National Technical University of Athens
School of Electrical and Computer Engineering
Division of Signals, Control and Robotics

Automatic Cover Song Generation from Audio Signal

Diploma Thesis

of

GERASIMOS MARKANTONATOS

Supervisor: Alexandros Potamianos
Associate Professor, ECE NTUA

Athens, July 2025

National Technical University of Athens
School of Electrical and Computer Engineering
Division of Signals, Control and Robotics

Automatic Cover Song Generation from Audio Signal

Diploma Thesis

of

GERASIMOS MARKANTONATOS

Supervisor: Alexandros Potamianos
Associate Professor, ECE NTUA

Approved by the examination committee on 1st July 2025.

(Signature)	*(Signature)*	*(Signature)*
.....	.....	.....
Alexandros Potamianos	Costas Tzafestas	Athanasios Rontogiannis
Associate Professor, ECE NTUA	Associate Professor, ECE NTUA	Associate Professor, ECE NTUA

Athens, July 2025

National Technical University of Athens
School of Electrical and Computer Engineering
Division of Signals, Control and Robotics

DISCLAIMER ON ACADEMIC ETHICS AND INTELLECTUAL PROPERTY RIGHTS

Being fully aware of the implications of copyright laws, I expressly state that this diploma thesis, as well as the electronic files and source codes developed or modified in the course of this thesis, are solely the product of my personal work and do not infringe any rights of intellectual property, personality and personal data of third parties, do not contain work / contributions of third parties for which the permission of the authors / beneficiaries is required and are not a product of partial or complete plagiarism, while the sources used are limited to the bibliographic references only and meet the rules of scientific citing. The points where I have used ideas, text, files and / or sources of other authors are clearly mentioned in the text with the appropriate citation and the relevant complete reference is included in the bibliographic references section. I fully, individually and personally undertake all legal and administrative consequences that may arise in the event that it is proven, in the course of time, that this thesis or part of it does not belong to me because it is a product of plagiarism.

(Signature)

.....

Gerasimos Markantonatos

Electrical & Computer Engineering Graduate, NTUA

1st July 2025

Περίληψη

Η δημιουργία διασκευών τραγουδιών αποτελεί μια απαιτητική πρόκληση στον τομέα της Ανάκτησης Μουσικής Πληροφορίας. Η παρούσα εργασία πραγματεύεται βασικούς περιορισμούς του πεδίου: την έλλειψη δεδομένων για μη-ποπ μουσικά είδη και την απουσία μοντέλων δημιουργίας διασκευών για όργανα πέραν του πιάνου. Παρουσιάζουμε δύο συνεισφορές σε επίπεδο συνόλων δεδομένων: το σύνολο GreekSong2Piano, το οποίο περιλαμβάνει 659 ελληνικά τραγούδια με αντίστοιχες διασκευές για πιάνο, και το σύνολο Pop2Guitar, που περιλαμβάνει 40 ζεύγη τραγουδιού-κιθάρας για προσαρμογή μεταξύ οργάνων. Τα σύνολα αυτά επιτρέπουν τη συστηματική μελέτη προσεγγίσεων μεταφοράς μάθησης σε περιβάλλοντα περιορισμένων πόρων. Η μεθοδολογία μας βασίζεται σε αρχιτεκτονική Transformer τύπου encoder-decoder με πυρήνα το μοντέλο T5, αντιμετωπίζοντας τη δημιουργία διασκευών ως πρόβλημα μετάφρασης ακολουθιών: από φασματογραφήματα ήχου σε συμβολικές αναπαραστάσεις MIDI. Συγκρίνουμε συστηματικά τρεις στρατηγικές εκπαίδευσης: εκπαίδευση από την αρχή, μερική προσαρμογή (fine-tuning) και πλήρη προσαρμογή. Επιπλέον, εισάγουμε μια νέα προσέγγιση διαδοχικής προσαρμογής τομέα: από διασκευές πιάνου δυτικής ποπ μουσικής σε ελληνικές διασκευές για πιάνο, και εν συνεχεία σε διασκευές για κιθάρα. Τα πειραματικά αποτελέσματα αναδεικνύουν πλεονεκτήματα των προσεγγίσεων μεταφοράς μάθησης έναντι της εκπαίδευσης από την αρχή. Για τις ελληνικές διασκευές πιάνου, οι στρατηγικές προσαρμογής επιτυγχάνουν έως και 21,0% βελτίωση στην ακρίβεια Melody Chroma σε σχέση με τα βασικά μοντέλα. Η διαδοχική προσαρμογή παρουσιάζει ιδιαίτερη δυναμική για τη δημιουργία διασκευών για κιθάρα, με το μερικώς προσαρμοσμένο μοντέλο να σημειώνει την υψηλότερη βαθμολογία ομοιότητας (3,31$\pm$0,33) σε μελέτες χρηστών, προσεγγίζοντας την ανθρώπινη απόδοση (4,17$\pm$0,28). Το συνολικό πλαίσιο αξιολόγησης συνδυάζει αντικειμενικούς δείκτες (ομοιότητα μελωδίας, αναγνώριση διασκευών, μετρικές βασισμένες σε embeddings) με υποκειμενική αξιολόγηση από χρήστες, αναδεικνύοντας στενή συσχέτιση μεταξύ υπολογιστικών μετρήσεων και ανθρώπινης αντίληψης. Η διπλωματική εργασία εισάγει μια νέα προσέγγιση στη μουσική ενορχήστρωση, με επίκεντρο την πολιτισμική ευαισθησία και την οργανολογική ποικιλομορφία. Συνεισφέρει τόσο στη δημιουργική αξιοποίηση της τεχνητής νοημοσύνης όσο και στην εμβάθυνση της υπολογιστικής κατανόησης της μουσικής μετάφρασης.

Λέξεις Κλειδιά

Δημιουργία Διασκευών Τραγουδιών, Βαθιά Μάθηση, Αρχιτεκτονική Μετασχηματιστή, Μεταφορά Μάθησης, Προσαρμογή στο Πεδίο, Ανάκτηση Μουσικής Πληροφορίας

Abstract

Cover song generation represents a challenging task in Music Information Retrieval, requiring systems to preserve the musical essence of original compositions while adapting them to specific instruments and styles. This thesis addresses key limitations in the field: the scarcity of training data for non-pop musical genres and the lack of cover generation models for instruments beyond piano.

We present two key dataset contributions: the GreekSong2Piano dataset, containing 659 Greek songs paired with piano covers across eight distinct genres (Rembetiko, Laiko, Entexno, etc.), and the Pop2Guitar dataset with 40 song-guitar pairs for cross-instrument domain adaptation. These datasets enable systematic investigation of transfer learning approaches in low-resource scenarios.

Our methodology employs a T5-based encoder-decoder Transformer architecture that treats cover generation as a sequence-to-sequence translation problem, converting audio spectrograms to symbolic MIDI representations. We systematically compare three training strategies: from-scratch training, partial fine-tuning, and full fine-tuning. Additionally, we introduce a novel sequential fine-tuning approach that performs multi-step domain adaptation from Western pop piano covers to Greek piano covers to guitar covers.

Experimental results demonstrate clear advantages for transfer learning approaches over from-scratch training. For Greek piano covers, fine-tuning strategies achieve up to 21.0% improvement in Melody Chroma Accuracy compared to baseline models. The sequential fine-tuning approach shows particular promise for guitar generation, with the partial fine-tuned model achieving the highest similarity ratings (3.31 ± 0.33) in user studies, approaching human performance (4.17 ± 0.28).

Our comprehensive evaluation framework combines objective metrics (melody similarity, cover song identification, embedding-based measures) with subjective user assessment, demonstrating strong correlation between computational measures and human perception. This work establishes a foundation for culturally-aware and instrument-diverse music arrangement systems, contributing to both creative applications and computational understanding of musical translation across cultural and instrumental boundaries.

Keywords

Cover Song Generation, Deep Learning, Transformer Architecture, Transfer Learning, Domain Adaptation, Music Information Retrieval

Dedicated to the memory of my grandfather Manolis

Acknowledgements

Αρχικά θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου Αλέξανδρο Ποταμιάνο για τις κατευθύνσεις και την καθοδήγηση που παρείχε στην διαδικασία εκπόνησης της διπλωματικής μου εργασίας. Μέσα από τις διαλέξεις του αγάπησα την Αναγνώριση Προτύπων και θέλησα να ασχοληθώ σε βάθος με το πεδίο.

Στη συνέχεια θα ήθελα να ευχαριστήσω βαθιά τον υποψήφιο διδάκτορα Χαρίλαο Παπαϊωάννου για τις αμέτρητες συμβουλές και γνώσεις και τον χρόνο που αφιέρωσε σε συζητήσεις, σχόλια και παρατηρήσεις.

Τέλος, θα ήθελα να ευχαριστήσω την οικογένεια μου, τους φίλους μου και όλους αυτούς που ήταν εκεί να με στηρίξουν κατά τη διάρκεια τόσο της διπλωματικής αλλά και όλης της φοιτητικής μου πορείας.

Athens, July 2025

Gerasimos Markantonatos

Contents

List of Figures

List of Tables

Κεφάλαιο 0

Εκτεταμένη Ελληνική Περίληψη

0.1 Εισαγωγή

0.1.1 Κίνητρο

Οι μουσικές διασκευές αποτελούν μία από τις πιο μακροχρόνιες μορφές καλλιτεχνικής έκφρασης, ξεπερνώντας πολιτισμικά όρια και ιστορικές περιόδους. Από τις αρχαίες λαϊκές παραδόσεις μέχρι τη σύγχρονη εποχή, η πρακτική της μουσικής επανερμηνείας παραμένει κεντρικό στοιχείο της μουσικής κουλτούρας. Ωστόσο, η δημιουργία ποιοτικών διασκευών απαιτεί παραδοσιακά σημαντική μουσική τεχνογνωσία, γνώσεις ειδικές για κάθε όργανο και σημαντική επένδυση χρόνου. Οι μουσικοί πρέπει να αναλύσουν την αρχική σύνθεση, να κατανοήσουν την αρμονική της δομή και να αναπτύξουν μια διασκευή που διατηρεί την ουσία του τραγουδιού ενώ παράλληλα αναδεικνύει την καλλιτεχνική τους προοπτική.

Η πρόσφατη επανάσταση στην τεχνητή νοημοσύνη και τη βαθιά μάθηση έχει ανοίξει ευκαιρίες για την αυτοματοποιημένη παραγωγή μουσικού περιεχομένου. Τα μοντέλα έχουν επιδείξει αξιοσημείωτες ικανότητες στη δημιουργία μουσικής υψηλής ποιότητας, οδηγώντας φυσικά στη δυνατότητα παραγωγής διασκευών. Παρά αυτές τις πολλά υποσχόμενες εξελίξεις, διάφορες προκλήσεις εμποδίζουν την πρόοδο: το πιο θεμελιώδες εμπόδιο είναι η έλλειψη δεδομένων, καθώς τα συγχρονισμένα ζεύγη τραγουδιού-διασκευής είναι εξαιρετικά περιορισμένα, ιδιαίτερα για όργανα πέρα από το πιάνο και για μουσικά είδη εκτός του δυτικού ποπ. Η παρούσα διπλωματική εργασία στοχεύει να αντιμετωπίσει αυτές τις προκλήσεις αναπτύσσοντας εξειδικευμένα σύνολα δεδομένων, αξιοποιώντας τεχνικές μεταφοράς μάθησης και θέτοντας ένα αντικειμενικό πλαίσιο αξιολόγησης των παραγόμενων μουσικών κομματιών

0.1.2 Συνεισφορά

Η παρούσα διπλωματική εργασία συνεισφέρει στον τομέα της Ανάκτησης Μουσικής Πληροφορίας αναφορικά με τη δημιουργία διασκευών τραγουδιών. Δεδομένου ότι το αντικείμενο της δημιουργίας διασκευών έχει περιοριστεί σε διασκευές πιάνου δυτικής ποπ μουσικής, και ότι μουσικοί τομείς με περιορισμένους πόρους όπως η ελληνική μουσική και όργανα όπως η κιθάρα δεν έχουν μελετηθεί εκτενώς, η εργασία αυτή εξερευνά προσεγγίσεις μεταφοράς μάθησης και τεχνικές προσαρμογής πεδίου για τη δημιουργία διασκευών σε διαφορετικά μουσικά στυλ και όργανα. Οι βασικές συνεισφορές της παρούσας εργασίας είναι:

- **Σύνολο δεδομένων GreekSong2Piano:** Ένα νέο σύνολο δεδομένων που περιλαμ-

βάνει 659 ελληνικά τραγούδια με τις αντίστοιχες διασκευές τους για πιάνο σε μορφές ήχου και MIDI, χωρισμένες ανά είδος και με τους στίχους τους.

- **Σύνολο δεδομένων Pop2Guitar:** Ένα σύνολο δεδομένων 40 ποπ τραγουδιών και των αντίστοιχων διασκευών τους για κιθάρα, ενορχηστρωμένων από διαφορετικούς μουσικούς, που επιτρέπει την επέκταση πέρα από τις προσεγγίσεις που επικεντρώνονται στο πιάνο.

- **Ανάλυση στρατηγικών εκπαίδευσης για δημιουργία διασκευών σε συνθήκες περιορισμένων πόρων:** Σύγκριση της εκπαίδευσης από το μηδέν, της μερικής προσαρμογής και της πλήρους προσαρμογής, αναδεικνύοντας την αποτελεσματικότητα της μεταφοράς μάθησης σε σενάρια περιορισμένων πόρων και την προσαρμογή σε διαφορετικά στυλ και όργανα.

- **Μεθοδολογία αντικειμενικής αξιολόγησης:** Ένα πρωτόκολλο αξιολόγησης που αξιοποιεί προεκπαιδευμένα μοντέλα για *ταυτοποίηση διασκευών* και *κατανόησης μουσικής*, για να αξιολογηθούν αντικειμενικά οι παραγόμενες διασκευές.

0.2 Μηχανική Μάθηση

Η Μηχανική Μάθηση αποτελεί έναν υποτομέα της τεχνητής νοημοσύνης που επικεντρώνεται στην ανάπτυξη αλγορίθμων και συστημάτων που μπορούν να μαθαίνουν και να λαμβάνουν αποφάσεις από δεδομένα χωρίς να προγραμματίζονται ρητά για κάθε συγκεκριμένη εργασία [18]. Αντί να ακολουθούν προ-γραμμένες οδηγίες, τα συστήματα μηχανικής μάθησης αναγνωρίζουν μοτίβα στα δεδομένα και χρησιμοποιούν αυτά τα μοτίβα για να κάνουν προβλέψεις ή αποφάσεις σχετικά με νέες, άγνωστες πληροφορίες. Οι προσεγγίσεις μηχανικής μάθησης κατηγοριοποιούνται ευρέως σε τρία κύρια παραδείγματα: την επιβλεπόμενη μάθηση [19] που προπονεί αλγορίθμους σε επισημασμένα σύνολα δεδομένων, τη μη-επιβλεπόμενη μάθηση [20] που εργάζεται με δεδομένα χωρίς προκαθορισμένες ετικέτες, και την ενισχυτική μάθηση [21] όπου οι πράκτορες μαθαίνουν μέσω αλληλεπίδρασης με το περιβάλλον.

Η βαθιά μάθηση αντιπροσωπεύει μια σημαντική εξέλιξη από τις παραδοσιακές προσεγγίσεις μηχανικής μάθησης, χαρακτηριζόμενη από τη χρήση τεχνητών νευρωνικών δικτύων με πολλαπλά επίπεδα που μπορούν αυτόματα να μαθαίνουν ιεραρχικές αναπαραστάσεις από δεδομένα [2]. Βασικές αρχιτεκτονικές όπως τα Πολυεπίπεδα Πέρσεπτρονς (MLPs) [2], τα Συνελικτικά Νευρωνικά Δίκτυα (CNNs) [22] για δεδομένα με δομή πλέγματος, και τα Επαναληπτικά Νευρωνικά Δίκτυα (RNNs) [23] για ακολουθιακά δεδομένα, έχουν οδηγήσει στην επιτυχία της βαθιάς μάθησης σε διαφορετικούς τομείς. Η εισαγωγή των μηχανισμών προσοχής και της αρχιτεκτονικής Transformer [4] αντιπροσώπευσε μια θεμελιώδη ανακάλυψη στη μοντελοποίηση ακολουθιών.

Στον τομέα της μουσικής, οι πληροφορίες μπορούν να αναπαρασταθούν με δύο κύριους τρόπους: συμβολικές και ηχητικές αναπαραστάσεις. Οι συμβολικές αναπαραστάσεις κωδικοποιούν τη μουσική ως διακριτά σύμβολα παρά ως συνεχή σήματα ήχου, με το MIDI να αναδεικνύεται ως η κυρίαρχη συμβολική αναπαράσταση στις υπολογιστικές εφαρμογές

μουσικής. Οι σύγχρονες νευρωνικές προσεγγίσεις έχουν αναπτύξει εξειδικευμένα λεξικά tokens που επεκτείνονται πέρα από τα παραδοσιακά MIDI events. Από την άλλη πλευρά, οι ηχητικές αναπαραστάσεις αποτυπώνουν τη συνεχή μορφή κύματος ή τα φασματικά χαρακτηριστικά ενός μουσικού σήματος, περιλαμβάνοντας αναπαραστάσεις όπως τα φάσματα συχνότητας, τα mel-spectrograms, και τις μαθημένες ηχητικές ενσωματώσεις μέσω βαθιών νευρωνικών δικτύων.

Η δημιουργία μουσικής στοχεύει στη δημιουργία νέου μουσικού περιεχομένου μέσω υπολογιστικών μοντέλων. Η συμβολική δημιουργία επικεντρώνεται στην παραγωγή δομημένων αναπαραστάσεων όπως MIDI ή παρτιτούρες, ενώ η άμεση ηχητική δημιουργία συνθέτει ακατέργαστες μορφές κύματος ή φάσματα, στοχεύοντας να αποτυπώσει τον πλούτο και την εκφραστικότητα της μουσικής ερμηνείας. Μοντέλα όπως το MuseNet [24], το WaveNet [25] και το Jukebox [26] έχουν αποδείξει ότι είναι δυνατή η δημιουργία μουσικής υψηλής ποιότητας που ικανοποιεί διαφορετικά στυλιστικά και δομικά κριτήρια.

0.3 Δημιουργία Διασκευών Μουσικού Κομματιού

Σε αυτό το κεφάλαιο εξετάζουμε το σώμα εργασιών που σχετίζεται με τη δημιουργία διασκευών. Αρχίζουμε εξερευνώντας την Αυτόματη Μουσική Μεταγραφή, συμπεριλαμβανομένων των προσεγγίσεων βαθιάς μάθησης. Στη συνέχεια, μιλάμε για τον Μουσικό Μετασχηματισμό, επικεντρώνοντας σε μεθόδους που προσαρμόζουν και τροποποιούν το μουσικό περιεχόμενο. Έπειτα εστιάζουμε στην Αναγνώριση Διασκευών Τραγουδιών, η οποία περιγράφει πώς μπορούν να αναγνωριστούν οι διασκευές και μας παρέχει βαθιά κατανόηση της φύσης τους. Τέλος,μελετάμε τον τομέα της δημιουργίας διασκευών, αναδεικνύοντας τόσο τις παραδοσιακές τεχνικές όσο και τις πρόσφατες εξελίξεις που ενημερώνουν την προσέγγισή μας.

0.3.1 Αυτόματη Μεταγραφή Μουσικής

Η Αυτόματη Μεταγραφή Μουσικής είναι η διαδικασία μετατροπής ενός ακουστικού μουσικού σήματος σε συμβολική αναπαράσταση που περιλαμβάνει τόνο, χρόνο έναρξης, διάρκεια και τύπο οργάνου [6]. Στο πλαίσιο της δημιουργίας διασκευών, η Αυτόματη Μεταγραφή Μουσικής μπορεί να χρησιμοποιηθεί για να βοηθήσει στην δημιουργία συνόλων δεδομένων εκπαίδευσης μεταγράφοντας μουσικά κομμάτια.

Η βαθιά μάθηση έχει επηρεάσει σημαντικά τη μουσική μεταγραφή. Κυριότερες εξελίξεις περιλαμβάνουν το σύστημα Sonic του Marolt [27] με δίκτυα χρονικής καθυστέρησης, το μοντέλο Onsets and Frames [7] που χρησιμοποιεί συνελικτικά και επαναλαμβανόμενα νευρωνικά δίκτυα, και το σύστημα υψηλής ανάλυσης του Kong [8] για μεταγραφή πιάνου. Πιο πρόσφατα, ο Hawthorne [7] έδειξε ότι γενικοί κωδικοποιητές-αποκωδικοποιητές Transformer μπορούν να επιτύχουν αντίστοιχη απόδοση, ενώ το μοντέλο MT3 [10] επέκτεινε τη μεταγραφή σε πολλαπλά όργανα ταυτόχρονα.

Τα κυριότερα σύνολα δεδομένων περιλαμβάνουν: το **MAPS** για μεταγραφή πιάνου με μεμονωμένες νότες και πλήρη κομμάτια, το **SLAKH2100** με 2100 μικτά κομμάτια πολλαπλών οργάνων, το **GUITARSET** με 360 αποσπάσματα κιθάρας, το **MUSICNET** με 330 κλασικές

εγγραφές, και το **MAESTROV3** με 200 ώρες παραστάσεων πιάνου υψηλής ποιότητας.

0.3.2 Μετασχηματισμός Μουσικής

Ο μετασχηματισμός μουσικής αποτελεί μια διαδικασία που αντιστοιχεί ένα μουσικό απόσπασμα σε ένα άλλο, διατηρώντας ορισμένα μουσικά χαρακτηριστικά ενώ τροποποιεί άλλα, διατηρώντας παράλληλα τη μουσική συνοχή [13]. Ένα μουσικό απόσπασμα ορίζεται ως ένας συνδυασμός μουσικών μέτρων για κάποια φωνή, μαζί με το αντίστοιχο αρμονικό περιβάλλον. Τα μουσικά χαρακτηριστικά περιλαμβάνουν στοιχεία όπως οι νότες της μελωδίας, οι αρμονίες, το tempo και η τονικότητα.

Η **μεταφορά μουσικού στυλ** επικεντρώνεται στην αλλαγή του στυλ ενός μουσικού αποσπάσματος τροποποιώντας στοιχεία όπως ο τόνος, το timbre και η αρμονία, διατηρώντας τη βασική μελωδία και τον ρυθμό. Σύμφωνα με τον ορισμό του [1], το στυλ αναφέρεται στον μοναδικό τρόπο με τον οποίο κάθε διασκευαστής ερμηνεύει και συνθέτει κατά τη δημιουργία μιας διασκευής. Το σύστημα Groove2Groove [14] παρουσιάζει μια προσέγγιση one-shot style transfer για συμβολική μουσική χρησιμοποιώντας εποπτευόμενα συνθετικά δεδομένα. Το μοντέλο ακολουθεί το πρότυπο encoder-decoder, χρησιμοποιώντας δύο κωδικοποιητές: έναν για το μουσικό περιεχόμενο και έναν για το στυλ και έναν αποκωδικοποιητή που παράγει την έξοδο. Άλλες προσεγγίσεις περιλαμβάνουν την κωδικοποίηση μουσικού στυλ με transformer autoencoders [28], όπου οι συγγραφείς εισάγουν ένα μοντέλο που καταγράφει αναπαραστάσεις στυλ υψηλού επιπέδου χρησιμοποιώντας αυτοκωδικοποιητές βασισμένους σε Transformer. Τέλος το MuseNet [29] αποτελεί μια σημαντική εξέλιξη, διότι είναι ικανό να δημιουργεί τετράλεπτες συμβολικές μουσικές συνθέσεις με διαφορετικά όργανα και στυλ.

