την κατασκευάστρια καρτών γραφικών NVIDIA και έτσι μπορεί να τρέχει το μοντέλο σε μια ή περισσότερες GPU, αντί της CPU, μειώνοντας τελικά κατά πολύ τον χρόνο εκτέλεσης. Η βελτίωση της ταχύτητας έρχεται ως αποτέλεσμα της εκτέλεσης παράλληλου κώδικα που προσφέρεται από τις κάρτες γραφικών, ανάγκη η οποία συναντάται πολύ έντονα στην εκπαίδευση νευρωνικών δικτύων. Για την σελίδα αξιολόγησης χρησιμοποιήθηκε το framework για web development Flask. Αυτό είναι υπεύθυνο για την διαχείριση των requests, την λειτουργία του backend στον server κ.α. Το backend ουσιαστικά είναι πανομοιότυπο με τα μοντέλα όπως αυτά περιγράφηκαν παραπάνω, ενώ για το frontend χρησιμοποιήθηκε το framework Bootstrap. Για την δημιουργία, επεξεργασία midi αρχείων ή την εξαγωγή πληροφοριών από midi αρχεία χρησιμοποιήθηκαν οι βιβλιοθήκες mido και PrettyMidi της Python. Για την εύκολη διαχείριση των πινάκων που αναπαριστούν ακολουθίες νοτών και συγχορδιών χρησιμοποιήθηκε η βιβλιοθήκη PyPianoroll.

Βιβλιογραφία

[1] Understanding LSTM Networks, August 27, 2015 Link: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

[2] Guide to LSTMs and Recurrent Neural Networks Chris Nicholson, Link: https://pathmind.com/wiki/lstm

[3] Coconet: the ML model behind today's Bach Doodle Huang, Cheng-Zhi Anna and Cooijmans, Tim and Roberts, Adam and Courville, Aaron and Eck Douglas, Link: https://magenta.tensorflow.org/coconet

[4] Training recurrent neural networks, Ilya Sutskever 2013, Link: http://www.cs.utoronto.ca/~ilya/pubs/ilya_sutskever_phd_thesis.pdf

[5] Understanding Encoder-Decoder Sequence to Sequence Model, Simeon Kostadinov, Link: https://towardsdatascience.com/understanding-encoder-decoder-sequence-to-sequence-model-679e04af4346

[6] Denny Britz, Attention and Memory in Deep Learning and NLP , Link: http://www.wildml.com/2016/01/attention-and-memory-in-deep-learning-and-nlp/

[7] Attention and Augmented Recurrent Neural Networks , Chris Olah Sept 2016 Link: https://distill.pub/2016/augmented-rnns/

[8] ttention in Deep Networks with Keras, Thushan Ganegedara Link: https://towardsdatascience.com/light-on-math-ml-attention-with-keras-dc8dbc1fad39

[9] MULTI-SCALE CONTEXT AGGREGATION BY DILATED CONVOLUTIONS , Fisher Yu, Vladlen Koltun L;ink https://arxiv.org/pdf/1511.07122.pdf

[10] A guide to convolution arithmetic for deep learning, Vincent Dumoulin and Francesco Visin Link: https://arxiv.org/pdf/1603.07285.pdf

[11] CS231n Convolutional Neural Networks for Visual Recognition Link: http://cs231n.github.io/convolutional-networks/#conv

[12] Understanding separable convolutions Link: https://www.machinecurve.com/index.php/2019/09/23/understanding-separable-convolutions/#depthwise-separable-convolutions

[13] Convolutional Neural Networks, Dive into Deep Learning, Link: https://d2l.ai/chapter_convolutional-neural-networks/index.html

[14] Deep-dive into Convolutional Networks, Antonio Ingargiola, 2019 Link: https://towardsdatascience.com/deep-dive-into-convolutional-networks-48db75969fdf

[15] Attention-based Sequence-to-Sequence in Keras, Wanasit Tanakitrungruang 2017, Link: https://wanasit.github.io/attention-based-sequence-to-sequence-in-keras.html

[16] COUNTERPOINT BY CONVOLUTION Cheng-Zhi Anna Huang, Tim Cooijmans, Adam Roberts, Aaron Courville, Douglas Eck Link: https://arxiv.org/pdf/1903.07227.pdf

[17] Understanding Categorical Cross-Entropy Loss, Raul Gomez 2017, Link: https://gombru.github.io/2018/05/23/cross_entropy_loss/

[18] Time series forecasting with deep stacked unidirectional and bidirecctional LSTMs, Manohar Battula, Link: https://towardsdatascience.com/time-series-forecasting-with-deep-stacked-unidirectional-and-bidirectional-lstms-de7c099bd918

[19] NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE Dzmitry Bahdanau, KyungHyun Cho, Yoshua Bengio Link: https://arxiv.org/pdf/1409.0473.pdf

[20] On the Properties of Neural Machine Translation: Encoder–Decoder Approaches, Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau Link: https://arxiv.org/pdf/1409.1259.pdf

[21] Sequence to Sequence Learning with Neural Networks, Ilya Sutskever, Oriol Vinyals, Quoc V. Le Link: https://arxiv.org/pdf/1409.3215.pdf

[22] MuseGAN: Multi-Track Sequential Generative Adversarial Networks for Symbolic Music Generation and Accompaniment, Hao-Wen Dong, Wen-Yi Hsiao, Li-Chia Yang, Yi-Hsuan Yang Link: https://salu133445.github.io/musegan/pdf/musegan-aaai2018-paper.pdf

[23] Bayesian Inference: Gibbs Sampling, Ilker Yildirim Link: http://www.mit.edu/~ilkery/papers/GibbsSampling.pdf

[24] A Study of the Recurrent Neural Network Encoder-Decoder for Large Vocabulary Speech Recognition, Liang Lu, Xingxing Zhang, Kyunghyun Cho, and Steve Renals, Link: https://homepages.inf.ed.ac.uk/srenals/ll-rnn-is15.pdf

[25] Attention Mechanism, Gabriel Loye Link: https://blog.floydhub.com/attention-mechanism/

[26] How to Configure an Encoder-Decoder Model for Neural Machine Translation Jason Brownle Link: https://machinelearningmastery.com/configure-encoder-decoder-model-neural-machine-translation/

[27] Practical PyTorch: Translation with a Sequence to Sequence Network and Attention Link: https://github.com/spro/practical-pytorch/blob/master/seq2seq-translation/seq2seq-translation.ipynb

[28]Accuracy, Precision, Recall or F1? Koo Ping Shung Link: https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9

[29] RECURRENT NEURAL NETWORKS Design and Applications L.R. Medsker L.C. Jain Link: https://doc.lagout.org/science/Artificial%20Intelligence/Neural%20networks/Recurrent%20Neural%20Networks%20Design%20And%20Applications%20-%20L.R.%20Medsker.pdf