Η **μείωση μουσικής** αναφέρεται στη διαδικασία απλοποίησης μιας σύνθετης μουσικής σύνθεσης διατηρώντας τα βασικά της στοιχεία. Η μείωση για πιάνο είναι ιδιαίτερα σημαντική, καθώς μετατρέπει πολυφωνικά έργα σε διατάξεις κατάλληλες για εκτέλεση από πιάνο. Οι [15] παρουσιάζουν μια μέθοδο για την αυτόματη απλοποίηση σύνθετων μουσικών συνθέσεων για πιάνο, εισάγοντας έναν αλγόριθμο επιλογής φράσεων που αξιολογεί τη σημασία διαφορετικών τμημάτων της σύνθεσης. Άλλες μελέτες [30] αναπτύσσουν διαδραστικά συστήματα διάταξης πιάνου που παρέχουν ανατροφοδότηση σε πραγματικό χρόνο. Πιο πρόσφατα, οι [31] χρησιμοποιούν ένα μοντέλο εποπτευόμενης μάθησης βασισμένο σε CNN για τη δημιουργία παρτιτούρων κατάλληλων για πιάνο από τραγούδια που αποτελούνται από πολλαπλά μέρη, δείχνοντας ότι οι τεχνικές βαθιάς μάθησης μπορούν να εφαρμοστούν αποτελεσματικά στην εργασία μείωσης πιάνου. Η σύνδεση μεταξύ μείωσης πιάνου και δημιουργίας διασκευών έγκειται στην ικανότητα διατήρησης μελωδικών και αρμονικών δομών ενώ τροποποιούν την οργανική διάταξη για να ταιριάζει σε ένα συγκεκριμένο πλαίσιο εκτέλεσης.

0.3.3 Αναγνώριση Διασκευών Τραγουδιών

Η αναγνώριση διασκευών τραγουδιών (Cover Song Identification - CSI) αποτελεί σημαντικό τομέα στην ανάκτηση μουσικής πληροφορίας, με εφαρμογές στη μουσική βιομηχανία και την προστασία πνευματικών δικαιωμάτων.

Μια διασκευή ορίζεται ως μια εναλλακτική απόδοση ηχογραφημένου τραγουδιού που μπορεί να διαφέρει ως προς το ηχόχρωμα, ρυθμό, δομή, κλίμακα, ή γλώσσα [32]. Οι

διασκευές κατηγοριοποιούνται σε τύπους με κύριους τους ορχηστρικό, ακουστικό, ρεμίξ, ζωντανή εκτέλεση, κ.ά. [33, 34, 35].

Οι παραδοσιακές μέθοδοι με χειροκίνητα χαρακτηριστικά [32, 36] αντικαταστάθηκαν από μοντέλα βαθιάς μάθησης βασισμένα σε CNN. Σημαντικά συστήματα περιλαμβάνουν το TPPNet [37], ByteCover [38], PiCKINet [39], και CoverHunter [40], που επιτυγχάνουν state-of-the-art απόδοση μέσω συνδυασμού ταξινόμησης και μετρικής μάθησης.

Τα μοντέλα CSI είναι καλά εξοπλισμένα ώστε να κατανοούν τις διασκευές και τον τρόπο με τον οποίο αυτές συνδέονται με τα αρχικά κομμάτια. Έχουν την ικανότητα να αναγνωρίζουν και να μετρούν την ομοιότητα ανάμεσα σε μια διασκευή και το πρωτότυπό της, γεγονός που αποδεικνύεται ιδιαίτερα χρήσιμο κατά την αξιολόγηση των παραγόμενων διασκευών. Με την βοήθεια ενός μοντέλου CSI, μπορούμε να αποτιμήσουμε την ποιότητα ενός μοντέλου δημιουργίας διασκευών, είτε συγκρίνοντας την ομοιότητα ανάμεσα στο ζεύγος πρωτότυπο-διασκευή είτε υπολογίζοντας την απόσταση μεταξύ του αποτελόντας ένα πολύτιμο εργαλείο αξιολόγησης.

0.3.4 Δημιουργία Διασκευών

Η δημιουργία διασκευών (cover generation) αποτελεί τη διαδικασία δημιουργίας μιας νέας εκδοχής ενός υπάρχοντος τραγουδιού. Η παραγωγή μιας διασκευής απαιτεί συνήθως σημαντικό χρόνο, προσπάθεια και προχωρημένες μουσικές δεξιότητες. Το πεδίο της αυτόματης δημιουργίας διασκευών προσπαθεί να αντιμετωπίσει αυτό το πρόβλημα.

Οι πρώιμες υπολογιστικές προσεγγίσεις περιλάμβαναν συστήματα όπως το Song2Quartet [16], που δημιουργούσε διασκευές string quartet συνδυάζοντας πιθανοτικά μοντέλα με ανάλυση ήχου, και το Song2Guitar [17], που επικεντρωνόταν στη δημιουργία διασκευών κιθάρας.

Ο τομέας της δημιουργίας διασκευών έχει βιώσει μια μετάβαση από παραδοσιακές προσεγγίσεις βασισμένες σε κανόνες σε σύγχρονες μεθοδολογίες βαθιάς μάθησης. Η εισαγωγή του Pop2Piano από τους [1] έδειξε ότι είναι δυνατή η δημιουργία διασκευών πιάνου απευθείας από είσοδο ήχου χρησιμοποιώντας μια βασισμένη σε δεδομένα προσέγγιση, χωρίς να βασίζεται σε ενδιάμεση μεταγραφή ή μουσική ανάλυση. Το Pop2Piano βασίζεται στην αρχιτεκτονική T5 Transformer [5], προσαρμόζοντας το sequence-to-sequence framework που χρησιμοποιείται συνήθως στην επεξεργασία φυσικής γλώσσας για το πεδίο της μουσικής. Το σύστημα αντιμετωπίζει τη δημιουργία διασκευών ως πρόβλημα μετάφρασης, όπου η ακολουθία εισόδου αποτελείται από frames φασματογραφήματος ήχου και η ακολουθία εξόδου περιέχει συμβολικά MIDI events.

Η αξιολόγηση αυτόματα παραγόμενων διασκευών παρουσιάζει προκλήσεις, καθώς απαιτεί την εκτίμηση τόσο της τεχνικής πιστότητας όσο και της μουσικής ποιότητας. Το Pop2Piano [1] έθεσε ένα πλαίσιο αξιολόγησης που συνδυάζει αντικειμενικές μετρήσεις και υποκειμενική εκτίμηση μέσο ηλεκτρονικού ερωτηματολογίου. Για την αντικειμενική αξιολόγηση, οι συγγραφείς χρησιμοποίησαν την Melody Chroma Accuracy (MCA) για να μετρήσουν πόσο καλά οι παραγόμενες διασκευές πιάνου διατήρησαν το μελωδικό περιεχόμενο των αρχικών τραγουδιών.

0.4 Σύνολα Δεδομένων

0.4.1 Υφιστάμενα Σύνολα Δεδομένων

Στον τομέα παραγωγής διασκευών δεν υπάρχουν πολλά σύνολα δεδομένων, κυρίως λόγω του υψηλού κόστους δημιουργίας υψηλής ποιότητας συλλογών. Επίσης, τα υπάρχοντα σύνολα δεδομένων εστιάζουν κυρίως σε διασκευές πιάνου ποπ τραγουδιών.

Pop2Piano Dataset: Το σύνολο δεδομένων Pop2Piano [1] περιλαμβάνει 5.989 διασκευές πιάνου από 21 διασκευαστές μαζί με τα αντίστοιχα pop τραγούδια, συνολικής διάρκειας 307 ωρών. Μετά από συγχρονισμό και φιλτράρισμα, το τελικό σύνολο εκπαίδευσης περιλαμβάνει 4.989 κομμάτια. Το μέγεθός του υπερβαίνει τα 250 GB, με αποτέλεσμα να είναι δύσκολη η αξιοποίηση του.

POP909: Το POP909 [41] αποτελεί σύνολο δεδομένων 909 κινεζικών pop τραγουδιών σχεδιασμένο για έρευνα στο πεδίο της ανάκτησης μουσικής πληροφορίας. Κάθε τραγούδι περιλαμβάνει κομμάτια MIDI για φωνητική μελωδία, κύριο όργανο και συνοδεία πιάνου, ευθυγραμμισμένα με τον πρωτότυπο ήχο. Το μέγεθός του είναι περίπου 34 GB.

Τρία κύρια σύνολα δεδομένων υποστηρίζουν την έρευνα ελληνικής μουσικής. Το Greek Audio Dataset (GAD) [42] περιλαμβάνει μεταδεδομένα και χαρακτηριστικά για 1.000 ελληνικά τραγούδια σε οκτώ κατηγορίες ειδών (Ρεμπέτικο, Λαϊκό, Έντεχνο κ.ά.). Το Greek Music Dataset (GMD) επεκτείνει το GAD σε 1.400 κομμάτια με προ-υπολογισμένα χαρακτηριστικά ήχου, στίχων και συμβολικά χαρακτηριστικά. Το σύνολο δεδομένων Lyra [43] εστιάζει στην παραδοσιακή και λαϊκή μουσική με 1.570 κομμάτια (80 ώρες) από την ντοκιμαντέρ σειρά «Το Αλάτι της Γης», παρέχοντας λεπτομερή μεταδεδομένα όπως είδη και γεωγραφική προέλευση. Τα ελληνικά σύνολα δεδομένων αποτυπώνουν τα μοναδικά χαρακτηριστικά της ελληνικής μουσικής παράδοσης, παρέχοντας σημαντικούς πόρους για έρευνα και διατήρηση πολιτιστικής κληρονομιάς.

Για την αντιμετώπιση της έλλειψης διαθέσιμων συνόλων δεδομένων στον τομέα της δημιουργίας διασκευών, αναπτύξαμε δύο νέα σύνολα δεδομένων που επεκτείνουν την έρευνα πέρα από τις δυτικές pop μελωδίες και τις διασκευές πιάνου.

0.4.2 Σύνολο Δεδομένων GreekSong2Piano

Το πρώτο συγχρονισμένο σύνολο δεδομένων ελληνικών τραγουδιών και διασκευών πιάνου, αποτελούμενο από 659 ζεύγη που καλύπτουν 41 ώρες μουσικής σε 8 διαφορετικά ελληνικά είδη (Ρεμπέτικο, Λαϊκό, Έντεχνο, Μοντέρνο Λαϊκό, Rock, Hip Hop/R&B, Pop, Εναλλακτικό). Οι διασκευές συλλέχθηκαν κυρίως από το κανάλι του Γιάννη Γρηγορίου στο YouTube και μεταγράφηκαν σε MIDI χρησιμοποιώντας το μοντέλο του Kong [8]. Το σύνολο χωρίστηκε με αναλογία 80-10-10 για την εκπαίδευση.

0.4.3 Σύνολο Δεδομένων Pop2Guitar

Επεκτείνοντας πέρα από το πιάνο, δημιουργήσαμε ένα σύνολο 40 ζευγών τραγουδιών-διασκευών κιθάρας (2,52 ώρες) για την εξερεύνηση της προσαρμογής πεδίου μεταξύ οργάνων. Λόγω των περιορισμών στη μεταγραφή κιθάρας με το MT3 [10], συλλέξαμε αρχεία MIDI από

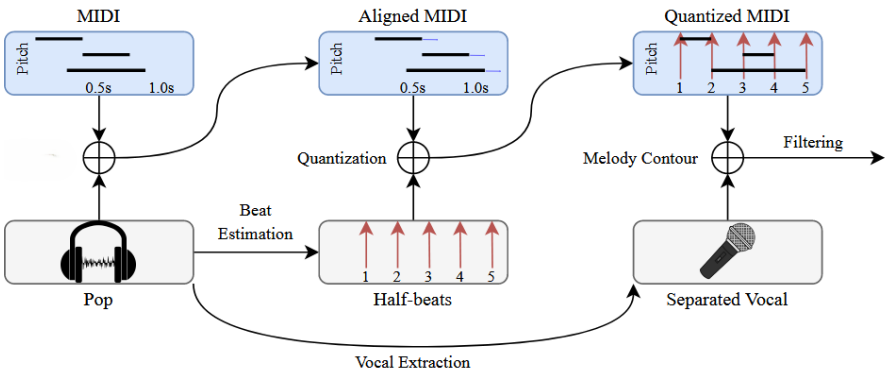
το MuseScore [24]. Λόγω του περιορισμένου μεγέθους του συνόλου δεδομένων Pop2Guitar (40 ζεύγη), εφαρμόσαμε 5-fold cross-validation για την απόκτηση αξιόπιστων εκτιμήσεων απόδοσης. Αυτή η προσέγγιση διασφαλίζει ότι κάθε ζεύγος τραγουδιού-διασκευής χρησιμεύει τόσο ως δεδομένο εκπαίδευσης όσο και ως δεδομένο επαλήθευσης. Έτσι μεγιστοποιούμε τη χρησιμότητα του περιορισμένου συνόλου δεδομένων μας.

Αυτά τα σύνολα δεδομένων αποτελούν τη βάση για την εξερεύνηση στρατηγικών μεταφοράς μάθησης σε σενάρια χαμηλών πόρων, επιτρέποντας την προσαρμογή σε πολιτισμικά πεδία και άλλα όργανα.

0.5 Μεθοδολογία

0.5.1 Προεπεξεργασία

Η διαδικασία προεπεξεργασίας περιλαμβάνει τρία στάδια που εξασφαλίζουν την ποιότητα και τη χρησιμότητα των δεδομένων εκπαίδευσης, ακολουθώντας τη μεθοδολογία της μελέτης [1].



Σχήμα 1. *Προεπεξεργασία. Πηγή: [1]*

Συγχρονισμός: Χρησιμοποιούμε το SynctoolBox [44] για την ακριβή ευθυγράμμιση των πρωτότυπων τραγουδιών με τις αντίστοιχες διασκευές τους. Η διαδικασία ξεκινά με κανονικοποίηση του ήχου και εφαρμογή δυναμικής παραμόρφωσης χρόνου (Dynamic Time Warping - DTW) για να αντιμετωπιστούν διαφορές σε τονικότητα και ρυθμό. Στη συνέχεια, οι χρόνοι των νοτών MIDI προσαρμόζονται μέσω γραμμικής παρεμβολής για να επιτευχθεί πλήρης συγχρονισμός.

Εξαγωγή Ρυθμού και Κβαντοποίηση: Με τη χρήση του Essentia [45], εξάγουμε τους ρυθμούς από τις ηχητικό κομμάτια και κβαντοποιούμε τους χρόνους των νοτών σε μονάδες όγδοου. Αυτή η προσέγγιση μετατρέπει την αναπαράσταση από συνεχή χρόνο σε δομημένη μορφή, μειώνοντας την εντροπία των δεδομένων και διευκολύνοντας την επεξεργασία από το μοντέλο.

Φιλτράρισμα Ποιότητας: Εφαρμόζουμε αυτόματο και χειροκίνητο φιλτράρισμα για την απομάκρυνση ζευγών χαμηλής ποιότητας. Υπολογίζουμε την Melody Chroma Accuracy (MCA) [46] μεταξύ των φωνητικών που εξάγονται με το Spleeter [47] και της κύριας μελωδικής γραμμής της διασκευής, απορρίπτοντας ζεύγη με MCA 0.10 ή λιγότερο. Επιπλέον,

αποκλείουμε ζεύγη με διαφορά μήκους μεγαλύτερη του 40% και επαληθεύουμε χειροκίνητα την ευθυγράμμιση όλων των δειγμάτων.

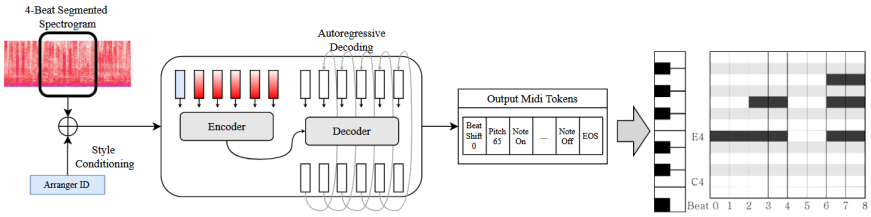
Αυτή η τριφασική διαδικασία εξασφαλίζει ότι το τελικό σύνολο δεδομένων περιέχει υψηλής ποιότητας, συγχρονισμένα ζεύγη τραγουδιού-διασκευής που είναι κατάλληλα για την εκπαίδευση μοντέλων αυτόματης δημιουργίας διασκευών.

0.5.2 Μοντέλο

Η δημιουργία μουσικών διασκευών αντιμετωπίζεται ως πρόβλημα sequence-to-sequence, όπου καρέ ήχου μετατρέπονται σε συμβολικά tokens που αντιπροσωπεύουν νότες του οργάνου διασκευής. Χρησιμοποιούμε μια γενική αρχιτεκτονική encoder-decoder Transformer όπου κάθε θέση εισόδου περιέχει ένα καρέ spectrogram και κάθε θέση εξόδου ένα γεγονός από λεξιλόγιο τύπου MIDI.

Το μοντέλο βασίζεται στην αρχιτεκτονική T5 [5], ακολουθώντας το Pop2Piano [1]. Χρησιμοποιεί τυπικούς Transformer blocks με σχετικές θεσιακές embeddings και αυτοπαλίνδρομη αποκωδικοποίηση. Υιοθετούμε το μοντέλο T5 "small" με 60 εκατομμύρια παραμέτρους.

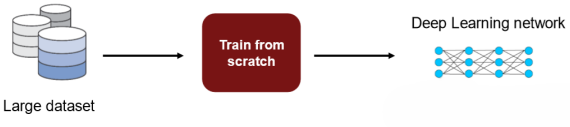
Ως είσοδος χρησιμοποιούνται log Mel spectrograms μαζί με token διασκευαστή. Η έξοδος είναι κατανομή softmax πάνω σε λεξιλόγιο που περιλαμβάνει: Note Pitch (128 τιμές, μόνο 88 για πιάνο), Note On/Off (2 τιμές), Beat Shift (100 τιμές για χρονική κβάντιση), και EOS/PAD tokens. Το λεξιλόγιο εμπνέεται από την προδιαγραφή MIDI [48] και εφαρμογές σε AMT [9, 10].



Σχήμα 2. *Αρχιτεκτονική του Μοντέλου. Πηγή: [1]*

0.5.3 Στρατηγικές Εκπαίδευσης Μοντέλων για Διασκευές Ελληνικών Τραγουδιών σε Πιάνο

Σε αυτήν την ενότητα, παρουσιάζουμε τρεις στρατηγικές εκπαίδευσης: εκπαίδευση από την αρχή (training from scratch), (partial fine-tuning), και (full fine-tuning). Το σύνολο δεδομένων GreekSong2Piano χρησιμοποιείται για όλες τις προσεγγίσεις εκπαίδευσης.

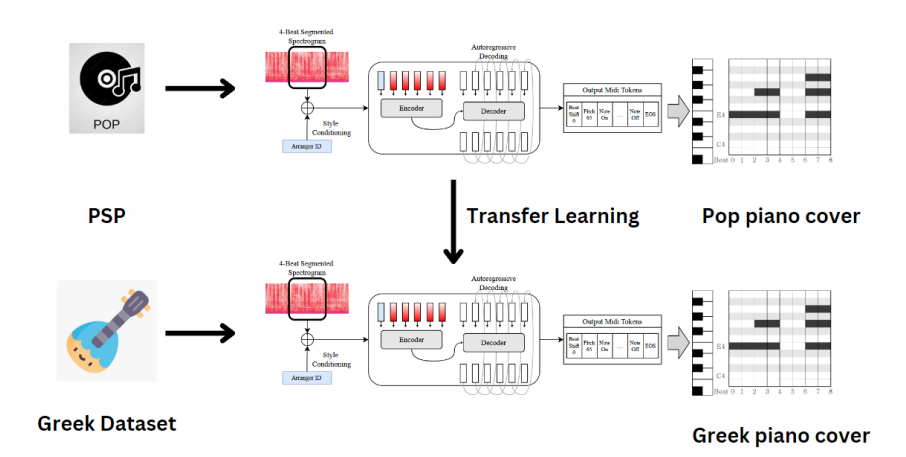


Σχήμα 3. *Εκπαίδευση από την αρχή*

Εκπαίδευση από την Αρχή: Για την ανάπτυξη ενός μοντέλου δημιουργίας διασκευών

πιάνου για ελληνικά τραγούδια, δοκιμάσαμε πρώτα να εκπαιδεύσουμε το μοντέλο από την αρχή. Χρησιμοποιούμε την ίδια αρχιτεκτονική μοντέλου και tokenizer όπως το [1]. Το πρόβλημα με αυτήν την προσέγγιση είναι ότι απαιτεί ένα μεγάλο σύνολο δεδομένων για να επιτύχει καλά αποτελέσματα και το δικό μας σύνολο δεδομένων δεν ήταν αρκετά μεγάλο. Το σύνολο δεδομένων μας έχει λιγότερα από 1000 τραγούδια και είναι περισσότερο από 5 φορές μικρότερο από το σύνολο δεδομένων που χρησιμοποιήθηκε για την εκπαίδευση του μοντέλου Pop2Piano.

Μεταφορά Μάθησης: Για να αντιμετωπίσουμε το πρόβλημα των περιορισμένων πόρων, εφαρμόζουμε Transfer Learning. Ξεκινάμε από το μοντέλο Pop2Piano [1] που είναι εκπαιδευμένο σε περίπου 5000 ζεύγη τραγουδιών-πιάνου και κάνουμε fine-tuning των παραμέτρων του στο συγκεκριμένο σύνολο δεδομένων μας.



Σχήμα 4. *Μεταφορά Μάθησης*

Δοκιμάσαμε δύο προσεγγίσεις fine-tuning:

Μερική: Πάγωμα των encoder στρωμάτων και εκπαίδευση μόνο των τελικών decoder στρωμάτων και της κεφαλής μοντελοποίησης γλώσσας, αξιοποιώντας το γεγονός ότι τα πρώιμα στρώματα μαθαίνουν γενικές μουσικές αναπαραστάσεις.

Πλήρης: Εκπαίδευση όλων των στρωμάτων του μοντέλου T5-small (6 encoder και 6 decoder στρώματα), επιτρέποντας πλήρη προσαρμογή στην εργασία-στόχο παρά την αυξημένη υπολογιστική απαίτηση [49].

Η συνδυασμένη προσέγγιση μεταφοράς μάθησης και πλήρους λεπτομερούς ρύθμισης αποδείχθηκε η πιο αποτελεσματική για την αντιμετώπιση των προκλήσεων των περιορισμένων δεδομένων και των ιδιαιτεροτήτων των ελληνικών μουσικών ειδών.

0.5.4 Στρατηγικές Εκπαίδευσης Μοντέλων για Διασκευές Τραγουδιών σε Κιθάρα

Για την ανάπτυξη μοντέλων δημιουργίας διασκευών κιθάρας αντιμετωπίζουμε τις προκλήσεις των περιορισμένων δεδομένων εκπαίδευσης μέσω στρατηγικών transfer learning και domain adaptation. Ενώ η δημιουργία διασκευών πιάνου για pop μουσική έχει επιτύχει αξιόλογα αποτελέσματα [1], συγκρίσιμα σύνολα δεδομένων για άλλα όργανα είναι σπάνια.

Τροποποίηση Λεξιλογίου: Προσαρμόζουμε τον tokenizer για έξοδο MIDI κιθάρας, καλύπτοντας το εύρος από E2 (νότα MIDI 40) έως E6 (νότα MIDI 88) και αλλάζοντας το όργανο απόδοσης από πιάνο σε ακουστική κιθάρα [48].

Εκπαίδευση από το Αρχή: Εκπαιδεύσαμε ένα μοντέλο από το μηδέν χρησιμοποιώντας το σύνολο δεδομένων Pop2Guitar των 40 ζευγών για να εγκαθιδρύσουμε μια βάση αναφοράς, παρόλο που το περιορισμένο μέγεθος του συνόλου δεδομένων οδήγησε σε υπερπροσαρμογή.

Μεταφορά Μάθησης: Εφαρμόζουμε transfer learning από το προεκπαιδευμένο μοντέλο Pop2Piano, διερευνώντας δύο προσεγγίσεις:

Μερική: Παγώνουμε τα στρώματα του encoder και τα πρώτα στρώματα του decoder, εκπαιδεύοντας μόνο τα τελευταία στρώματα και την κεφαλή γλωσσικής μοντελοποίησης. Αυτή η προσέγγιση διατηρεί τις γενικές μουσικές αναπαραστάσεις ενώ προσαρμόζεται στις ιδιαιτερότητες της κιθάρας.

Πλήρης: Ενημερώνουμε όλες τις παραμέτρους του μοντέλου, επιτρέποντας πλήρη προσαρμογή. Ενώ απαιτεί περισσότερους υπολογιστικούς πόρους, μπορεί να οδηγήσει σε καλύτερη απόδοση όταν οι τομείς πηγής και στόχου διαφέρουν σημαντικά.

0.5.5 Διαδοχική Εκπαίδευση για Δημιουργία Διασκευών Ελληνικών Τραγουδιών σε Κιθάρα

Για την αντιμετώπιση της πρόκλησης δημιουργίας διασκευών κιθάρας ειδικά για ελληνικά τραγούδια, προτείνουμε μια στρατηγική διαδοχικής εκπαίδευσης που αξιοποιεί τη διαδικασία μεταφοράς μάθησης.

Η διαδοχική μας προσέγγιση ακολουθεί μια διφασική εξέλιξη:

1. **Φάση 1**: Προσαρμογή Greek2Piano (Ελληνικά τραγούδια → διασκευές πιάνου)
2. **Φάση 2**: Προσαρμογή Greek2Guitar (Ελληνικά τραγούδια → διασκευές κιθάρας)

Αυτή η στρατηγική πρώτα προσαρμόζει το μοντέλο στα ελληνικά μουσικά χαρακτηριστικά διατηρώντας την οικεία μορφή εξόδου πιάνου, και στη συνέχεια προσαρμόζει το μοντέλο στους περιορισμούς της κιθάρας. Η υπόθεση είναι ότι αυτή η ενδιάμεση προσαρμογή θα διατηρήσει καλύτερα τα ελληνικά μουσικά μοτίβα κατά τη φάση της τελικής οργανικής προσαρμογής.

0.6 Πειράματα και Αποτελέσματα

0.6.1 Μοντέλα για Διασκευές Ελληνικών Τραγουδιών σε Πιάνο

Διαμόρφωση Εκπαίδευσης: Για τη διασφάλιση αναπαραγώγιμων αποτελεσμάτων, χρησιμοποιήσαμε σταθερή τιμή seed 3407 και πραγματοποιήσαμε την εκπαίδευση σε NVIDIA GeForce GTX 1080 Ti GPU με batch size 8. Η παρακολούθηση της εκπαίδευσης έγινε μέσω του πλαισίου wandb, ενώ υλοποιήσαμε συναρτήσεις callback για την αποθήκευση του καλύτερου μοντέλου βάσει του validation loss.

Εκπαίδευση από την Αρχή: Εκπαιδεύσαμε ένα μοντέλο 59.1 εκατομμυρίων παραμέτρων χρησιμοποιώντας AdaFactor [50] με learning rate $1e-3$ για 3000 epochs (11 ώρες).

Το μοντέλο παρουσίασε overfitting, με το καλύτερο checkpoint στο epoch 544, επιδεικνύοντας τις προκλήσεις της εκπαίδευσης με περιορισμένα δεδομένα.

Μεταφορά Μάθησης: Εφαρμόσαμε transfer learning από το προ-εκπαιδευμένο μοντέλο Pop2Piano [1], εξερευνώντας δύο προσεγγίσεις:

Μερική: Παγώσαμε τα πρώτα στρώματα και εκπαιδεύσαμε μόνο τα δύο τελευταία στρώματα decoder και την κεφαλή language model. Η εκπαίδευση διήρκεσε 200 epochs (20 λεπτά) με καλύτερο checkpoint στο epoch 165.

Πλήρης: Ενημερώσαμε όλες τις παραμέτρους του μοντέλου, επιτρέποντας πλήρη προσαρμογή στα ελληνικά δεδομένα. Η εκπαίδευση διήρκεσε 500 epochs (2 ώρες) με καλύτερο checkpoint στο epoch 164.

Και οι δύο στρατηγικές transfer learning επέδειξαν σημαντικά καλύτερα αποτελέσματα από την εκπαίδευση από την αρχή, επιβεβαιώνοντας την αποτελεσματικότητα της προσέγγισης για σενάρια με περιορισμένα δεδομένα όπως η δημιουργία διασκευών ελληνικής μουσικής.

0.6.2 Μοντέλα για Διασκευές Τραγουδιών σε Κιθάρα

Διαμόρφωση Εκπαίδευσης: Η εκπαίδευση των μοντέλων δημιουργίας διασκευών κιθάρας πραγματοποιήθηκε σε NVIDIA GeForce GTX 1080 Ti GPU με batch size 8 και σταθερό seed 3407 για αναπαραγωγιμότητα. Η παρακολούθηση της εκπαίδευσης έγινε μέσω του framework wandb όπως και για τα παραπάνω μοντέλα. Για λόγους συνέπειας στην παρουσίαση, τα αποτελέσματα εκπαίδευσης που παρατίθενται στις επόμενες εικόνες προέρχονται από την πρώτη πτυχή της διασταυρούμενης επικύρωσης. Αντιθέτως, οι τελικές μετρικές αξιολόγησης που παρουσιάζονται στην Ενότητα 0.6.4 αποτυπώνουν τον μέσο όρο και τα διαστήματα εμπιστοσύνης 95% που υπολογίστηκαν σε όλες τις πτυχές(συνολικά 5).

Εκπαίδευση από την Αρχή: Το μοντέλο εκπαιδεύτηκε για 2000 epochs στο σύνολο δεδομένων Pop2Guitar με 40 ζεύγη τραγουδιού-διασκευής. Η καλύτερη απόδοση επιτεύχθηκε στο epoch 1974, αλλά το μοντέλο εμφάνισε σαφή σημάδια overfitting λόγω του περιορισμένου μεγέθους των δεδομένων.

Μεταφορά Μάθησης: Χρησιμοποιήσαμε το προεκπαιδευμένο μοντέλο Pop2Piano [1] ως σημείο εκκίνησης και εφαρμόσαμε δύο στρατηγικές βελτίωσης:

Μερική: Παγώσαμε τα περισσότερα στρώματα και ενημερώσαμε μόνο τα δύο τελευταία στρώματα decoder. Εκπαίδευση για 1000 epochs (25 λεπτά), καλύτερο checkpoint στο epoch 929.

Πλήρης: Ενημερώσαμε όλα τα στρώματα του μοντέλου. Η εκπαίδευση κράτησε 1000 epochs (35 λεπτά), με καλύτερο checkpoint στο epoch 174. Παρατηρήθηκε overfitting μετά από 1000 βήματα λόγω του μικρού μεγέθους δεδομένων.

0.6.3 Μοντέλα για Διασκευές Ελληνικών Τραγουδιών σε Κιθάρα

Η διαδοχική μας προσέγγιση για τη δημιουργία διασκευών ελληνικών τραγουδιών σε κιθάρα ακολουθεί την διφασική πορεία: Το προεκπαιδευμένο μοντέλο Pop2Piano χρησιμεύει ως βάση.

Φάση 1: Εκπαίδευση στο σύνολο δεδομένων GreekSong2Piano χρησιμοποιώντας την βέλτιστη διαμόρφωση, δημιουργώντας ένα μοντέλο δημιουργίας διασκευών πιάνου.

Φάση 2: Χρησιμοποιώντας το μοντέλο Greek2Piano ως αρχικοποίηση, εκτελούμε τελική προσαρμογή στο σύνολο δεδομένων Pop2Guitar με διασκευές κιθάρας, εφαρμόζοντας tokenization ειδικό για κιθάρα.

0.6.4 Μεθοδολογία Αξιολόγησης

Σε αυτήν την ενότητα παρουσιάζουμε τις μεθόδους που χρησιμοποιήθηκαν για την αξιολόγηση της απόδοσης των μοντέλων δημιουργίας διασκευών. Η αξιολόγηση περιλαμβάνει τόσο αντικειμενικά όσο και υποκειμενικά κριτήρια. Η αντικειμενική αξιολόγηση βασίζεται σε ποσοτικά μέτρα, ενώ η υποκειμενική αξιολόγηση στηρίζεται σε μελέτη χρηστών.

Ανάλυση Απόδοσης με Μετρικές Αξιολόγησης

Υιοθετούμε τις ακόλουθες μετρικές για την αξιολόγηση της ποιότητας των παραγόμενων διασκευών από πολλαπλές οπτικές γωνίες. Αυτές οι μετρικές αξιολογούν τόσο την ομοιότητα των διασκευών με το αρχικό τραγούδι όσο και την προσκόλλησή τους στις στυλιστικές συμβάσεις και την εσωτερική συνοχή που χαρακτηρίζει τις ανθρώπινες διασκευές.

Melody Chroma Accuracy: αξιολογεί την ομοιότητα μεταξύ δύο μονοφωνικών μελωδικών ακολουθιών. Η μελωδική γραμμή παίζει καθοριστικό ρόλο στην απόφαση για το αν μια διασκευή μοιάζει με το αρχικό τραγούδι. Ακολουθώντας τις οδηγίες από [1], υπολογίζουμε την MCA μεταξύ των φωνητικών που εξάγονται από το Spleeter [47] από τον ήχο και της κορυφαίας μελωδικής γραμμής που εξάγεται από το MIDI της διασκευής χρησιμοποιώντας τον αλγόριθμο skyline.

Αναγνώριση Διασκευών Τραγουδιών: Για να αξιολογήσουμε την ομοιότητα μεταξύ του αρχικού κομματιού και της παραγόμενης διασκευής, χρησιμοποιούμε μια μετρική εμπνευσμένη από την αναγνώριση διασκευών τραγουδιών, δηλαδή τη μετρική $\mathcal{Q}_{\max}$ [36]. Η μετρική $\mathcal{Q}_{\max}$ αξιολογεί την ομοιότητα του αρμονικού περιεχομένου μεταξύ της παραγόμενης cover και της αναφοράς, με χαμηλότερες τιμές να υποδεικνύουν στενότερη αντιστοιχία.

Επιπλέον, χρησιμοποιήσαμε ένα σύγχρονο μοντέλο CSI που συμμετείχε και κατετάγη τρίτο στον διαγωνισμό MIREX 2024 Cover Song Identification. Χρησιμοποιούμε το CoverHunter [40] για να εξάγουμε embeddings από τις αρχικές ηχογραφήσεις και τις διασκευές. Στη συνέχεια, υπολογίζουμε την απόσταση συνημίτονου μεταξύ των embeddings των αρχικών και των παραγόμενων διασκευών.

Ομοιότητα Βασισμένη σε Embeddings: Βασιζόμενοι στην ιδέα ότι μπορούμε να υπολογίσουμε την ομοιότητα με τη χρήση embeddings, χρησιμοποιήσαμε το MERT (Music undERstanding model with large-scale self-supervised Training) [51], ένα μοντέλο μεγάλης κλίμακας, αυτο-εποπτευόμενης μάθησης σχεδιασμένο για την κατανόηση ακουστικής μουσικής. Στα πειράματά μας, χρησιμοποιούμε το μοντέλο MERT-v1-95M, επιλέγοντας αυτή τη μικρότερη παραλλαγή λόγω των υπολογιστικών περιορισμών μας.

Αξιολόγηση Διασκευών Πιάνου: Ο Πίνακας 1 συνοψίζει την απόδοση των μοντέλων δημιουργίας διασκευών πιάνου σε διάφορες αντικειμενικές μετρικές. Για τις διασκευές πιάνου, οι στρατηγικές fine-tuning δείχνουν σαφείς βελτιώσεις τόσο έναντι του Pop2Piano baseline όσο και της εκπαίδευσης από την αρχή. Η προσέγγιση πλήρους fine-tuning επιτυγχάνει την καλύτερη απόδοση με MCA 0.443 ± 0.021, $\mathcal{Q}_{\max}$ 0.064 ± 0.013 και απόσταση

CoverHunter 0.146 ± 0.013, αντιπροσωπεύοντας βελτιώσεις 21.0% στην MCA και 14.% στο $\mathcal{Q}_{\max}$ σε σύγκριση με το baseline.

Πίνακας 1. *Μετρικές Αξιολόγησης για Παραγώμενες Διασκευές Πιάνου (πείραμα 5 πτυχών). Υψηλότερες τιμές MCA και ομοιότητας εμβυθισμάτων (MERT) είναι προτιμότερες, ενώ χαμηλότερες τιμές CSI $\mathcal{Q}_{\max}$ και αποστάσεις CoverHunter υποδηλώνουν καλύτερη απόδοση.*

Μοντέλο	**MCA (↑)**	**CSI ($\mathcal{Q}_{\max}$) (↓)**	**Απόσταση CoverHunter (↓)**	**Ομοιότητα MERT (↑)**
Pop2Piano [1]	0.363 ± 0.019	0.075 ± 0.017	0.159 ± 0.015	0.808 ± 0.007
Greek2Piano-Scratch	0.372 ± 0.020	0.100 ± 0.020	0.175 ± 0.012	0.802 ± 0.008
Greek2Piano-Partial	**0.443 ± 0.021**	0.068 ± 0.017	0.155 ± 0.013	0.809 ± 0.007
Greek2Piano-Full	0.439 ± 0.022	**0.064 ± 0.013**	**0.146 ± 0.013**	**0.811 ± 0.009**
Human Piano	0.389 ± 0.029	0.093 ± 0.028	0.142 ± 0.014	0.794 ± 0.017
Human Piano (Audio)	–	0.087 ± 0.026	0.134 ± 0.013	0.834 ± 0.007

Αξιολόγηση Διασκευών Κιθάρας:

Ο Πίνακας 2 συνοψίζει την απόδοση των μοντέλων δημιουργίας διασκευών κιθάρας. Στην δημιουργία διασκευών κιθάρας, οι διαφορές απόδοσης μεταξύ των στρατηγικών είναι σημαντικές. Και οι δύο προσεγγίσεις fine-tuning επιτυγχάνουν ισχυρά αποτελέσματα, με το πλήρως fine-tuned μοντέλο να φτάνει το υψηλότερο MCA (0.363 ± 0.042) και ομοιότητα MERT (0.783 ± 0.024), ενώ το μερικώς fine-tuned μοντέλο επιτυγχάνει την καλύτερη απόδοση CSI (0.152 ± 0.050 $\mathcal{Q}_{\max}$) και τη χαμηλότερη απόσταση CoverHunter (0.153 ± 0.010). Αντίθετα, το μοντέλο από την αρχή που εκπαιδεύτηκε σε μόλις 40 ζεύγη διασκευών κιθάρας έχει κακή απόδοση σε όλες τις μετρικές (0.189 ± 0.016 MCA, 0.576 ± 0.105 $\mathcal{Q}_{\max}$), αναδεικνύοντας τη σημασία της μεταφοράς μάθησης όταν εργαζόμαστε με περιορισμένα σύνολα δεδομένων.

Πίνακας 2. *Μετρικές αξιολόγησης για παραγώμενες διασκευές κιθάρας (πείραμα 5 πτυχών). Οι υψηλότερες τιμές MCA και ομοιότητας εμβυθισμάτων (MERT) είναι προτιμότερες, ενώ οι χαμηλότερες τιμές CSI $\mathcal{Q}_{\max}$ και αποστάσεις CoverHunter υποδηλώνουν καλύτερη απόδοση.*

Μοντέλο	**MCA (↑)**	**CSI ($\mathcal{Q}_{\max}$) (↓)**	**Απόσταση CoverHunter (↓)**	**Ομοιότητα MERT (↑)**
Pop2Guitar-Scratch	0.189 ± 0.016	0.576 ± 0.105	0.181 ± 0.030	0.735 ± 0.007
Pop2Guitar-Partial	0.358 ± 0.043	**0.152 ± 0.050**	**0.153 ± 0.010**	0.781 ± 0.016
Pop2Guitar-Full	**0.363 ± 0.042**	0.169 ± 0.062	0.156 ± 0.014	**0.783 ± 0.024**
Human Guitar	0.288 ± 0.018	0.211 ± 0.053	0.168 ± 0.022	0.777 ± 0.014

Αξίζει να σημειωθεί ότι τα μοντέλα μας έχουν καλύτερα αποτελέσματα από αυτά των ανθρώπινων διασκευών σε πολλαπλά αντικειμενικά μετρικά, όπως το MCA (0.363 ± 0.042 έναντι 0.288 ± 0.018 για την κιθάρα), παρά το γεγονός ότι οι ανθρώπινες διασκευές λαμβάνουν ανώτερες υποκειμενικές αξιολογήσεις. Αυτή η φαινομενική αντίφαση αντανακλά μια θεμελιώδη διαφορά προσέγγισης: ενώ τα μοντέλα μας στοχεύουν στην πιστότητα της μελωδίας, οι ανθρώπινοι διασκευαστές δίνουν προτεραιότητα στην καλλιτεχνική ερμηνεία έναντι της κατά λέξη αναπαραγωγής, εισάγοντας δημιουργικές παραλλαγές και τεχνικές που σχετίζονται με το εκάστοτε όργανο. Αυτές οι παρεμβάσεις ενισχύουν την εκφραστικότητα της μουσικής, αλλά μειώνουν την μετρήσιμη ομοιότητα. Αυτό το μοτίβο συμφωνεί με τα ευρήματα της αρχικής δημοσίευσης Pop2Piano [1], όπου οι ανθρώπινες διασκευές σημείωσαν παρόμοια χαμηλότερες βαθμολογίες στα υπολογιστικά μετρικά, αλλά έλαβαν υψηλότερες

υποκειμενικές αξιολογήσεις. Τα υψηλότερα σκορ ομοιότητας MERT για τις ανθρώπινες διασκευές υποδηλώνουν ότι αυτές οι δημιουργικές αποκλίσεις, παρόλο που μειώνουν την ακρίβεια σε επίπεδο νότας, συμβάλλουν τελικά στη συνολική μουσική ποιότητα που εκτιμούν οι ακροατές.

Ενδιαφέρον παρουσιάζει το γεγονός ότι, όταν οι ίδιες ανθρώπινες εκτελέσεις αξιολογούνται απευθείας στη μορφή ηχογράφησής τους (*Human Piano (Audio)*), επιτυγχάνουν όχι μόνο υψηλότερη ομοιότητα MERT, αλλά και βελτιωμένη απόσταση CoverHunter και CSI $Q_{\max}$, υπερέχοντας τόσο των ανθρώπινων διασκευών σε μορφή MIDI όσο και όλων των εξόδων που παράγουν τα μοντέλα. Αυτή η απόκλιση αναδεικνύει πώς η διαδικασία μεταγραφής, προεπεξεργασίας, αναπαραγωγής εισάγει υποβαθμίσεις που καταστέλλουν τις μετρικές.

Αντίληψη Χρηστών και Υποκειμενική Αξιολόγηση Ποιότητας

Για την υποκειμενική αξιολόγηση, διεξήγαμε μια μελέτη χρηστών. Στην μελέτη μας πήραν μέρος 26 μη-επαγγελματίες συμμετέχοντες, οι οποίοι αξιολόγησαν αποσπάσματα 10 δευτερολέπτων από τραγούδια του test set σε τρεις διαστάσεις: Ομοιότητα με το Πρωτότυπο (SI), Μουσική Συνοχή (CO) και Απόλαυση Ακροατή (LE). Τα αποσπάσματα παρουσιάστηκαν ανώνυμα σε τυχαία σειρά για να εξασφαλιστεί αμερόληπτη αξιολόγηση.

Οι συμμετέχοντες κλήθηκαν να ακούσουν αυτά τα ηχητικά κλιπ και να παρέχουν βαθμολογίες σε κλίμακα Likert 5 βαθμών για τις ακόλουθες πτυχές:

- **Ομοιότητα με το Πρωτότυπο (SI):** Ο βαθμός ομοιότητας μεταξύ των εκτελέσεων πιάνου/κιθάρας και του αρχικού τραγουδιού.
- **Μουσική Συνοχή (CO):** Ο βαθμός αντιληπτής ροής στη μουσική, αντιπροσωπεύοντας την ομαλότητα και συνοχή των εκτελέσεων πιάνου/κιθάρας.
- **Απόλαυση Ακροατή (LE):** Πόσο αρέσει στους συμμετέχοντες η διασκευή πιάνου/κιθάρας συνολικά.

Πίνακας 3. *Μετρικές Αξιολόγησης για Παραγόμενες Διασκευές Πιάνου. Υψηλότερες τιμές για Ομοιότητα, Μουσική Συνοχή και Απόλαυση Ακροατή δείχνουν καλύτερη απόδοση.*

Μοντέλο	Ομοιότητα με το Πρωτότυπο (↑)	Μουσική Συνοχή (↑)	Απόλαυση Ακροατή (↑)
Pop2Piano [1]	2.29 ± 0.20	2.60 ± 0.26	2.40 ± 0.25
Greek2Piano-Scratch	1.81 ± 0.21	2.42 ± 0.26	1.97 ± 0.21
Greek2Piano-Partial	2.67 ± 0.22	2.60 ± 0.23	2.46 ± 0.24
Greek2Piano-Full	$\mathbf{2.94 \pm 0.21}$	$\mathbf{2.91 \pm 0.23}$	$\mathbf{2.72 \pm 0.25}$
Human Piano	4.06 ± 0.23	3.94 ± 0.25	3.78 ± 0.28

Τα αποτελέσματα της υποκειμενικής αξιολόγησης στον Πίνακα 3 καταδεικνύουν ισχυρή αντιστοιχία με τις αντικειμενικές μας μετρήσεις. Για τις διασκευές πιάνου, η προσέγγιση πλήρους fine-tuning έλαβε τις υψηλότερες βαθμολογίες σε όλες τις διαστάσεις (αναφέρονται ως μέσος όρος με διαστήματα εμπιστοσύνης 95%: SI: 2,94±0,21, CO: 2,91±0,23, LE: 2,72±0,25) προσεγγίζοντας το ανθρώπινο σημείο αναφοράς και ξεπερνώντας τόσο τη βασική γραμμή Pop2Piano όσο και το μοντέλο από την αρχή. Αυτό επιβεβαιώνει ότι το fine-tuning

Πίνακας 4. *Μετρικές Αξιολόγησης για Παραγόμενες Διασκευές Κιθάρας. Υψηλότερες τιμές για Ομοιότητα, Μουσική Συνοχή και Απόλαυση Ακροατή δείχνουν καλύτερη απόδοση.*

Μοντέλο	**Ομοιότητα με το Πρωτότυπο(↑)**	**Μουσική Συνοχή (↑)**	**Απόλαυση Ακροατή(↑)**
Pop2Guitar-Scratch	1.54 ± 0.28	1.77 ± 0.27	1.54 ± 0.24
Pop2Guitar-Partial	$\mathbf{2.56 \pm 0.29}$	$\mathbf{2.50 \pm 0.28}$	$\mathbf{2.29 \pm 0.31}$
Pop2Guitar-Full	2.27 ± 0.25	2.35 ± 0.26	2.06 ± 0.26
Human Guitar	3.13 ± 0.26	2.87 ± 0.34	2.71 ± 0.33

ενισχύει όχι μόνο τις τεχνικές μετρήσεις αλλά και την αντιληπτή μουσικότητα και απόλαυση. Τα μοντέλα κιθάρας στον Πίνακα 6.5 δείχνουν παρόμοιο μοτίβο αλλά με πιο έντονες διαφορές. Το μοντέλο από την αρχή σημείωσε χαμηλές επιδόσεις σε όλες τις μετρήσεις (SI: 1,54±0,28, CO: 1,77±0,27, LE: 1,54±0,24), ενώ τα fine-tuned μοντέλα πέτυχαν σημαντικά υψηλότερες βαθμολογίες, με την προσέγγιση μερικού fine-tuning να λαμβάνει ιδιαίτερα υψηλές βαθμολογίες για ομοιότητα (2,56±0,29) και συνοχή (2,50±0,28).

Πίνακας 5. *Μετρικές Αξιολόγησης για Διασκευές Ελληνικών τραγουδιών σε Κιθάρα. Υψηλότερες τιμές για Ομοιότητα, Μουσική Συνοχή και Απόλαυση Ακροατή δείχνουν καλύτερη απόδοση.*

Μοντέλο	**Ομοιότητα με το Πρωτότυπο(↑)**	**Μουσική Συνοχή (↑)**	**Απόλαυση Ακροατή (↑)**
Base (No Fine-tuning)	2.37 ± 0.32	2.10 ± 0.31	1.90 ± 0.29
Sequential-Partial	$\mathbf{3.31} \pm 0.33$	$\mathbf{3.00} \pm 0.33$	$\mathbf{3.00} \pm 0.33$
Sequential-Full	3.19 ± 0.28	2.67 ± 0.29	2.50 ± 0.29
Human (Greek Guitar)	4.17 ± 0.28	3.85 ± 0.31	3.65 ± 0.38

Ιδιαίτερα αξιοσημείωτο είναι το πείραμα διαδοχικού fine-tuning (Greek2Guitar) που αποδεικνύει τις δυνατότητες της στρωματοποιημένης μεταφοράς μάθησης. Το μερικώς εκπαιδευμένο διαδοχικό μοντέλο πέτυχε τη υψηλότερη βαθμολογία ομοιότητας (3.31±0.33) μεταξύ όλων των μοντέλων κιθάρας, πλησιάζοντας το ανθρώπινο σημείο αναφοράς (4.17±0.28). Αυτό υποδηλώνει ότι η διαδρομή μεταφοράς γνώσης από το δυτικό ποπ πιάνο στο ελληνικό πιάνο και στη συνέχεια στην κιθάρα συλλαμβάνει αποτελεσματικά σημαντικά μουσικά χαρακτηριστικά που βελτιώνουν την αντιληπτή ποιότητα των παραγόμενων διασκευών.

0.7 Συμπεράσματα και Μελλοντική Κατευθύνσεις

0.7.1 Συμπεράσματα

Η δημιουργία διασκευών τραγουδιών αποτελεί σημαντική πρόκληση στο πεδίο της μουσικής ανάκτησης πληροφοριών, απαιτώντας από τα συστήματα να διατηρούν την ουσία των πρωτότυπων συνθέσεων ενώ ταυτόχρονα τις προσαρμόζουν σε συγκεκριμένα όργανα και στυλ. Η παρούσα διπλωματική εργασία αντιμετώπισε δύο περιορισμούς του πεδίου: την έλλειψη δεδομένων εκπαίδευσης για μη-δυτικές μουσικές παραδόσεις και την απουσία μοντέλων δημιουργίας διασκευών για όργανα πέραν του πιάνου. Μέσα από συστηματική διερεύνηση προσεγγίσεων μεταφοράς μάθησης και τη δημιουργία εξειδικευμένων συνόλων δεδομένων, προτείναμε στρατηγικές για την αυτόματη δημιουργία διασκευών σε σενάρια περιορισμένων πόρων.

Η κύρια συνεισφορά μας στα σύνολα δεδομένων είναι το GreekSong2Piano dataset, που περιλαμβάνει 659 ελληνικά τραγούδια και τις αντίστοιχες διασκευές τους για πιάνο, συνολικά 41 ώρες μουσικής σε οκτώ διαφορετικά ελληνικά είδη, όπως Ρεμπέτικο, Λαϊκό και Έντεχνο. Το σύνολο αυτό αποτυπώνει τα χαρακτηριστικά της ελληνικής μουσικής παράδοσης, παρέχοντας την πρώτη συγχρονισμένη συλλογή ειδικά σχεδιασμένη για την αυτόματη δημιουργία διασκευών ελληνικής μουσικής. Επιπλέον, δημιουργήσαμε το Pop2Guitar dataset με 40 ζεύγη τραγουδιού-κιθάρας, επιτρέποντας την εξερεύνηση της προσαρμογής πεδίου σε όργανα πέραν του πιάνου.

Η συστηματική ανάλυσή μας έδειξε σαφή πλεονεκτήματα απόδοσης για τις προσεγγίσεις μεταφοράς μάθησης έναντι της εκπαίδευσης από την αρχή. Συγκρίνοντας την εκπαίδευση από την αρχή, το μερικό fine-tuning και το πλήρες fine-tuning στο ελληνικό μας σύνολο δεδομένων, και οι δύο προσεγγίσεις fine-tuning ξεπέρασαν την απόδοση της βάσης αναφοράς Pop2Piano, με το μερικό fine-tuning να φτάνει το υψηλότερο MCA των 0.443 ± 0.021, σημειώνοντας βελτίωση 21,0% σε σχέση με τη βάση αναφοράς. Αξιοσημείωτο είναι ότι ακόμη και το μοντέλο που εκπαιδεύτηκε αποκλειστικά σε ελληνική μουσική ανταγωνίστηκε στενά το αρχικό Pop2Piano σε ελληνικά τραγούδια, επιβεβαιώνοντας την αξία της εκπαίδευσης ειδικά προσανατολισμένης στο πεδίο. Στην περίπτωση της δημιουργίας διασκευών για κιθάρα, η μεταφορά μάθησης αποδείχθηκε ακόμη πιο καθοριστική, με τις προσεγγίσεις fine-tuning να υπερέχουν σαφώς της εκπαίδευσης από την αρχή, λόγω των περιορισμένων διαθέσιμων δεδομένων.

Επιπλέον, εισαγάγαμε μια νέα στρατηγική διαδοχικού fine-tuning, η οποία περιλαμβάνει προσαρμογή πεδίου πολλαπλών βημάτων: από δυτικές ποπ διασκευές πιάνου σε ελληνικές διασκευές πιάνου και, στη συνέχεια, σε διασκευές κιθάρας. Αυτή η προσέγγιση απέδωσε ιδιαίτερα θετικά αποτελέσματα, με το μερικά εκπαιδευμένο διαδοχικό μοντέλο να καταγράφει τις υψηλότερες βαθμολογίες ομοιότητας (3.31 ± 0.33) μεταξύ των μοντέλων κιθάρας, πλησιάζοντας την ανθρώπινη απόδοση (4.17 ± 0.28). Αυτό δείχνει ότι η γνώση μπορεί να μεταφερθεί αποτελεσματικά τόσο σε πολιτισμικά όσο και σε οργανικά όρια μέσω προσεκτικά σχεδιασμένων μονοπατιών προσαρμογής.

Το πλαίσιο αξιολόγησής μας συνδύασε αντικειμενικές μετρήσεις και υποκειμενική αξιολόγηση, προσφέροντας μια ολοκληρωμένη εκτίμηση της ποιότητας. Χρησιμοποιήσαμε το Melody Chroma Accuracy (MCA), μετρήσεις αναγνώρισης διασκευών τραγουδιών, καθώς και προσεγγίσεις βασισμένες σε embeddings με πρωτοποριακά μοντέλα όπως το CoverHunter και το MERT. Η υποκειμενική αξιολόγηση μέσω μελετών χρηστών επικύρωσε τα ευρήματά μας, δείχνοντας στενή συσχέτιση μεταξύ των υπολογιστικών μετρήσεων και της ανθρώπινης αντίληψης της ποιότητας των διασκευών.

0.7.2 Περιορισμοί και Μελλοντικές Κατευθύνσεις

Περιορισμοί

Ενώ η παρούσα εργασία καταδεικνύει τις δυνατότητες της μεταφοράς μάθησης για τη διαπολιτισμική και διαοργανική δημιουργία διασκευών, αρκετοί περιορισμοί αναδεικνύουν πεδία προς περαιτέρω βελτίωση. Τα σύνολα δεδομένων μας παρουσιάζουν εγγενείς περιορισμούς ποιότητας. Σε αντίθεση με επαγγελματικά ηχογραφημένα σύνολα δεδομένων όπως

το MAESTRO [52], το οποίο προσφέρει εκτελέσεις πιάνου με υψηλής ακρίβειας χρονική ευθυγράμμιση (περίπου 3ms) μεταξύ των ετικετών νότας και των κυματομορφών ήχου, ή το GuitarSet [53] με τις εξαφωνικές καταγραφές του από ειδικά πηνία, τα δικά μας δεδομένα προέρχονται από το YouTube και συγχρονίζονται μέσω υπολογιστικών μεθόδων οι οποίες, παρότι αποτελεσματικές, δεν επιτυγχάνουν το ίδιο επίπεδο ακρίβειας. Αυτό ενδέχεται να εισάγει χρονικές αποκλίσεις που επηρεάζουν την ποιότητα της εκπαίδευσης των μοντέλων. Επιπλέον, η υποκειμενική μας αξιολόγηση, αν και παρείχε χρήσιμες πληροφορίες, βασίστηκε στη συμμετοχή 26 μη επαγγελματιών ακροατών, γεγονός που υποδεικνύει ότι μελέτες μεγαλύτερης κλίμακας, οι οποίες θα ενσωματώνουν πιο ετερόκλητο ακροατήριο και επαγγελματίες μουσικούς, θα μπορούσαν να προσφέρουν πιο ισχυρή επιβεβαίωση των ευρημάτων μας.

Επίσης, τεχνικοί περιορισμοί περιόρισαν το πειραματικό μας πεδίο. Η εξάρτησή μας από προϋπάρχοντα αρχεία MIDI από το MuseScore για το σύνολο δεδομένων Pop2Guitar επιβλήθηκε από την κακή απόδοση των τρεχόντων μοντέλων αυτόματης μουσικής μεταγραφής σε ηχογραφήσεις κιθάρας. Παρά τον σχεδιασμό του MT3 για μεταγραφή πολλαπλών οργάνων [10], συχνά αναγνώριζε λανθασμένα τις εκτελέσεις κιθάρας ως άλλα όργανα, καθιστώντας το ακατάλληλο για τη δημιουργία των συγχρονισμένων ζευγών ήχου–MIDI που είναι απαραίτητα για την προσέγγισή μας.

Αυτός ο περιορισμός μας ανάγκασε να εργαστούμε με ένα σημαντικά μικρότερο σύνολο δεδομένων κιθάρας (40 ζεύγη) συγκριτικά με το σύνολο δεδομένων πιάνου μας (659 ζεύγη). Επιπλέον, τα πειράματα μας περιορίστηκαν από τους υπολογιστικούς μας πόρους. Όλα τα πειράματα διεξήχθησαν στον διακομιστή του εργαστηρίου SLP-NTUA, εξοπλισμένο με δύο GPU των 12ΓΒ (NVIDIA GeForce GTX 1080 Ti και GeForce GTX TITAN X), γεγονός που περιόρισε την ικανότητά μας να πειραματιστούμε με μεγαλύτερα μήκη πλαισίου και μεγέθη παρτίδων στη διαδικασία εκπαίδευσης των μοντέλων.

Παρομοίως, κατά την αξιολόγηση, περιοριστήκαμε στη χρήση του MERT-95M [51] αντί του πιο ικανού μοντέλου MERT-330M λόγω περιορισμών μνήμης, επηρεάζοντας πιθανώς την ποιότητα των αξιολογήσεων ομοιότητας βασισμένων σε embeddings.

Μελλοντικές Κατευθύνσεις

Μελλοντική έρευνα θα πρέπει να εξερευνήσει διευρυμένα πολιτισμικά πεδία και μουσικά ύφη πέραν της ελληνικής και της δυτικής ποπ παράδοσης που εξετάζονται στην παρούσα εργασία. Η ελληνική μουσική, με τα χαρακτηριστικά ρυθμικά μοτίβα και τις ιδιαίτερες δομικές ιδιομορφίες της [42], αποτέλεσε αποτελεσματική περίπτωση δοκιμής για τη διαπολιτισμική προσαρμογή, αλλά οι αρχές που παρουσιάζονται εδώ μπορούν να επεκταθούν και σε άλλες παραδόσεις με μοναδικά χαρακτηριστικά, όπως η ινδική κλασική μουσική [54], τα αραβικά συστήματα maqam [55] ή οι μουσικές μορφές της Ανατολικής Ασίας [56]. Τέτοιες επεκτάσεις θα δοκίμαζαν περαιτέρω τη γενικευσιμότητα των προσεγγίσεων transfer learning σε ένα ευρύτερο φάσμα μουσικών παραμέτρων.

Όπως η αυτόματη μεταγραφή μουσικής (AMT) έχει επεκταθεί επιτυχώς από πιανοκεντρικά συστήματα σε ποικίλες οικογένειες οργάνων, έτσι και η παραγωγή διασκευών θα μπορούσε να ακολουθήσει παρόμοια πορεία, εφόσον υπάρξει κατάλληλη ανάπτυξη συνόλων

δεδομένων. Πρόσφατες εξελίξεις στην AMT έχουν επιτύχει μεταγραφή σε έγχορδα (βιολί, τσέλο) [57], ξύλινα πνευστά (φλάουτο) [58], κρουστά [59] και παραδοσιακά εθνολογικά όργανα όπως το αραβικό φλάουτο [60]. Η παραγωγή διασκευών θα μπορούσε επίσης να επεκταθεί σε αυτά τα όργανα, αν και αυτό προϋποθέτει την ανάπτυξη συγχρονισμένων συνόλων δεδομένων μεγάλης κλίμακας και πιο εξελιγμένων σχημάτων tokenization για να ληφθούν υπόψη τεχνικές που είναι ειδικές για κάθε όργανο.

Μια ιδιαίτερα υποσχόμενη προέκταση περιλαμβάνει τη δημιουργία ολοκληρωμένων συστημάτων από άκρο σε άκρο που μετατρέπουν απευθείας ηχητική είσοδο σε παρτιτούρα έτοιμη για εκτέλεση. Βασιζόμενα στην υπάρχουσα ροή audio-to-MIDI, τέτοια συστήματα θα μπορούσαν να ενσωματώνουν υπομονάδες μετα-επεξεργασίας για τη δημιουργία μουσικής σημειογραφίας. Πρόσφατη εργασία στον τομέα της μετατροπής από MIDI σε παρτιτούρα, όπως το MIDI2ScoreTransformer [61], καταδεικνύει τη δυνατότητα μετατροπής συμβολικής μουσικής σε παρτιτούρα πιάνου. Ομοίως, εξειδικευμένα συστήματα για κιθάρα μπορούν να αξιοποιήσουν μοντέλα μετατροπής από MIDI σε tablature [62] για την παραγωγή κατάλληλης σημειογραφίας ανά όργανο. Ένα ολοκληρωμένο σύστημα θα μπορούσε δυνητικά να ενοποιήσει τρία στάδια: (1) μετατροπή ήχου σε συμβολική αναπαράσταση μέσω των εκπαιδευμένων μοντέλων παραγωγής διασκευών, (2) μετατροπή της συμβολικής αναπαράστασης σε σημειογραφία μέσω εξειδικευμένων μοντέλων απεικόνισης και (3) κοινή βελτιστοποίηση σε ολόκληρη τη ροή.

Τέλος, μία ακόμη κατεύθυνση αφορά την εξαρτώμενη παραγωγή διασκευών με βάση κειμενική ή πολυτροπική είσοδο. Μέσω της ενσωμάτωσης διαφορετικών τύπων εισόδου όπως συγχορδίες, μελωδικές γραμμές, στίχοι και περιγραφικά κείμενα οι χρήστες όχι μόνο μπορούν να αλληλεπιδρούν πιο δυναμικά με τη διαδικασία δημιουργίας μουσικής, αλλά και να αποκτούν μεγαλύτερο και πιο λεπτομερή έλεγχο στο τελικό αποτέλεσμα [63]. Μέσα από τη χρήση μοντέλων όπως το ChatMusician [64], το Llark [65], ή την επέκταση του μοντέλου T5 [5], μελλοντικά μοντέλα θα μπορούσαν να δέχονται περιγραφές φυσικής γλώσσας, όπως «δημιούργησε μία μελαγχολική διασκευή πιάνου στο ύφος μιας κλασικής μπαλάντας» ή «παρήγαγε μία ζωηρή διασκευή για κιθάρα κατάλληλη για φεστιβάλ παραδοσιακής μουσικής».

Ελπίζουμε ότι η εργασία μας θα προσελκύσει περισσότερη ερευνητική προσοχή στο απαιτητικό πρόβλημα της δημιουργίας μουσικών διασκευών και θα εμπνεύσει νέα ερευνητικά εγχειρήματα σχετικά με τη διαπολιτισμική ανταλλαγή στη μουσική μέσω υπολογιστικών προσεγγίσεων. Μέσα από τη γεφύρωση διαφορετικών μουσικών παραδόσεων και μορφών οργάνων, συστήματα όπως το δικό μας ενδέχεται να συμβάλουν τόσο σε δημιουργικές εφαρμογές όσο και σε βαθύτερη υπολογιστική κατανόηση της μουσικής μετάφρασης πέρα από πολιτισμικά όρια.

Chapter 1

Introduction

1.1 Motivation

Music covers represent one of humanity's most enduring forms of artistic expression, transcending cultural boundaries and historical periods. From ancient folk traditions where songs evolved through oral transmission to classical composers creating arrangements of existing works, the practice of reinterpreting music has been central to musical culture. In contemporary times, cover versions serve multiple purposes: they preserve musical heritage, introduce songs to new audiences, and allow artists to express their unique interpretative vision. However, creating quality covers traditionally requires substantial musical expertise, instrument-specific knowledge, and considerable time investment. Musicians must analyze the original composition, understand its harmonic structure, adapt it to their chosen instrument's capabilities, and develop an arrangement that maintains the song's essence while showcasing their artistic perspective.

The recent revolution in artificial intelligence and deep learning has opened unprecedented opportunities for automated content generation across multiple domains. Breakthrough models in natural language processing, computer vision, and audio synthesis have demonstrated remarkable capabilities in generating human quality text, images, and music. In the music domain specifically, we have witnessed significant advances from unconditional music generation systems to more sophisticated conditional approaches that can generate music based on specific constraints or inputs. This progression naturally leads to the possibility of cover generation, transforming existing songs into arrangements for specific instruments while preserving their musical identity. Such systems could democratize music arrangement, making it accessible to musicians regardless of their theoretical background or arrangement experience.

Despite these promising developments, several challenges impede progress in cover generation. The most fundamental obstacle is data scarcity: unlike text or image datasets, synchronized song-cover pairs suitable for training are extremely limited, particularly for instruments beyond piano and the pop music genre. This creates a low-resource learning scenario that traditional deep learning approaches struggle to address effectively. Additionally, the evaluation of generated covers presents difficulties, as musical quality assessment involves both objective measures of similarity and subjective judgments of artistic merit.

This thesis aims to address these challenges and advance the field in both technical and cultural dimensions. To overcome data limitations, we develop two datasets including the GreekSong2Piano dataset, which captures the unique characteristics of Greek musical traditions, and the Pop2Guitar dataset, expanding cover generation beyond piano to guitar arrangements. Our methodological approach leverages transfer learning and domain adaptation techniques to effectively utilize limited training data, demonstrating how knowledge from larger datasets can be adapted to specialized musical contexts. We establish a comprehensive evaluation framework that combines objective metrics with subjective assessments, providing a more complete picture of generation quality. By extending automatic cover generation to Greek music and guitar arrangements, this work creates a foundation for culturally-aware and instrument-diverse music arrangement systems, opening new possibilities for preserving musical heritage while expanding creative expression through technology.

1.2 Contribution

This thesis contributes to the MIR field regarding cover song generation. Given that the specific subject of cover generation from audio input has been limited to piano covers of Western pop music, and that low-resource musical domains such as Greek music and alternative instruments like guitar have not been extensively studied, this thesis explores transfer learning approaches and domain adaptation techniques for cover generation across different musical styles and instruments. Firstly, we address the challenge of generating piano covers for Greek music by creating the first synchronized dataset of Greek songs and their corresponding piano covers, and by exploring different training strategies including training from scratch, partial fine-tuning, and full fine-tuning of pre-trained models. Then, we investigate cross-instrument domain adaptation by developing guitar cover generation models, exploring how knowledge learned from piano cover generation can be transferred to guitar arrangements. Specifically, we examine a sequential fine-tuning approach that leverages the path from Western pop piano covers to Greek piano covers to guitar covers, which constitutes a multi-step domain adaptation strategy. Finally, we establish a comprehensive evaluation framework that combines melody-based metrics with embedding-based similarity measures and cover song identification techniques.

The main contributions of this thesis are:

- **GreekSong2Piano dataset:** A new dataset of 659 Greek songs accompanied by piano covers in audio and MIDI formats, with manual annotations capturing genre-specific traits and lyrical content.
- **Pop2Guitar dataset:** A dataset of 40 pop songs and corresponding guitar covers arranged by diverse musicians, enabling extension beyond piano-centric approaches.
- **Analysis of training strategies for low-resource cover generation:** Comparison of training from scratch, partial fine-tuning, and full fine-tuning approaches, demon-

strating the effectiveness of transfer learning in low-resource scenarios enabling adaptation across stylistic and instrumental domains.

- **Objective evaluation methodology:** An evaluation protocol leveraging pre-trained *cover song identification* and *acoustic music understanding* models to objectively assess generated covers.

1.3 Thesis Outline

This thesis is structured as follows:

- In Chapter 2, we provide the theoretical background, covering machine learning fundamentals, deep learning architectures, music representations, and music generation techniques.
- In Chapter 3, we examine related work relevant to our task, including automatic music transcription, music transformation, cover song identification, and cover generation approaches.
- In Chapter 4, we review existing datasets in the field and introduce our newly created datasets for Greek piano covers and guitar cover generation.
- In Chapter 5, we detail our methodology, including the preprocessing pipeline, model architecture, and training strategies for both piano and guitar cover generation.
- In Chapter 6, we describe our experimental setup for each training approach and evaluate our models using both objective metrics and subjective user studies.
- In Chapter 7, we summarize our findings and outline potential directions for future work.

Chapter 2

Machine Learning

2.1 Overview of Machine Learning

Machine Learning (ML) is a subfield of artificial intelligence that focuses on developing algorithms and systems that can learn and make decisions from data without being explicitly programmed for every specific task [18]. Rather than following pre-written instructions, machine learning systems identify patterns in data and use these patterns to make predictions or decisions about new, unseen information.

The concept of machine learning was first formally introduced by Arthur Samuel in 1959, who defined it as a "field of study that gives computers the ability to learn without being explicitly programmed" [66]. Samuel demonstrated this concept by creating a checkers-playing program that improved its performance through self-play, learning strategies and tactics that were not directly programmed by its creator. This early example illustrated the fundamental promise of machine learning: systems that could adapt and improve their performance based on experience.

Machine learning approaches can be broadly categorized into three main paradigms, each addressing different types of learning problems:

Supervised Learning involves training algorithms on labeled datasets, where both input data and desired outputs are provided [19]. The system learns to map inputs to outputs by identifying patterns in the training examples. Common applications include image classification, speech recognition, and regression tasks. In the context of music, supervised learning might involve training a model to classify songs by genre using labeled examples of different musical styles [67, 68].

Unsupervised Learning works with data that has no predefined labels or target outputs [20]. These algorithms seek to discover hidden patterns, structures, or relationships within the data. Clustering algorithms that group similar data points and dimensionality reduction techniques that identify the most important features in complex datasets are typical examples. In music applications, unsupervised learning has been applied to diverse tasks including music segmentation and ocal feature discovery for genre classification [69, 70] .

Reinforcement Learning takes a different approach, where agents learn through interaction with an environment, receiving rewards or penalties based on their actions [21]. The system learns to maximize cumulative rewards over time through trial and error.

This paradigm has shown remarkable success in game-playing systems and robotics, and in music, it has been applied to tasks such as interactive composition and performance systems [71, 72].

The transition from traditional rule-based programming to data-driven machine learning represents a fundamental shift in how we approach complex problems. Traditional programming requires developers to explicitly define rules and logic for every possible scenario, which becomes increasingly difficult as problems grow in complexity. Machine learning, by contrast, allows systems to discover these rules automatically from data, enabling solutions for problems where explicit rule formulation would be impractical or impossible.

This data-driven approach has proven particularly valuable in domains like music, where the complexity of human creativity and cultural expression makes it difficult to encode comprehensive rules manually. The ability of machine learning systems to learn from large collections of musical data has opened new possibilities for understanding and generating music, setting the foundation for the deep learning advances that have transformed the field in recent years.

2.2 Deep Learning Fundamentals

Deep learning represents a significant evolution from traditional machine learning approaches, characterized by the use of artificial neural networks with multiple layers that can automatically learn hierarchical representations from data [2]. While early neural networks were limited by computational constraints, advances in hardware capabilities, optimization techniques, and architectural design have enabled the development of increasingly sophisticated models capable of handling complex tasks across diverse domains.

2.2.1 Key Architectures

Several fundamental architectural innovations have driven the success of deep learning across different domains:

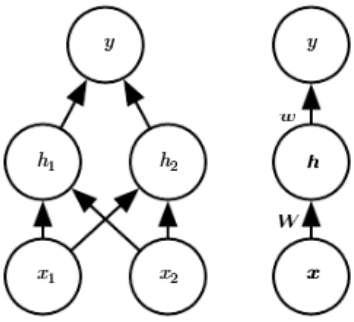


Figure 2.1. *An example of a feedforward network, drawn in two different styles. Source: [2]*

Multi-Layer Perceptrons (MLPs) serve as the foundation of deep learning, consisting of fully connected layers that transform input representations through successive non-linear transformations [2]. While conceptually simple, MLPs remain effective for many structured data problems and serve as building blocks for more complex architectures.

Convolutional Neural Networks (CNNs) introduced the concept of local connectivity and weight sharing, making them particularly well-suited for processing grid-like data such as images and spectrograms. The foundational work by LeCun et al. demonstrated that CNNs could effectively learn hierarchical features from raw pixel data, achieving state-of-the-art performance on handwritten digit recognition [22]. In music applications, CNNs have proven effective for tasks involving time-frequency representations, where local patterns in both time and frequency dimensions carry important musical information.

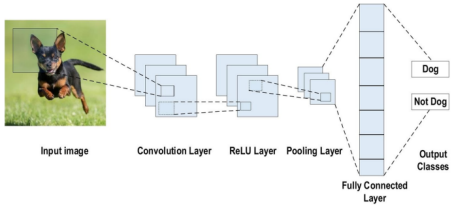


Figure 2.2. *An example of CNN architecture for image classifcation. Source: [3]*

Recurrent Neural Networks (RNNs) and their variants were designed to handle sequential data by maintaining internal memory states. However, traditional RNNs suffered from the same vanishing gradient problems that plagued deep feedforward networks. This limitation was addressed by the introduction of Long Short-Term Memory (LSTM) networks by Hochreiter and Schmidhuber [23], which used gating mechanisms to selectively retain and forget information over long sequences. LSTMs became particularly important for music generation and analysis tasks, where temporal dependencies and long-range relationships between musical events are crucial for maintaining coherence and structure.

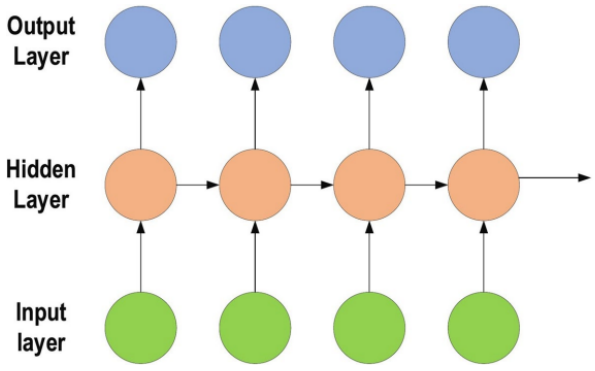


Figure 2.3. *Typical unfolded RNN diagram. Source: [3]*

2.2.2 Attention Mechanisms and Transformers

The introduction of attention mechanisms represented a fundamental breakthrough in sequence modeling, addressing the limitations of RNNs in capturing long-range depen-

dencies. The revolutionary "Attention Is All You Need" paper by Vaswani et al. demonstrated that attention mechanisms alone could achieve superior performance compared to recurrent and convolutional approaches [4].

Self-Attention extends this concept by allowing each position in a sequence to attend to all other positions, enabling the model to capture complex relationships within the sequence. This mechanism has proven particularly powerful for understanding musical structure, where relationships between distant musical events (such as motifs that appear throughout a composition) are essential for maintaining coherence.

Transformer Architecture builds upon self-attention to create a fully parallel processing framework that eliminates the sequential bottlenecks inherent in RNNs [4]. The Transformer consists of encoder and decoder components, each containing multiple layers of self-attention and feed-forward networks. This architecture has achieved state-of-the-art results across numerous sequence-to-sequence tasks.

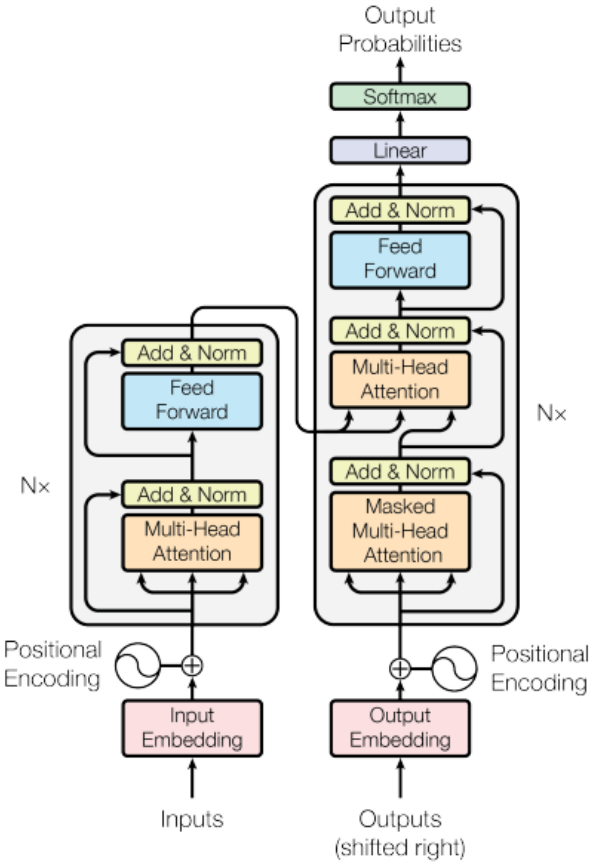


Figure 2.4. *The Transformer - model architecture. Source: [4]*

Sequence-to-Sequence Models leverage the Transformer architecture to map input sequences to output sequences of potentially different lengths. This paradigm is particularly relevant for music applications that involve translating between different representations (such as audio to symbolic notation) or generating musical content based on input conditioning.

T5 (Text-to-Text Transfer Transformer) [5] illustrates this approach by treating every

task as a text-to-text problem using a unified encoder-decoder framework. The model is pre-trained on a large corpus using a denoising objective, where spans of text are masked and the model learns to predict the missing content. This pre-training strategy enables effective transfer to downstream tasks with minimal task-specific modifications. T5's flexibility in handling variable-length input-output mappings and its proven transferability across domains has made it a choice for music generation tasks, where the ability to capture long-range dependencies is essential for maintaining musical coherence and structure.

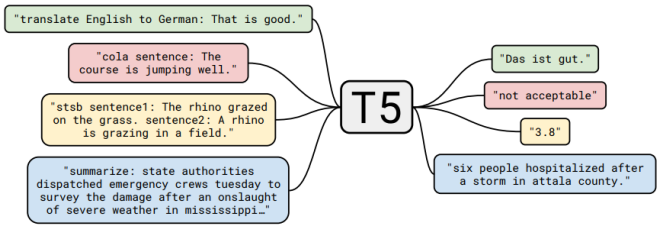


Figure 2.5. *The text-to-text framework used by T5. Every task—translation, question answering, classification is posed as generating target text from input text, enabling a single model and training objective across diverse tasks. Source: [5]*

2.3 Music Representations

Musical information can be represented in many different ways. We consider two widely used music representations: symbolic, and audio representations.

2.3.1 Symbolic Representations

Symbolic music representation encodes musical information as discrete symbols rather than continuous audio signals, providing a structured format that captures the essential elements of musical composition while abstracting away performance-specific details [73]. These representations form the foundation for computational music analysis and generation tasks, as they offer a compact, interpretable, and manipulable format for musical data.

MIDI Format

The Musical Instrument Digital Interface (MIDI) has emerged as the dominant symbolic representation in computational music applications [48]. MIDI encodes music as a sequence of discrete events, each containing specific parameters:

- **Note Events**: Each note is represented by a note-on event (with pitch and velocity) and a corresponding note-off event, where pitch values range from 0-127 (with middle C as 60) [48]

- **Timing Information**: Events are timestamped either in absolute time or as delta times between successive events, typically quantized to musical subdivisions

- **Velocity**: Represents the force or intensity of a note (0-127), approximating dynamics in musical performance

- **Control Messages**: Additional parameters such as program changes (instrument selection), pitch bend, and continuous controllers

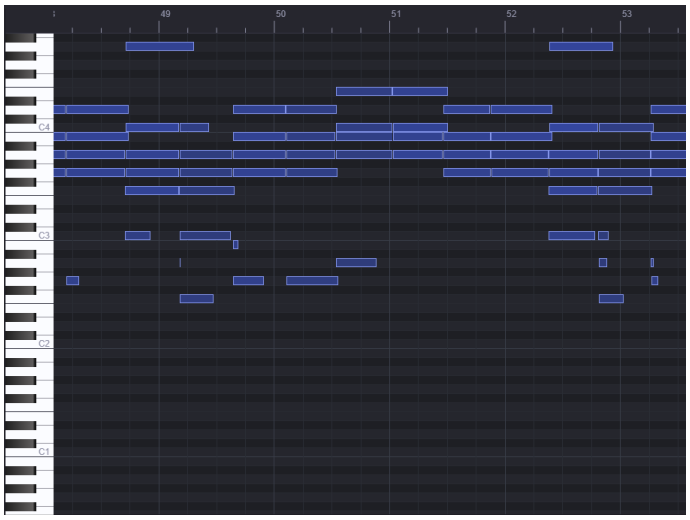


Figure 2.6. *MIDI piano roll view*

MIDI's widespread adoption in Music Information Retrieval (MIR) stems from several key advantages. Its compact representation enables efficient storage and processing of large music collections, as demonstrated in the Million Song Dataset which includes MIDI-aligned features [74]. The symbolic nature of MIDI facilitates various MIR tasks like automatic chord recognition [75], key detection [76], beat tracking [77], transcription [9] and cover generation [1].

Musical Scores and Notation

Beyond MIDI, other symbolic formats capture different aspects of musical information:

- **MusicXML**: Provides a comprehensive representation of Western musical notation, including visual layout information, articulation marks, and score-specific annotations [78]. While richer than MIDI in notational detail, its complexity makes it less suitable for neural sequence modeling.

- **ABC Notation**: A text-based format originally designed for folk music, using ASCII characters to represent pitches, durations, and basic musical structures [79]. Its simplicity and human-readability have made it useful for certain music generation tasks, particularly in folk and traditional music domains.

- **Kern Format**: Developed for musicological analysis, Kern represents polyphonic music with separate spines for each voice, facilitating computational analysis of counterpoint and voice leading [80].

Token-Based Representations

Modern neural approaches to music generation have developed specialized token vocabularies that extend beyond traditional MIDI events. These representations are designed to optimize learning and generation within transformer architectures:

- **MIDI-Like Event Sequences**: Frameworks such as MT3 [10] and Pop2Piano [1] tokenize music as sequences of discrete events. The vocabulary typically includes note pitches (128 discrete values), note-on and note-off events, time shifts (quantized to musical subdivisions), and special tokens such as end-of-sequence (EOS) and padding (PAD). This design mirrors the structure of MIDI while enabling autoregressive prediction within transformer-based models.
- **Compound Tokens**: Some approaches, such as the REMI format in Pop Music Transformer [81] and the MuMIDI representation in PopMAG [82], combine multiple note attributes (e.g., pitch, duration, velocity) into single tokens to reduce sequence length while maintaining musical coherence.

2.3.2 Audio Representations

Audio-based representations capture the continuous waveform or spectral characteristics of a musical signal, encompassing nuances of timbre, dynamics, and performance expression that symbolic representations abstract away. They provide a rich basis for a wide range of Music Information Retrieval (MIR) tasks, including genre classification, source separation, transcription, and style transfer.

Time-Domain Representations

At the most fundamental level, audio signals are represented as time-domain waveforms, typically sampled at a fixed rate:

- **Raw Waveform**: Continuous waveform samples (e.g., 44.1 kHz sampling rate) that preserve the full dynamic and spectral information of the audio signal [83].
- **Amplitude Envelope**: Simplified representation capturing the overall amplitude variation of the waveform, useful for rhythm and energy analysis [84].

Frequency-Domain Representations

Applying the Fourier Transform to audio signals yields frequency-domain representations, highlighting the spectral content:

- **Magnitude Spectrum**: Represents the distribution of energy across frequency bins [84].
- **Phase Spectrum**: Encodes phase information, which can be critical for accurate waveform reconstruction but is often discarded in some MIR tasks [84].

Time-Frequency Representations

To capture the evolution of spectral characteristics over time, time-frequency representations provide a two-dimensional view of audio signals:

- **Short-Time Fourier Transform (STFT)**: Decomposes the signal into overlapping frames, producing a spectrogram that visualizes energy distribution across frequency and time [85].
- **Mel-Spectrogram**: Applies a Mel-scale filter bank to the spectrogram to approximate human auditory perception, widely used in deep learning-based music tasks [83].
- **Constant-Q Transform (CQT)**: Uses logarithmically spaced frequency bins to better match musical pitch intervals, useful for tasks such as pitch tracking and transcription [86].

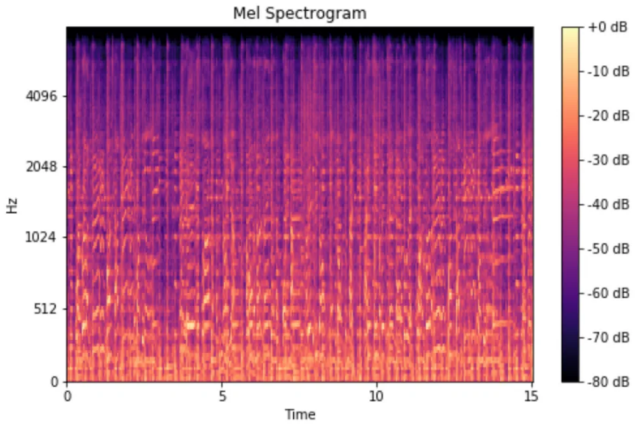


Figure 2.7. *Mel-Spectrogram of a piano excerpt.*

Learned Audio Embeddings

Recent advances leverage deep neural networks to extract high-level audio embeddings, enabling transfer learning and improved performance on downstream tasks:

- **VGGish**: Embeddings learned from large-scale audio data (e.g., YouTube-8M) using a VGG-like CNN, capturing perceptually relevant features for music classification [87].
- **OpenL3**: Trained on paired audio-visual data to create robust representations for audio similarity and classification [88].
- **AudioCLIP**: Embeddings that align audio with text and vision modalities for cross-modal applications [89].

2.4 Music Generation

2.4.1 Overview

Music generation aims to create new musical content through computational models. The first music generated by computer appeared in the late 1950s, shortly after the invention of the first computer. The Illiac Suite is the first score composed by a computer

[90] and was an early example of algorithmic music composition, making use of stochastic models (Markov chains) for generation, as well as rules to filter generated material according to desired properties. This field has advanced rapidly in recent years, driven by deep learning and large-scale data availability, enabling applications in creative AI, music production, and educational tools [73].

2.4.2 Symbolic Music Generation

Symbolic generation focuses on producing structured representations such as MIDI or sheet music, capturing pitch, rhythm, and structural hierarchy. The scope of symbolic music generation encompasses various levels of musical complexity and creative tasks. Accompaniment generation involves creating harmonic and rhythmic support for existing melodies, requiring understanding of chord progressions and musical relationships. Melody generation focuses on creating coherent melodic lines that exhibit musical logic. Whole song generation represents the most comprehensive task, involving the creation of complete musical compositions with structure, development, and coherence across multiple sections.

Specific instrument generation tailors the output to the constraints and capabilities of particular instruments, considering factors such as range, polyphony limitations, and idiomatic playing techniques. Within this category, cover generation emerges as a specialized task that involves adapting existing songs for specific instruments while preserving the recognizable essence of the original composition. This task bridges music transcription, arrangement, and style transfer, requiring models to understand both the source material and the target instrument's characteristics.

Several works have shaped the landscape of symbolic music generation, each introducing novel approaches for the field. DeepBach [91] introduced a steerable graphical model for generating Bach-style chorales using pseudo-Gibbs sampling, demonstrating that non-autoregressive approaches could achieve remarkable stylistic consistency. MuseGAN [92] pioneered the application of Generative Adversarial Networks to multi-track symbolic music generation, addressing the challenges of modeling multiple instruments simultaneously. Music Transformer [81] adapted the Transformer architecture with relative attention mechanisms, enabling the modeling of long-term musical structure and minute-long compositions with compelling coherence. MuseNet [29] scaled up the Transformer approach to generate 4-minute compositions with up to 10 instruments across diverse musical styles from classical to pop. Finally, Transformer-GANs [93] combined Transformer architectures with adversarial training, addressing exposure bias problems and improving long sequence generation quality.

2.4.3 Audio Music Generation

Direct audio generation models synthesize raw waveforms or spectrograms, aiming to capture the richness and expressivity of musical performance. The scope of audio music generation encompasses several distinct subtasks, each addressing different aspects of musical audio creation. Music synthesis involves generating raw audio from symbolic

representations or other high-level musical information. Text-to-audio generation enables the creation of musical content from textual descriptions, while audio continuation focuses on extending existing musical audio with coherent material. Style transfer in the audio domain involves transforming the acoustic characteristics of existing recordings while preserving musical content.

Audio music generation has been shaped by several works that demonstrated the feasibility of generating music directly in the waveform domain. WaveNet [25] pioneered autoregressive modeling of raw audio waveforms using dilated convolutional networks, originally for text-to-speech but demonstrating remarkable results for music generation through sample-by-sample prediction. Jukebox [26] represented a significant leap forward, introducing a hierarchical VQ-VAE approach combined with autoregressive Transformers to generate full songs with vocals, demonstrating controllable generation conditioned on artist, genre, and lyrics. Finally, MusicLM [94] extended text-to-audio generation to music, enabling the creation of musical content from textual descriptions and showcasing the potential for cross-modal music generation.

Chapter 3

Cover Generation of Music Piece

In this chapter, we examine the body of work relevant to cover generation. We begin by exploring Automatic Music Transcription (AMT), including the deep learning approaches that have significantly advanced its accuracy and robustness. Next, we discuss Music Transformation, focusing on methods that adapt and alter musical content. We then turn to Cover Song Identification, which outlines how covers can be recognized and gives us deep insight in their nature. Finally, we review the field of cover generation, highlighting both traditional techniques and recent developments that inform our approach.

3.1 Automatic Music Transcription

In this section we give a brief overview of AMT, present the deep learning advancements, the key datasets and show how these architectures can be used as a backbone for cover generation. Also, how transcription models can be utilized for creating training data for a symbolic music generation model like our own.

3.1.1 Definition and Scope

Automatic Music Transcription (AMT) is the process of converting an acoustic music signal into a symbolic representation, such as musical notation, that details elements like pitch, onset time, duration, and instrument type. This complex task involves several subtasks, including multi-pitch estimation, onset and offset detection, instrument recognition, beat and rhythm tracking, and the interpretation of expressive timing and dynamics [6].

AMT has a wide range of applications, including music education, music information retrieval, music creation, music production, music search, and musicology [95].

In the context of automated cover generation, AMT is important as it provides the note-level data necessary to recreate or reinterpret existing music pieces. By accurately transcribing audio signals into symbolic formats like MIDI, AMT enables systems to analyze and reproduce music with high fidelity, facilitating the creation of training datasets for cover generation.

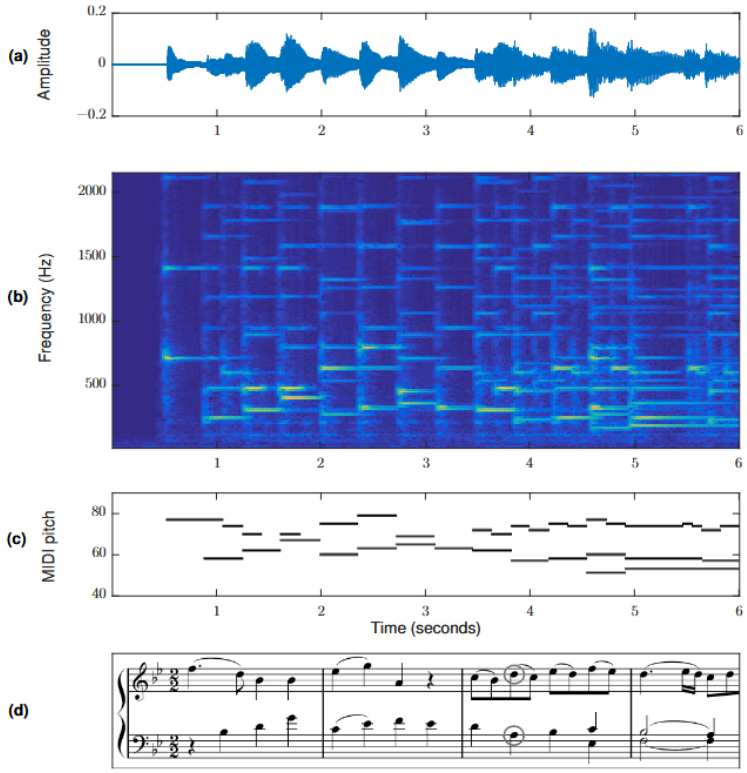


Figure 3.1. *Data represented in an AMT system. (a) Input waveform, (b) Internal time-frequency representation, (c) Output piano-roll representation, (d) Output music score, with notes A and D marked in gray circles. The example corresponds to the first 6 seconds of W. A. Mozart's Piano Sonata No. 13, 3rd movement (taken from the MAPS database).Source [6]*

3.1.2 Deep Learning Advancements

Deep learning has significantly influenced music transcription and music signal processing in recent years, as it has in many pattern recognition tasks. More specifically, Neural Networks (NNs) can learn complex nonlinear mappings between inputs and outputs using optimization techniques like stochastic gradient descent [2]. Although there have been many advancements, the field has moved slower than other fields like image processing [6].

One of the first approaches in this direction was the Marolt's Sonic system [27]. A central component in this approach was the use of time-delay (TD) networks, which resemble convolutional networks in the time direction [2], and were employed to analyze the output of adaptive oscillators, in order to track and group partials in the output of a gammatone filterbank. Because of its competitiveness it appears in comparisons even in recent publications [96].

The first succesfull system was presented by Böck and Schedl [97] which used two spectrograms as input to enable the network to exploit both a high time accuracy (when estimating the note onset position) and a high frequency resolution (when disentangling notes in the lower frequency range). The network is composed of one (or more) Long Short-Term Memory (LSTM) layers [2].

Next, focus is given to long-range dependencies by combining an acoustic front-end

with with a symbolic level module resembling a language model as used in speech processing [98]. The information from the MIDI files is used to train a recurrent network to predict the active notes in the next time frame given the past one. Although the training was done on a large MIDI-based dataset the improvements were small.

The development of AMT models continues with the Onsets and Frames model [7] proposed by the Google Brain team. They use a deep convolutional and recurrent neural network which is trained to jointly predict onsets and frames. One network is used to detect note onsets and its output is used to inform a second network, which focuses on detecting note lengths. Training was carried out using the MAPS dataset [99] which contains audio and corresponding annotations of isolated notes, chords, and complete piano pieces. It is pointed out in [6], this can be interpreted from a probabilistic point of view: note onsets are rare events compared to frame-wise note activity detections – the split into two network branches can thus be interpreted as splitting the representation of a relatively complex joint probability distribution over onsets and frame activity into a probability over onsets and a probability over frame activities, conditioned on the onset distribution. Since the temporal dynamics of onsets and frame activities are quite different, this can lead to improved learning behavior for the entire network when trained jointly.

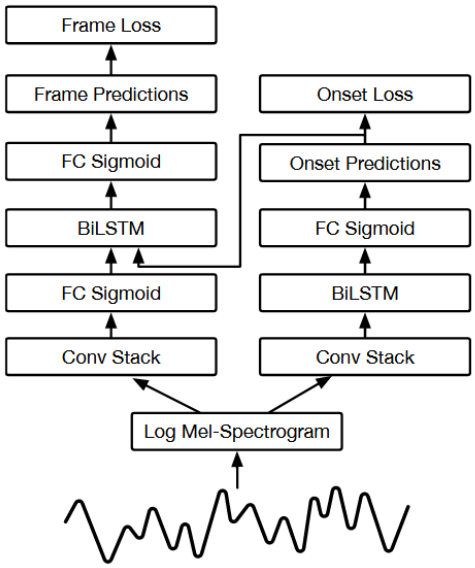


Figure 3.2. *Diagram of Network Architecture. Source [7]*

Figure 3.3 gives us a better understanding of how the model works from the input log-magnitude mel-frequency spectrogram to the output transcription.

Kong [8] proposed a high-resolution piano transcription system by regressing the precise onset and offset times of piano notes and pedals. This approach involves training a neural network to predict continuous values representing the exact timing of note events, rather than relying on discrete frame-wise classifications. During inference, an analytical algorithm calculates these precise times, enhancing the system's ability to capture the nuances of piano performances. Using the MAESTRO dataset [52], the proposed system

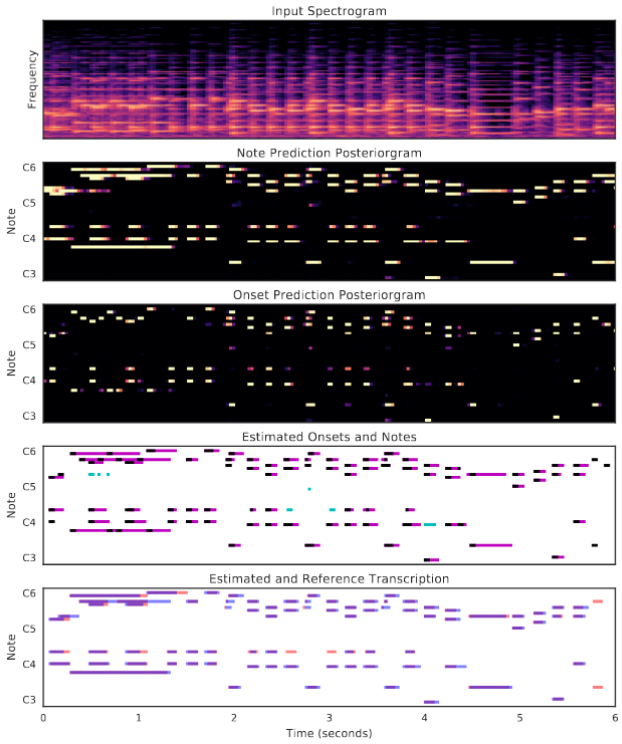


Figure 3.3. *Inference on 6 seconds of MAPS MUS-mz 331 3 ENSTDkCl.wav. Source [7]*

achieves an onset F1 score of 96.72%, surpassing the previous "Onsets and Frames" model's score of 94.80%. Figure 3.4 shows the framework of the proposed high-resolution piano transcription system.

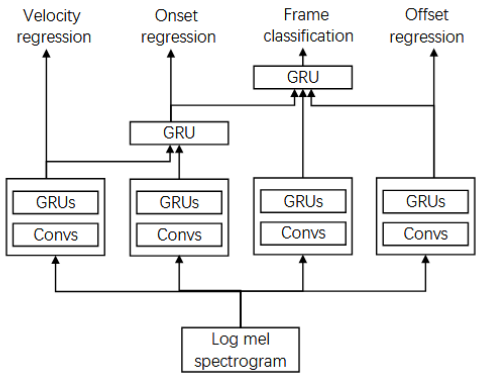


Figure 3.4. *High-resolution piano transcription system by regressing velocities, onsets, offsets and frames. Source [8]*

In contrast with previous approaches, which required domain-specific design of network architectures, input/output representations, and complex decoding schemes Hawthorne, [9] shows equivalent performance can be achieved using a generic encoder-decoder Transformer with standard decoding methods. The model can learn to translate spectrograms to MIDI-like events removing the need for task-specific architectures. This finding suggests that focusing on dataset quality and labeling may be more beneficial for advancing music transcription systems than developing increasingly complex model architectures.

Figure 3.5 visualizes their simple but effective architecture.

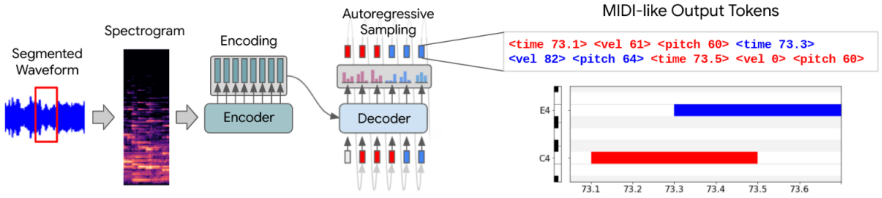


Figure 3.5. *The model is a generic encoder-decoder Transformer architecture where each input position contains a single spectrogram frame and each output position contains an event from our MIDI-like vocabulary. Outputs tokens are autoregressively sampled from the decoder, at each step taking the token with maximum probability. Source [9]*

Building on top of [9] the MT3 (Multi-Task Multitrack Music Transcriptio) model [10] is created. The authors introduce a general-purpose Transformer model capable of transcribing multiple instruments simultaneously. MT3 leverages a sequence-to-sequence framework, enabling the model to jointly transcribe various combinations of musical instruments across multiple datasets. This unified training approach allows MT3 to learn shared representations that enhance transcription performance, particularly for instruments with limited available data. They also introduced and applied a consistent evaluation method using note onset+offset+instrument F1 scores, using a standard instrument taxonomy. Finally, the transcriptions from the model could be used as training data for a symbolic music generation model. Figure 3.6 illustrates the model's ability to transcribe a raw spectrogram into MIDI representations for each instrument.

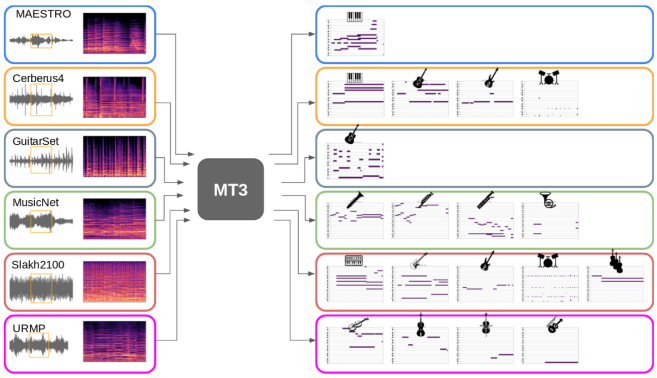


Figure 3.6. *Shown here are real 4-second audio clips, pianorolls reconstructed from the model's tokenized output, and the corresponding instrument labels (additional Slakh2100 instruments omitted due to space). Note that in some cases, multiple notes predicted from a monophonic instrument (such as clarinet or French horn) reflects an ensemble containing multiple players of that instrument. Source [10]*

3.1.3 Datasets

To train these promising Automatic Transcription Models large synchronized instrument-audio datasets were used. We present the most influential datasets in the field of AMT

that have helped the field move forward. Most of them are focused on piano automatic transcription with a little attention given to other instruments.

MAPS

MAPS – standing for MIDI Aligned Piano Sounds – [100] is a database of MIDI-annotated piano recordings. MAPS has been designed for the development and the evaluation of algorithms for single-pitch or multipitch estimation and automatic transcription of music. It is composed by isolated notes, random-pitch chords, usual musical chords and pieces of music. The database provides a large amount of sounds obtained in various recording conditions. MAPS provides recordings with CD quality (16-bit, 44-kHz sampled stereo audio) and the related aligned MIDI files as ground truth. The overall size of the database is about 40GB, i.e. about 65 hours of audio recordings.

SLAKH2100

The Lakh MIDI Dataset [101] is a collection of 176,581 unique MIDI files scraped from publicly-available sources on the Internet, spanning multiple genres. By taking 2100 files from Lakh MIDI and constructing high-quality renderings using high-quality sample-based synthesis the Synthesized Lakh Dataset (Slakh, or Slakh2100) [11] is created. Recordings in Slakh are generated using professional-grade virtual instruments used by countless musicians and composers. Slakh2100, contains 2100 automatically mixed tracks and accompanying MIDI files separated into training (1500 tracks), validation (375 tracks), and testing (225 tracks) subsets, and totals 145 hours of mixtures. Additionally, the technique described can lead to a virtually endless supply of high-quality mixtures and sources. The 2100 files selected all contain at least piano, bass, guitar, and drums, where each of these four instruments plays at least 50 notes. Figure 3.7 gives us a better view.

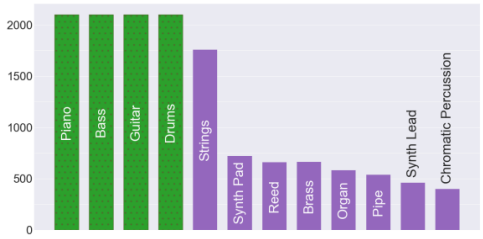


Figure 3.7. *Number of mixtures in Slakh2100 that contain at least one instrument from the following categories. Every mixture has piano, bass, guitar, and drums (the four leftmost bars, shown in green.) Source [11]*

GUITARSET

GuitarSet [53] is a dataset comprising high-quality guitar recordings accompanied by time-aligned annotations. It contains 360 excerpts, generated by six guitarists who each performed 30 lead sheets (songs) in two distinct playing styles: *comping* (rhythmic

accompaniment) and *soloing* (melodic improvisation). These lead sheets represent a diverse combination of five musical genres—Rock, Singer-Songwriter, Bossa Nova, Jazz, and Funk—three harmonic progressions—12-Bar Blues, *Autumn Leaves*, and *Pachelbel's Canon*—and two tempo variations (slow and fast). The dataset's annotations are originally provided in the JAMS format but can be converted to MIDI with standard evaluation libraries. Notably, GuitarSet does not include an official train-test split, leaving it to researchers to devise their own partitioning strategies for model training and evaluation. GuitarSet stands out due to its recording process, which utilises a hexaphonic pickup to capture individual string signals, allowing for detailed transcription and is a first step in helping build models for guitar music transcription.

MUSICNET

MusicNet [12] is a public collection of labels for 330 freely-licensed classical music recordings of a variety of instruments arranged in small chamber ensembles under various studio and microphone conditions. The recordings average 6 minutes in length. The shortest recording in the dataset is 55 seconds and the longest is almost 18 minutes. It also contains MIDI annotations. The annotations were aligned to recordings via dynamic time warping, and were then verified by trained musicians. Figure3.8 gives a detailed summary of the dataset statistics.

MusicNet

Minutes	Labels	Recordings	Error Rate
2,048	1,299,329	330	4.0%

Ensemble	Minutes	Labels
Solo Piano	917	576,471
String Quartet	405	259,702
Accompanied Violin	148	124,886
Piano Quartet	73	60,362
Accompanied Cello	63	37,557
String Sextet	48	33,248
Piano Trio	46	28,873
Piano Quintet	25	27,545
Wind Quintet	43	24,820
Horn Piano Trio	30	18,799
Wind Octet	23	14,635
Clarinet-Cello-Piano Trio	25	13,447
Pairs Clarinet-Horn-Bassoon	24	12,218
Clarinet Quintet	26	11,184
Solo Cello	49	10,876
Accompanied Clarinet	20	10,049
Solo Violin	30	8,837
Violin and Harpsichord	16	7,469
Viola Quintet	15	4,156
Solo Flute	8	2,214

Composer	Minutes	Labels
Beethoven	1,085	736,072
Schubert	253	146,648
Brahms	192	133,109
Mozart	156	99,641
Bach	184	62,782
Dvorak	56	46,261
Cambini	43	24,820
Faure	33	22,349
Ravel	27	21,243
Haydn	15	6,404

Instrument	Minutes	Labels
Piano	1346	794,532
Violin	874	230,484
Viola	621	99,407
Cello	800	99,132
Clarinet	173	24,426
Bassoon	102	14,954
Horn	132	11,468
Oboe	66	8,696
Flute	69	8,310
Harpsichord	16	4,914
String Bass	38	3,006

	Piano	Violin	Cello	Viola	Clarinet	Bassoon	Horn	Oboe	Flute	Bass	Harpsichord
Notes	83	51	51	51	41	36	41	28	37	43	51

Figure 3.8. *Summary statistics of the MusicNet dataset. Source [12]*

URMP

The University of Rochester Multi-Modal Music Performance (URMP) Dataset [102] contains audio-visual recordings and ground-truth annotations for 44 pieces of classical chamber music pieces, ranging from duets to quintets. It has a size of 12.5 GB and the total duration of the dataset is approximately 1 hour and 18 minutes. The dataset includes

both MIDI scores and sheet music (PDF).. The audio recordings consist of high-quality WAV files (48 kHz, 24-bit), available for both individual instruments and full ensemble mixes, following the same track order as the score. The video recordings, encoded in H264 MP4 format (1080p, 29.97 FPS), feature performers arranged horizontally according to the score's track order. Additionally, the dataset provides annotation files, including ground-truth frame-level pitch trajectories and note-level transcriptions in ASCII format.

MAESTROV3

The MAESTRO (MIDI and Audio Edited for Synchronous TRacks and Organization) v3 dataset [52] contains 198.7 hours of piano performances captured via a Disklavier piano equipped with a MIDI capture device which ensures fine alignment (≈ 3ms) between note labels and audio waveforms. The MAESTRO dataset contains mostly classical music and only includes piano performances (no other instruments).

MAESTRO includes a standard train/validation/test split, which ensures that the same composition does not appear in multiple subsets. 962 performances are in the train set, 137 are in the validation set, and 177 are in the test set.

3.2 Music Transformation

In this section, we explore music transformation focusing on two common approaches: music style transfer and music reduction. We begin with a brief overview of music transformation, outlining its significance before delving into an in-depth discussion of each approach.

3.2.1 Definition and Scope

Even though, music transformation has a degree of fuzziness, accounting to the fact that music correctness regarding the result of a transformation is in many ways a subjective judgment, by defining a few terms we bring a little more concreteness to what music transformation is. The definitions here are drawn from [13].

We define **music fragment** as a combination of some number (typically small) of music measures for some voice, along with their corresponding harmonic contexts. A music fragment can be qualified with a temporal extent, i.e., given by beginning and ending whole times, that defines the precise portion of a line and harmony track to which a music transformation could be applied. The term **music feature** refers to some significant aspect of a music fragment, such as the notes in the melody - their pitch values, durations, and offsets, more precisely. It also refers to the key and tempo of the music. And it refers to the chords, their chord types, roots, and inversions, and durations. By **musical cohesion** we mean that for a given music fragment, its music features conform to some music practice that in some way, even subjectively, makes musical sense. In common vernacular, "the music sounds right" or "as the composer intended". Music cohesion can implicitly imply an understood music style or genre, or other criteria that

are considered 'proper' for the music under consideration as source and/or target of a transformation.

A music transformation or **music transform** maps a music fragment into another in such a way that some music features are preserved while others are changed to meet user provided criteria, while at the same time preserving musical cohesion.

Suppose for example, we want 'melodic preservation' in using a transform. The meaning would vary over circumstances. In one interpretation, 'melodic preservation" might mean 'the melodic notes remain the same, identical in pitch, duration, and offset'. That kind of identity transform is generally too restrictive to be meaningful. However, in a key shift in the same modality, the notes pitches are changed, but otherwise the melodies are isomorphic (identical) regarding durations, relative note offsets, pitch, and overall shape. If the modality is changed though and possibly as well as tonal root, we want the melodic shape or contour to be less isomorphic but rather homeomorphically preserved even though the resulting pitches may vary significantly in pitch from the original. So, we are not speaking of a strict isomorphic relationship here. This is illustrated in Figure 3.9 in the transform from C Major to G Melodic Minor.

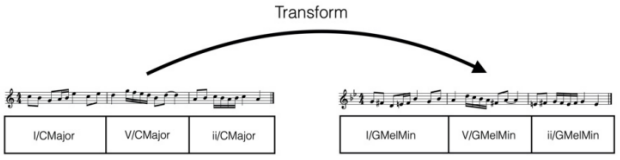


Figure 3.9. *Music Transformation. Source [13]*

We have defined key concepts such as music fragments, music features, and musical cohesion. These ideas help us understand how a music transformation works. Next, we focus on two specific ways to apply these ideas: music style transfer and music reduction. Both methods show how we can change some aspects of the music while keeping other important parts intact.

3.2.2 Music Style Transfer

In the sections that follow, we will first look at **music style transfer**. This method focuses on changing the style of a music fragment by altering elements such as tone, timbre and harmony while keeping the basic melody and rhythm. In this study, we adopt the definition of style from [1]: the unique manner in which each arranger interprets and composes when creating a cover of a song.

Music style transfer by utilizing deep learning techniques has caught attention in recent years. Researchers have approached this task using various models and techniques. We explore synthetic data and one-shot learning, recurrent neural networks with autoregressive models and transformer architectures. Each approach adds a unique view on how style can be understood and transferred between musical pieces.

Groove2Groove [14] presents one-shot style transfer for symbolic music with the use of supervised synthetic data. It focuses on the case of accompaniment styles in popular music and jazz. This approach allows the model to learn from a controlled set of examples,

reducing the need for large amounts of labeled real data. Their model follows the encoder-decoder pattern. It uses two encoders, one for music content and one for style and a decoder that subsequently generates the output. The detailed architecture is shown in Figure 3.10. The work effectively preserves the essential musical structure while adapting stylistic elements and shows promise in this direction. On the other hand, it is limited to symbolic music and relying on synthetic data limits the model's ability to generalize to more varied real-world musical styles.

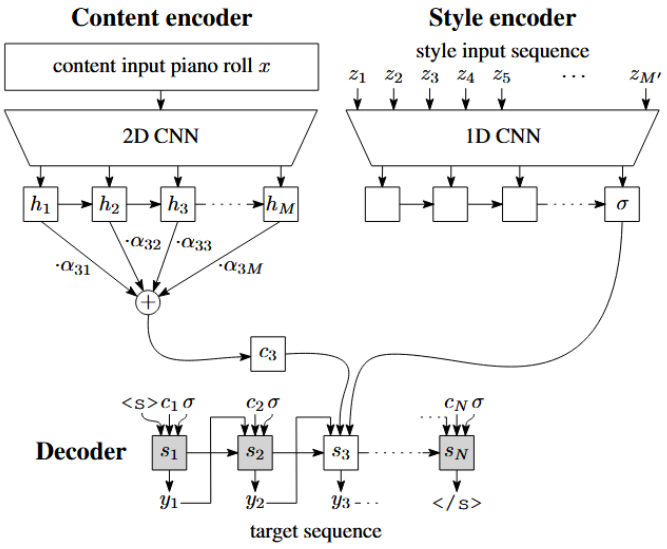


Figure 3.10. *A detailed view of the model architecture. Source [14]*

Another approach is encoding musical style with transformer autoencoders [28]. The authors introduce a model that captures high-level stylistic representations of musical performances using Transformer-based autoencoders. By aggregating encodings across time, a global style embedding is extracted. This allows to independently control the style and melody during the music generation process. To train the model the Maestro dataset [52] and 10000 hours of YouTube audio was used. The model encodes separately performance and melody input and then combines them so that they can be used by the Transformer decoder to generate new sequences. the model is able to effectively capture style and also maintain the input performance structure. The results show that incorporating a global style representation improves performance in terms of log-likelihood and subjective listening tests compared to baseline models.

One more relevant method for music style transfer is found in [103]. This paper focuses on changing the accompaniment of a song while keeping the melody and chord progression the same. The model uses a DeepBach-based recurrent network [91] along with a WaveNet autoregressive model [104] to modify how the accompaniment sounds. The method applies Gibbs sampling, an iterative process that gradually changes the accompaniment to fit a target style, such as Bach chorales or jazz. Unlike the above model, which captures a global style representation, this approach works locally over time, adjusting the harmony at each moment based on the surrounding notes.

A big leap forward was the creation of MuseNet [29] by OpenAI, a deep neural network

capable of creating 4-minute symbolic music compositions with different instruments and styles. It can use up to 10 instruments and blend styles from classical composers like Mozart to modern genres like country and pop. To train the model, they used data from many different sources. ClassicalArchives and BitMidi donated their large collections of MIDI files for this project, and we also found several collections online, including jazz, pop, African, Indian, and Arabic styles. Additionally, they used the MAESTRO dataset [52]. This training approach is similar to that used in GPT-2, focusing on unsupervised learning to predict the next token in a sequence. MuseNet's versatility allows it to combine different styles and instruments, creating unique compositions that maintain musical coherence.

Piano/Guitar cover generation has a lot in common with music style transfer. It tries to keep the essence of the song and at the same time transfer the music performance to the piano/guitar realm. Having discussed the work on music style transfer, we now move on to examine music reduction, an approach that focuses on simplifying music while retaining its essential qualities.

3.2.3 Music Reduction

In music, a reduction is an arrangement or transcription of an existing score or composition in which complexity is lessened to make analysis, performance, or practice easier or clearer; the number of parts may be reduced or rhythm may be simplified, such as through the use of block chords [105]. Another way to reduce a music piece is to perform it by a smaller group of instruments or a single performer, while preserving its musical content. Our review will focus on piano reduction, the process that arranges music for the piano by reducing the original music into the most basic components

One of the first papers in piano reduction [15] , presents a method for automatically simplifying complex musical compositions for piano. The system identifies and retains the most important musical elements from a multi-part score while ensuring that the arrangement remains playable on the piano. To achieve this, the authors introduce a phrase selection algorithm that evaluates the significance of different parts of the composition, selecting the most essential phrases while considering the piano's physical constraints, such as polyphony limits and hand span range.

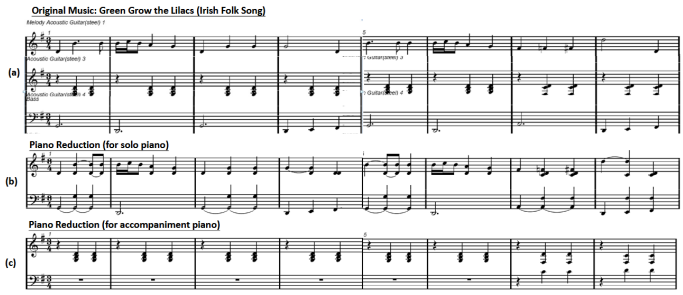


Figure 3.11. *(a) Original music: an excerpt from an Irish folk song "Green Grow the Lilacs" (b) Piano reduction for solo (c) Piano reduction for accompaniment. Source: [15]*

Another important study in piano reduction is [30] which introduces an automatic system that arranges orchestral scores for solo piano by analyzing how human composers

approach piano reductions. Thus, creating an interactive piano arrangement system, which provides real-time feedback to arrangers by detecting and warning them about potential playability issues.

Instead of hand-picked features [31] utilizes a CNN-based supervised learning model to generate piano-playable scores from songs consisting of multiple parts. The study demonstrates that deep learning techniques can be effectively applied to the task of piano reduction.

The link between piano reduction and cover generation lies in the ability to retain melodic and harmonic structures while modifying the instrumental arrangement to fit a specific performance context. In conclusion, we can view cover generation as a form of music reduction in the sense that songs are "reduced" to instrument covers.

3.3 Cover Song Identification

In this section, we analyze CSI (Cover Song Identification), exploring its definition, significance, advancements through deep learning. We also examine how these identification techniques inform the process of cover song generation and can help evaluate generated covers.

3.3.1 Definition and Scope

A cover version is an alternative rendition of a previously recorded song. Given that a cover may differ from the original song in timbre, tempo, structure, key, arrangement, or language of the vocals, automatically identifying cover songs in a given music collection is a rather difficult task [34].

Different versions are characterized by these 10 labels in the literature [33, 34, 35]:

- **Remaster**: Creating a new master for an album or song generally implies some sort of sound enhancement to a previously existing product.
- **Instrumental**: Sometimes, versions without any sung lyrics are released.
- **Mashup**: A mashup is a song or composition created by blending two or more prerecorded songs.
- **Live Performance**: A recorded track from live performances.
- **Acoustic**: The piece is recorded with a different set of acoustical instruments in a more intimate situation.
- **Demo**: A demo is a way for musicians to approximate their ideas on tape or disc and to provide an example of those ideas to record labels, producers, or other artists.
- **Standard**: In jazz music, musicians usually maintain the main melodic and/or harmonic structure but adapt other musical characteristics to their convenience.

- **Medley**: Mostly in live recordings, a band performs a set of songs without stopping between them and linking several themes.

- **Remix**: A remix is a reinterpreted version of a song that can range from minor adjustments in sound and structure to substantial alterations that transform the arrangement.

- **Quotation**: The incorporation of a relatively brief segment of existing music in another work, in a manner akin to quotation in speech or literature.

We can categorize piano/guitar covers with the instrumental label. CSI is an important field that encompasses several applications. In the music industry especially with the rise music platforms like YouTube and Spotify recommendations can be improved by taking into account the different versions of a song. Regarding copyright law, they can protect intellectual property rights by assisting in the detection of unauthorized use of adaptions of original work. In addition, in musicological research it can be useful for music historians to track the evolution and spread of a song across different cultures and time periods.

3.3.2 Deep Learning Advancements

In the early stages of cover song identification (CSI) research, manually designed features were used to achieve acceptable results [32, 36]. However, these traditional approaches have two main drawbacks: their accuracy is limited mainly in recall since handcrafted features struggle to accommodate diverse music styles and instruments, and their high computational cost makes them unsuitable for real-time online applications as the data scale increases.

To overcome these limitations, neural network methods have become the standard, showing promising progress on large datasets. The most common approach involves training a CNN-based model to extract version embeddings by minimizing both classification and contrastive losses. For instance, Yu [37] introduced TPPNet (temporal pyramid pooling) and later CNN-based CQTNet [106] for capturing cover song characteristics. On ther other hand, ByteCover [38] and ByteCover2 [107] achieved state-of-the-art performance using a ResNet-IBN50 backbone with multi-loss training (cross-entropy and triplet loss). Additionally, PiCKINet [39] proposed Pitch Class Blocks to preserve key invariance, and LyraC-Net [108] utilized WideResNet along with combined classification and metric learning to further enhance performance. Furthermore, CoverHunter [40] built on a Conformer-based backbone with an attention-based time pooling module and a coarse-to-fine training scheme reached state-of-the-art performance. Lastly, by building DISCOGS-VI [109], which offers over nine times the number of cliques and over four times the number of versions than existing datasets a baseline NN without extensive model complexities is trained. It achieves comparative results to the other models.

3.3.3 Cover Evaluation

CSI models are well-equipped to understand covers and their relationships to original tracks. They can identify and measure the similarity between a cover and its original, which proves useful in evaluating generated covers. By using a CSI model, we can assess a cover generation model either by comparing the similarity of the track-cover pair or by calculating the distance between them. In short, these models serve as a valuable tool for evaluation.

3.4 Cover Generation

In this section, we explore Cover Generation, first defining its meaning and emphasizing its significance. We review both traditional approaches and recent advancements introduced through deep learning methodologies. Furthermore, we discuss evaluation methods and highlight the associated challenges and limitations.

3.4.1 Definition and Scope

Cover generation is the process of creating a new version of an existing song. Producing a cover typically requires significant time, effort, and advanced musical skills. For instance, you might want to play your favorite song on the piano but cannot find a suitable piano cover available. To address this issue, researchers are developing automatic cover generation models.

3.4.2 Traditional Methods

Historically the methods employed for this creative task can been broadly categorized into manual techniques and computational approaches. Manual techniques primarily involve ear-based transcriptions and music theory-driven arrangements. First, we have the creation of ears-based covers. Skilled musicians relying in their auditory skills create versions of songs. This process requires deep understanding of music theory and aural proficiency. Additionally, musicians employ music theory-driven arrangements, where their knowledge of harmony, counterpoint, and orchestration guides the creation of new musical interpretations. This approach ensures the preservation of the original composition's essence while introducing unique stylistic variations. Although this method enables significant creative freedom, it is typically time-consuming and demands considerable musical expertise.

One of the first computational approaches was made by [16] developing a system for generating string quartet versions of popular songs by combining probabilistic models estimated from a corpus of symbolic classical music with the target audio file of any song.

The system combined audio analysis with score analysis to create cover songs in a specific style for all instruments of a string quartet (2 violins, viola, cello). The audio analysis focuses on rhythms, chord voicings and contrary motions to create a recognizable cover, where as the score analysis focuses on typical note onsets and pitch transitions

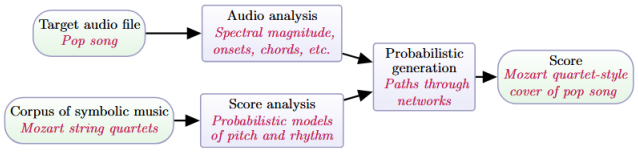


Figure 3.12. *Generating a cover song. Source: [16]*

capturing characteristics of a string quartet. Figure 3.12 gives an overview of the system while 3.13 and 3.14 go into more details about the contribution of each analysis.

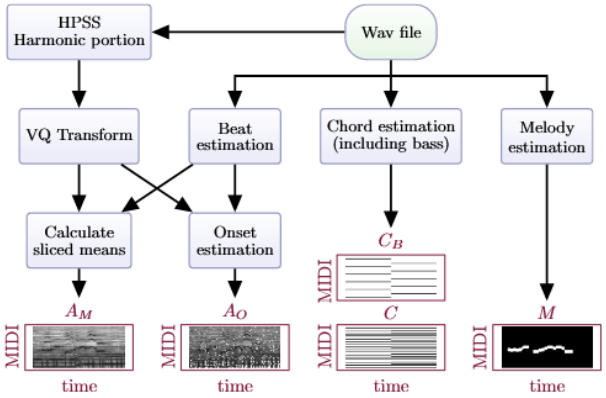


Figure 3.13. *Audio analysis (4 measures shown in examples). Source: [16]*

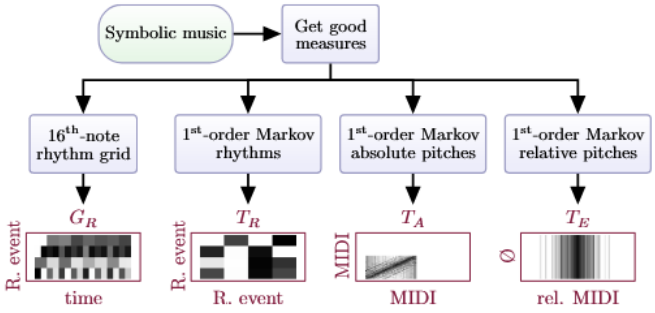


Figure 3.14. *Score analysis (Mozart cello in examples). Source: [16]*

Another study focuses on generating guitar cover songs from polyphonic audio of popular music [17]. Important features are extracted from audio signals such as F0 contour, beats and chords and feeded to an HMM (Hidden Markov Model) producing a tablature score. The system is difficulty aware taking into account the average movement of the index finger of a hand to hold the guitar and the average number of fingers to press the strings. In addition, an interface allowed the guitarist to practice and perform the generated cover. The following is the overview of the system.

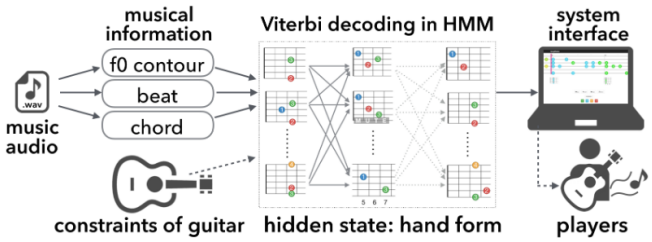


Figure 3.15. *Overview of the Song2Guitar system. Source: [17]*

3.4.3 Deep Learning Advancements

The field of cover generation has experienced a shift from traditional rule-based approaches to modern deep learning methodologies. This transition has been driven by the availability of larger datasets and improvements in neural network architectures, particularly in sequence-to-sequence modeling. While early systems required extensive manual feature engineering and domain-specific rules, recent deep learning approaches have demonstrated the potential for end-to-end learning from data.

The introduction of Pop2Piano by Choi and Lee [1] demonstrated that it is possible to generate piano covers directly from audio input using a purely data-driven approach, without relying on intermediate transcription or explicit musical analysis. Pop2Piano is built on the T5 Transformer architecture [5], adapting the sequence-to-sequence framework commonly used in natural language processing for the music domain. The system treats cover generation as a translation problem, where the input sequence consists of audio spectrogram frames and the output sequence contains symbolic MIDI events. The system architecture is illustrated in Figure 3.16

.

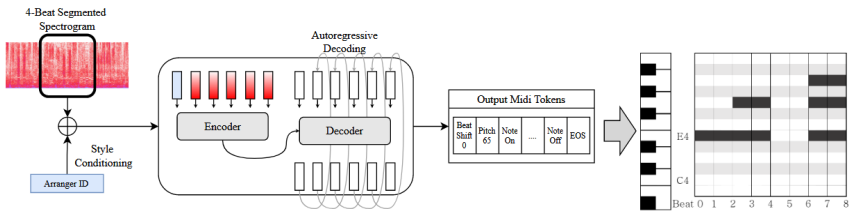


Figure 3.16. *Overview of the Pop2Piano system. Source: [1]*

The architecture includes several key components:

- **Input Processing**: Log mel-spectrograms serve as the audio representation
- **Encoder-Decoder Framework**: A standard Transformer processes input frames and generates output tokens autoregressively
- **Vocabulary Design**: A MIDI-inspired token vocabulary that includes note events, timing information, and control messages
- **Conditioning Mechanism**: Arranger tokens that allow the model to learn style-specific generation patterns

Pop2Piano was trained on a dataset of roughly 5000 pop song and piano cover pairs, representing approximately 307 hours of audio from 21 different arrangers. The dataset construction involved synchronization of audio and MIDI using dynamic time warping techniques [44], followed by quality filtering to ensure alignment accuracy. The detailed preprocessing pipeline is detailed in Figure 3.17.

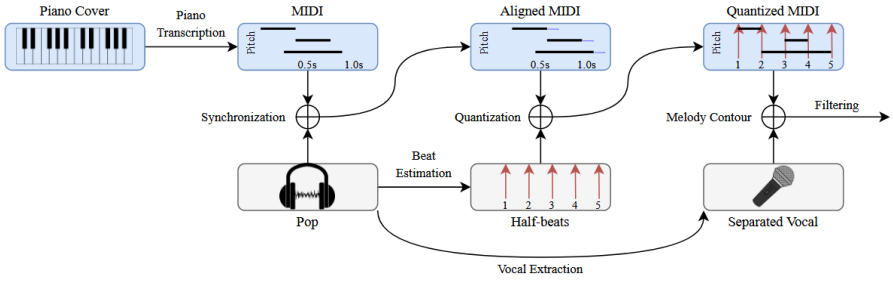


Figure 3.17. *Overview of the Pop2Piano system. Source: [1]*

The results from Pop2Piano demonstrate the viability of deep learning approaches for cover generation. The system achieves reasonable performance on both objective metrics (MCA [46]) and subjective evaluations (user preference study). More importantly, the work establishes a framework that can potentially be extended to other instruments and musical styles.

3.4.4 Evaluation of Generated Covers

The evaluation of automatically generated covers presents challenges, as it requires assessing both technical fidelity and musical quality. Different approaches in the literature have employed varying evaluation methodologies, reflecting the evolving understanding of what constitutes a "good" cover song.

One of the earliest works in automatic cover generation, Song2Quartet [16], relied primarily on informal listening tests without systematic evaluation metrics or user studies. The authors evaluated their string quartet arrangements through subjective listening, focusing on whether the generated covers maintained recognizable melodic and harmonic content from the original songs. While this approach provided basic validation of the system's functionality, it did not offer the methodological framework required for comparative analysis and objective performance assessment.

Song2Guitar [17] introduced a more structured evaluation approach tailored to the specific challenges of guitar arrangement. The authors conducted a user study with one professional guitarist who was asked to practice each generated score for 15 minutes before performing the intro, first verse, and chorus sections. Following the performance, they conducted an interview where the guitarist provided detailed feedback about the system's output, including comments on playability, musical coherence, and arrangement quality.

This evaluation methodology, while limited in scale, addressed important practical considerations for guitar covers, such as technical difficulty and instrument-specific constraints. The use of a professional musician provided insights into the real-world appli-

cability of the generated arrangements, though the single-participant design limited the generalizability of the results.

Pop2Piano [1] established a more comprehensive evaluation framework that combined both objective metrics and subjective assessment. For objective evaluation, the authors employed Melody Chroma Accuracy (MCA) to measure how well the generated piano covers preserved the melodic content of the original songs. This metric provides a quantitative measure of melodic fidelity by comparing the chroma features of the original vocal line with the top melodic line extracted from the generated piano arrangement.

The subjective evaluation involved a user study with 25 non-professional participants. They listened to 10-second clips selected from outside the training set and rated the generated covers on a 1-5 scale based on how naturally the piano arrangement represented the original song. This approach provided insights into listener acceptance and perceived quality of the generated covers.

Chapter 4

Datasets

The field of cover generation lacks readily available datasets, primarily due to the high costs associated with creating high-quality collections. Building such datasets demands significant resources, including skilled musicians and specialized sound equipment. As a result, there are no synchronized song-single instrument cover datasets with time-aligned annotations for any instrument. To address this gap, data are collected from YouTube and synchronization pipelines are implemented. Existing resources, such as the Pop2Piano Dataset [1] and the POP909 Dataset [41], focus on piano arrangements of pop songs. Drawing inspiration from the Pop2Piano Dataset, we developed two new datasets: one for piano and another for guitar cover generation.

In this section, we explore existing datasets in the field of MIR, with a particular focus on datasets tailored to cover generation and on Greek datasets. The existing datasets provide a foundational context and inspiration for the creation of our own datasets, which aim to expand this research area. Specifically, we present two new contributions: the **GreekSong2Piano Dataset**, designed to capture the unique characteristics of Greek music, and the **Pop2Guitar Dataset**, which broadens the scope from piano to guitar cover generation.

4.1 Existing Datasets

4.1.1 Pop2Piano Dataset

The Pop2Piano(PSP) Dataset is a collection of 5989 piano covers from 21 arrangers along with their corresponding pop songs, sourced from YoTube. After synchronizing and filtering the Pop, Piano Cover pairs, a total of 4,989 tracks are left, totaling 307 hours of audio, which forms their training set. In this dataset, each piano cover is unique, though the original songs may appear multiple times. This structure allows the model to learn the distinct stylistic interpretations of piano covers based on the arranger's approach, while also adapting to the acoustic features of the provided audio tracks. This dataset however, is substantial in size, exceeding 250 GB, which presented significant challenges for our current setup. It was used as a training dataset to create the Pop2Piano model [1].

4.1.2 POP909

POP909 [41] is a dataset of 909 Chinese pop songs, specifically designed for music arrangement research. Each song includes MIDI tracks for vocal melody, lead instrument melody, and piano accompaniment, all carefully aligned with their original audio. The dataset also provides detailed information like tempo, beat, key, and chords. To create a usable dataset for cover generation the original songs were downloaded from YouTube and roughly synchronized. It has a size of about 34 GB. Lastly, POP909 served as a benchmark for evaluation for the Pop2Piano model [1].

4.1.3 The Greek Audio Dataset

The Greek Audio Dataset (GAD) [42] is a freely available collection designed to support Music Information Retrieval (MIR) research. It includes metadata, audio features, and lyrics for 1,000 Greek songs but excludes the actual audio files due to copyright limitations. Instead, it provides YouTube links to the tracks for further feature extraction. This dataset manually annotates genre and mood classes, drawing inspiration from the MSD in such a way that it is useful and compatible with state-of-the-art MIR methodologies. The GAD will enable researchers to do genre classification, mood detection, and linguistic analyses, since its features are stored in HDF5 and comma-separated values format, so as to easily interface with data mining platforms.

The GAD concentrates on capturing the peculiarities of Greek music, from traditional to modern genres. The tracks fall into eight categories of genres that capture the rich variety of Greece's musical life: Rembetiko, Laiko, Entexno, and so on. Mood annotation follows the Thayer model, while 16 mood taxonomies have been divided according to dimensions of arousal and valence. Lyrics are included for linguistic analysis, comprising more than 143,000 words and close to 1.4 million characters. Audio features, timbral texture, rhythm, and pitches were extracted through jAudio with other tools such that a track with acoustic properties represents in detail. The creation process emphasized keeping a balance regarding genre representation as well as careful annotations. Genre classification involved listening tests to accurately tag the tracks, while in mood annotation, each annotation had consensus among multiple annotators, achieving an Inter-annotator Agreement of approximately 0.8 in F-measure. The data were extracted from personal collections, live performances, and public platforms like stixoi.info for the lyrics. This is a very holistic approach that will ensure the dataset is relevant not only for Greek music but also can be adapted for wider MIR research goals.

Although currently smaller in size compared to some international datasets, GAD constitutes an important step toward standardized resources for Greek music research, and its format and design allow for future expansions and methodological advancements.

4.1.4 The Greek Music Dataset

The GMD is an important extension of GAD, which is meant for MIR. It comprises 1,400 Greek music tracks. Pre-computed audio, lyrics, and symbolic features have been

provided. It includes manually tagged labels concerning mood and genre, general metadata, a manually selected MIDI file for 500 tracks, and YouTube links for further exploration. Although it does not include audio files due to intellectual property restrictions, it covers a wide range of Greek music from traditional to modern. GMD allows researchers to explore MIR tasks such as genre and mood classification with a focus on Greek musical traditions.

Greek music has its own musical and structural characteristics, reflecting the rich cultural heritage of the country. Traditional genres like ¨Ρεμπέτικο,¨ ¨Λαϊκό,¨ ανδ ¨Έντεχνο¨ are characterized by peculiar features, including uncommon rhythms-for example, 9/8 time-and traditional instruments such as the bouzouki and lyra. GMD focuses on these aspects by providing both audio and symbolic data, thus allowing feature extraction relevant for MIR processes. Manual genre annotations consist of eight categories that capture the rich variety of Greek music styles and provide insight into the evolution of Greek music over time.

It is provided both in HDF5 and CSV for the convenience of a number of data processing tools. There are 454 audio features, 530 linguistic features, and symbolic features from Music21 and jSymbolic. Finally, the addition of lyrics, their bag-of-words models, captures Greek language complexity, linguistic analysis, and mood analysis in the GMD. The dataset represents a robust tool for advancing MIR and exploring Greek music's unique attributes, setting a foundation for future expansions such as contextual metadata integration and symbolic representation enhancements.

4.1.5 Lyra Dataset

The "Lyra" dataset [43], representing an important contribution to computational ethnomusicology in the field of Greek traditional and folk music. It consists of 1,570 pieces, about 80 hours of high-quality audio and video material based on the Greek documentary series *To Alati tis Gis* (*Salt of the Earth*). The collection is augmented with metadata related to instrumentation, genres, geographical origins, and danceability. These data were annotated in detail by volunteers, ensuring both musicological accuracy and rich contextual detail. Unlike previous Greek music datasets such as GAD and GMD, which focused on diverse genres and varied recording quality, Lyra offers a consistent, fine-grained dataset tailored specifically for traditional and folk music research.

The dataset is structured for multiple musicological and computational tasks, supporting analyses of genre classification, geographic patterns, and instrumentation. Metadata: unique identifiers, timestamps, geographic coordinates, and binary labels for "danceability". Some key findings are that "traditional" is dominant at 78%, regional variations in music are influential, and violin, klarino, and laouto are the most central instruments in Greek ensembles. Baseline classification tasks using convolutional neural networks outperformed with promising results: macro F1-scores of 39.9% for genre classification and 34.4% for geographic origin showed that there is a good scope for further refinements and analysis.

The authors have also highlighted that the dataset can enhance the study of MIR and

allow different fields to interrelate on various aspects. In future work, the authors want to increase the metadata categories to include lyrics and dance types and add more pieces from similar series. The Lyra dataset, together with its public availability, positions itself as a robust resource for studying the points of intersection of Greek traditional music, computational tools, and cultural heritage preservation.

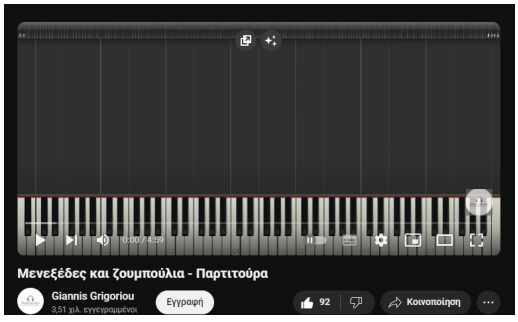
4.2 GreekSong2Piano Dataset

Overview

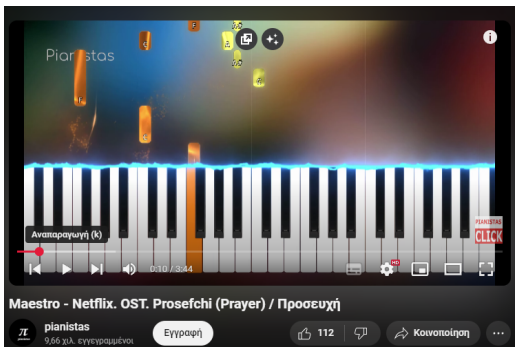
The GreekSong2Piano Dataset consists of 659 Greek songs, their corresponding piano covers in audio and MIDI, manually annotated labels pertaining genre styles of music and lyrics. It consist of roughly 41 hours of music with a size of 42 GB. Its main purpose is to be utilized for piano cover generation, but it can be used in other MIR-related tasks.

Challenges and Methods

Piano Cover In our attempt to create a piano cover generation model that specializes in Greek songs, we decided to construct a dataset consisting of Greek songs and their matching piano covers. The task of finding suitable piano covers was not easy. Also, we wanted the covers to be composed mostly from one arranger, so they are more coherent and have the same style. We searched YouTube specifically for piano covers of widely known Greek songs and found that there are not many available. An invaluable asset was the YouTube channel of Giannis Grigoriou, a Greek music teacher from whom most of our piano covers were collected. In addition, a portion of covers were sourced from the channel of pianistas, a Greek piano teacher and arranger and from the GreekSongsPiano channel. Below we can see two example videos, one from Giannis Grigoriou and one from pianistas.



(a) Piano Cover by Giannis Grigoriou

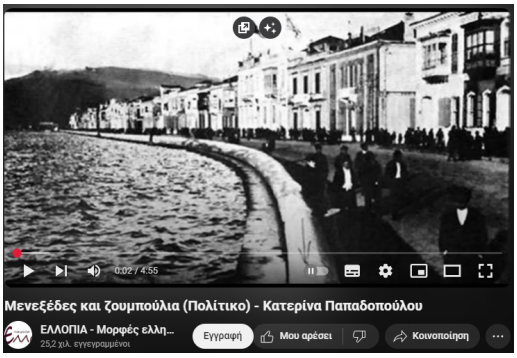


(b) Piano Cover by pianistas

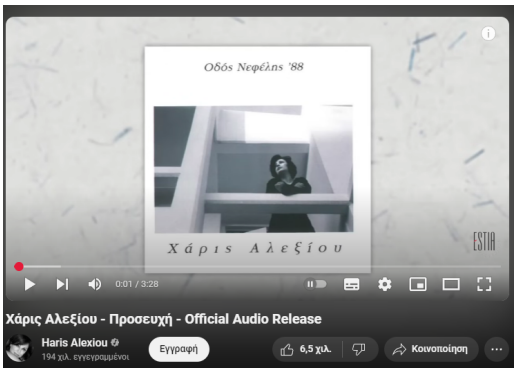
Figure 4.1. *Side-by-side Piano Cover illustrations*

Audio Track For each piano cover, we identified the corresponding original track. Both Giannis Grigoriou and pianistas had a detailed description about the music and the performers. In cases where multiple original tracks were available, we selected the one

with the highest audio quality, highest views, and the most closely matching duration. For example, for the above piano covers we choose these original tracks that ensure high quality, have millions of views and have closely matching duration.



(a) Audio Track with title Violets and Hyacinths (Politiko) by Katerina Papadopoulou



(b) Audio track with title Prayer by Charis Aleksiou

Figure 4.2. *Side-by-side Audio Track illustrations*

Both resources were downloaded from the Internet.

Lyrics For each song, the dataset also provides its corresponding lyrics. These lyrics were sourced from various platforms, including the website stixoi.info [110].

Genre Tags The dataset includes tracks from 8 distinct genres. We adopt the same genre classification as presented in [42]. Below, we provide the necessary explanations to distinguish the unique characteristics of these Greek genres:

- **Ρεμπέτικο (Rembetiko):** 19 tracks. Originating in urban centers with a strong Greek presence, Rembetiko is a type of folk music that emerged with notable influences from the Smyrnaic and Piraeus schools of classical Rembetiko [111]. Characteristic rhythms include "Zeibekiko," "Karsilamas," and "Hasapiko."

- **Λαϊκό (Laiko):** 163 tracks. Evolving from Rembetiko, Laiko represents Greek folk songs from the 1950s and 1960s and continues to evolve today [112]. The transition to Laiko music is marked by the incorporation of European instrument tuning, new rhythms, and harmonic songwriting.

- **Έντεχνο (Entexno):** 147 tracks. A sophisticated form of modern Greek music that blends musical artistry with poetry [113]. It differs from Laiko primarily in its lyrical content and musical style, including instrumentation and arrangement.

- **Μοντέρνο Λαϊκό (Modern Laiko):** 162 tracks. Considered the contemporary evolution of popular music, Modern Laiko incorporates elements of pop and electronic sounds. It is the most commonly performed genre in live Greek music venues, with themes adapted to current societal issues.

- **Rock:** 40 tracks. This category includes Greek Rock as well as 1980s Pop-Rock tracks.
- **Hip Hop / R&B:** 6 tracks. Featuring Greek interpretations of Hip Hop and R&B music styles.
- **Pop:** 74 tracks. Includes Dance-Club music styles and older Greek disco hits.
- **Εναλλακτικό (Enallaktiko):** 63 tracks. Although often equated with "Alternative Rock" [114], this category includes tracks that fuse modern Greek music styles such as Pop Rock and Entexno elements.

Figure 4.3 shows the frequencies of the genres in the dataset, with Μοντέρνο Λαϊκό and the Μοντέρνο Λαϊκό being the dominant ones constituting almost 25 percent of the total. Other genres like Ρεμπέτικο, Hip Hop and Rock have lower percentagies.

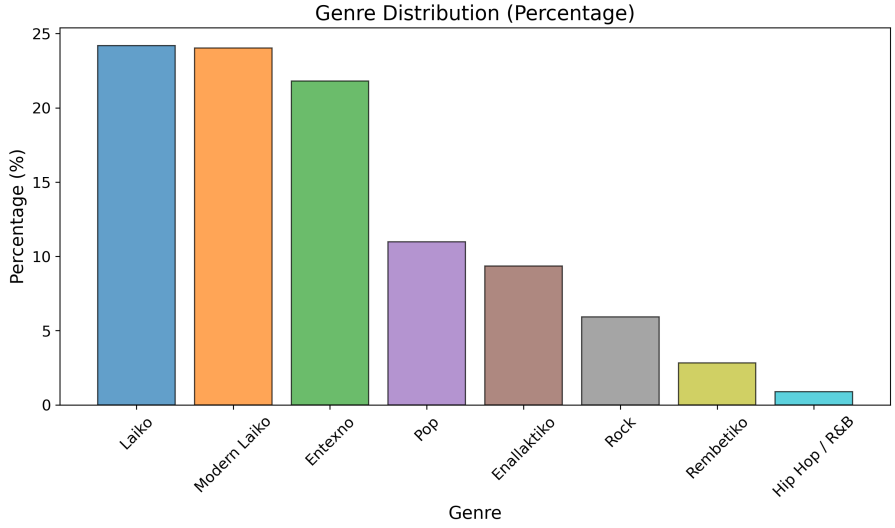


Figure 4.3. *Relative frequencies of the music genres in the dataset*

Transcribed piano covers into MIDI The piano covers in WAV (audio) format were transcribed into MIDI files using a state-of-the-art piano transcription model proposed by Kong et al. [8]. This model utilizes a high-resolution transcription method by directly regressing the onset and offset times of piano notes and pedal events, rather than relying on frame-based predictions.

The transcription process involved feeding the audio recordings into the model, which accurately identified the onset, offset, and pitch of each note, as well as the timing and dynamics of pedal usage. The model's ability to regress exact timing allowed it to produce MIDI files with high temporal precision, closely mirroring the original performances.

The transcription process took approximately 6 hours in total, with each song taking approximately 30 seconds to transcribe.

Using the transcription model, we converted the audio files (WAV format) into piano MIDI (MIDI format). In summary, our dataset consists of: Piano Cover audio, original track audio, piano cover MIDI, lyrics, genre of song.

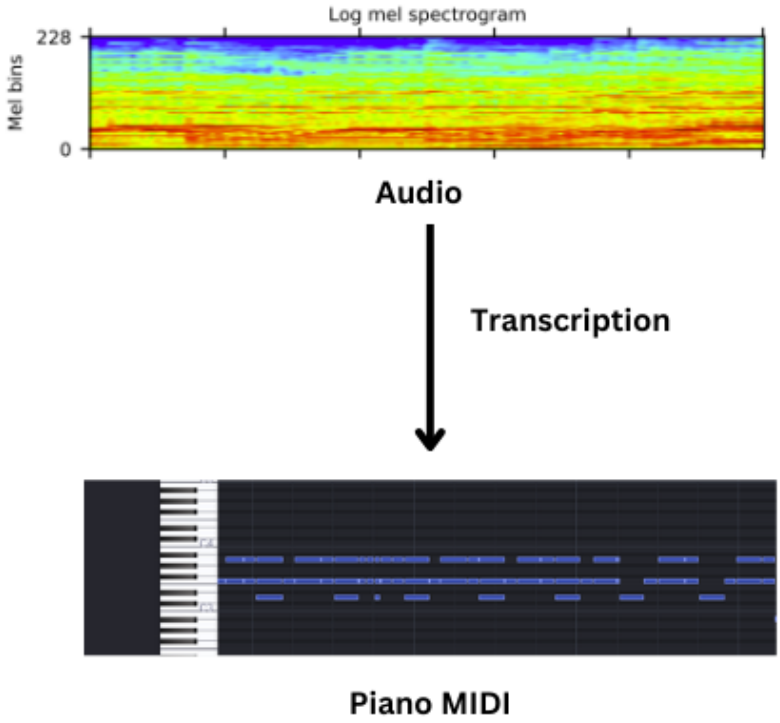


Figure 4.4. *Piano Transcription Adapted from [8]*

Next, the data was synchronized and filtered through the pipeline, which is detailed in Section 5.1.

Data Insights

Metadata In addition to the audio, MIDI, lyrics described in detail above the dataset includes for every one of its tracks a YAML file containing rich information. For both piano cover and the audio track, we collect the uploader, title, YouTube id, genre and duration.

Field	Description
Uploader	The name or identifier of the individual or organization uploading the track.
Title	The title of the music piece as it appears on YouTube or other metadata sources.
YouTube ID	The unique identifier for the track's YouTube video.
Genre	The genre label assigned to the track (e.g., Rempetiko, Laiko, Entexno etc.).
Duration	The length of the track in seconds (SSS format).

We calculated statistical information about the dataset. Figure 4.5 below illustrates the number of songs in each genre. It is evident that Laiko, Modern Laiko, and Entexno dominate the dataset, comprising the largest portions. In contrast, Rembetiko and Hip Hop/R&B are underrepresented. Although we aimed to gather a balanced number of covers across all genres, achieving this for the latter two proved particularly challenging.

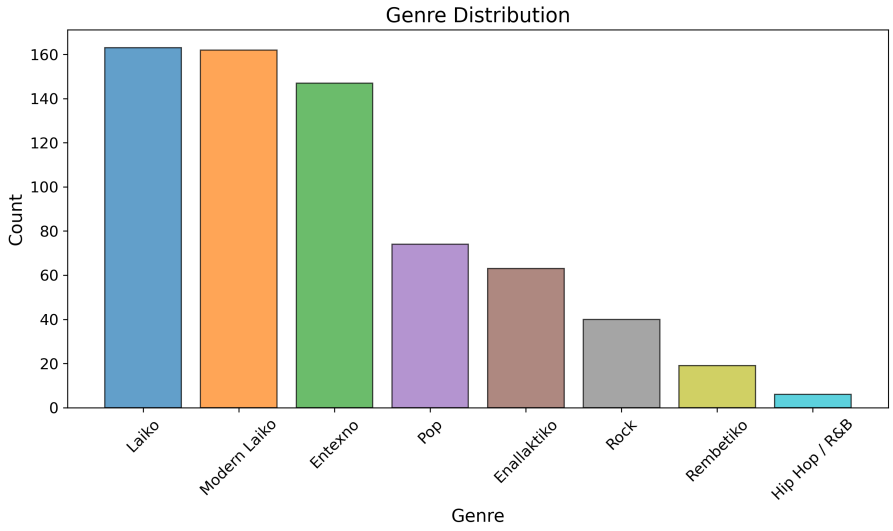


Figure 4.5. *Count of songs for every music genre in the dataset*

Figure 4.6 gives us insight about the number of songs taken from each YouTube channel. We can see that most of our piano covers are sourced from a single channel, indicating a significant reliance on this particular source for our dataset.

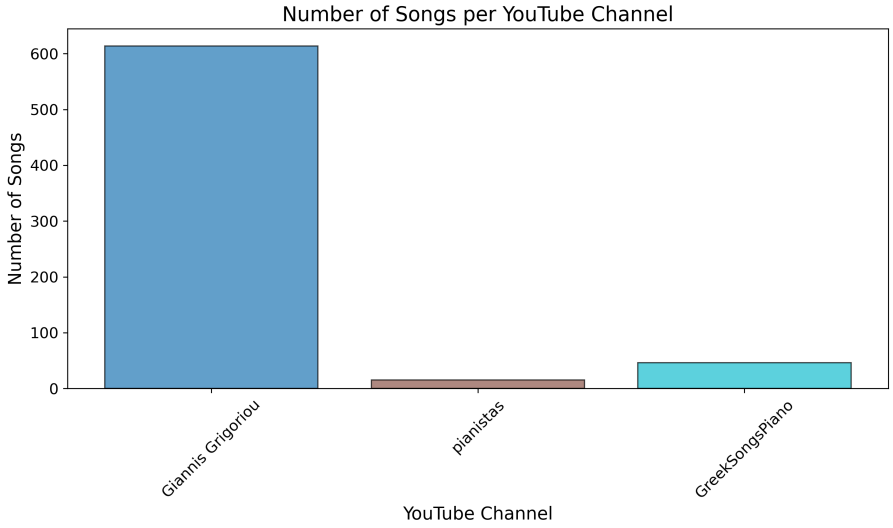


Figure 4.6. *Number of Songs per YouTube Channel*

We present the distributions of song durations and their corresponding piano cover durations in Figures 4.7 and 4.8. Our aim was to select pairs with similar durations to facilitate more accurate synchronization. The distributions confirm that the durations are closely aligned, validating our approach.

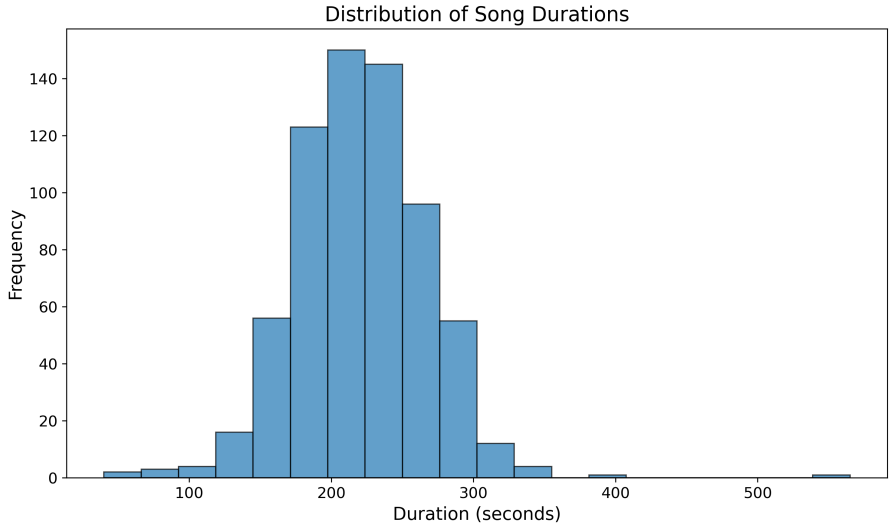


Figure 4.7. *Distribution of Songs Durations*

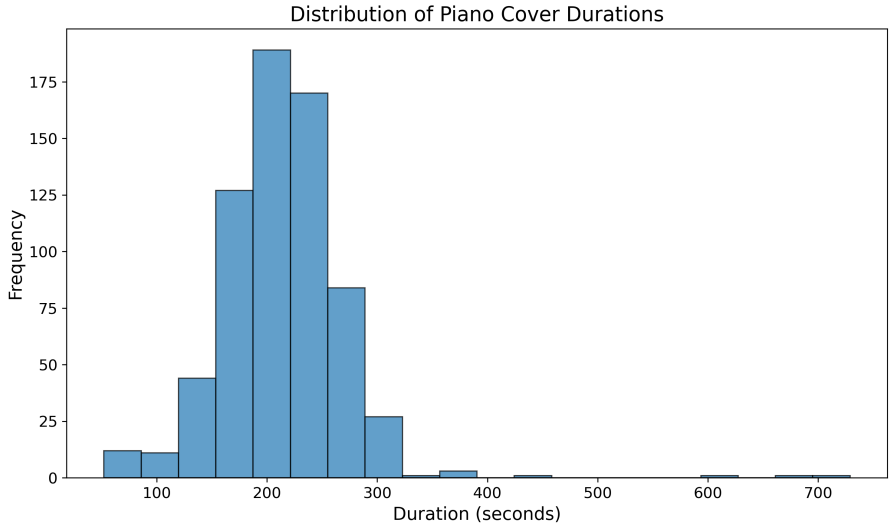


Figure 4.8. *Distribution of Piano Cover Durations*

Focusing on the MIDI files we present the note distribution for every genre. The notes are represented in MIDI note number and gives us a better understanding of the transcribed piano covers and the different distributions according to the specific genre.

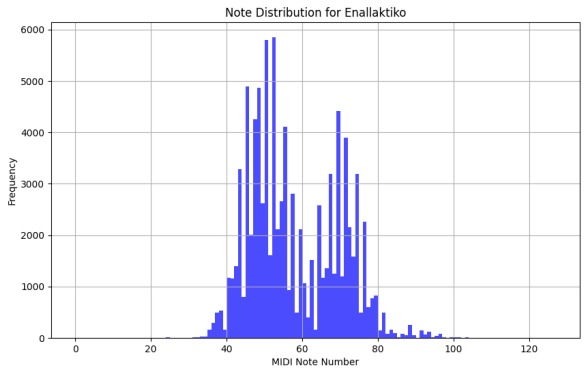


Figure 4.9. *Note Distribution of Enallaktiko*

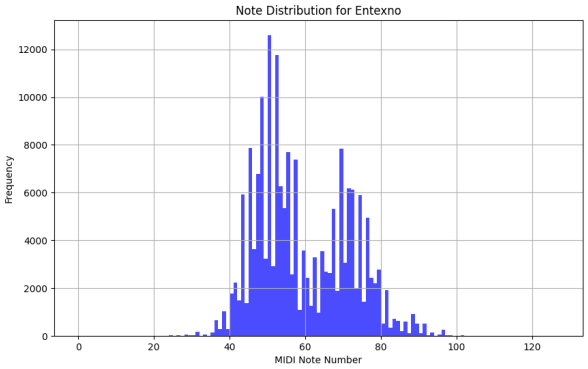


Figure 4.10. *Note Distribution of Entexno*

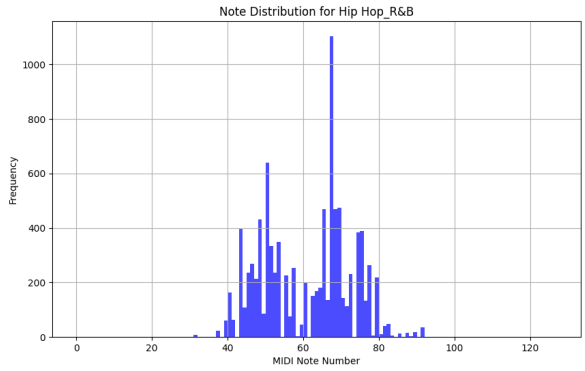


Figure 4.11. *Note Distribution of Hip Hop/ R&B*

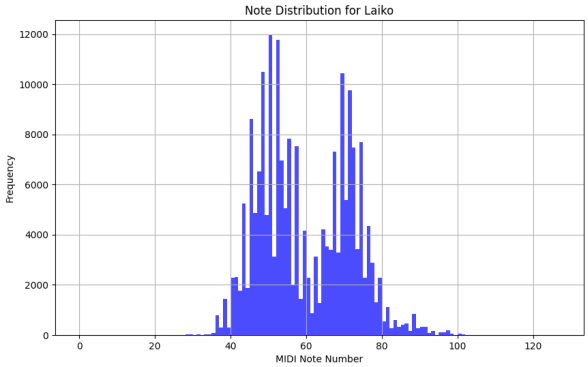


Figure 4.12. *Note Distribution of Laiko*

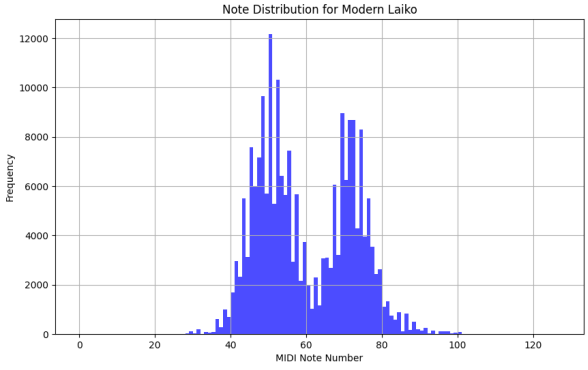


Figure 4.13. *Note Distribution of Modern Laiko*

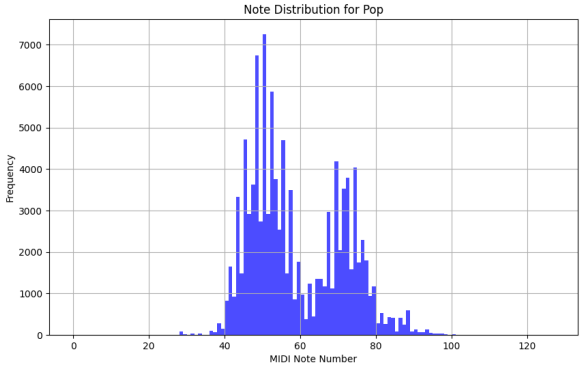


Figure 4.14. *Note Distribution of Pop*

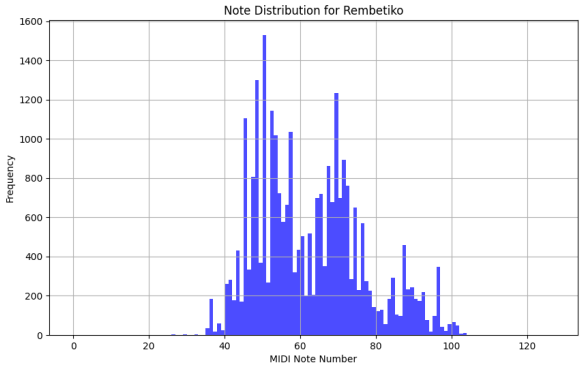


Figure 4.15. *Note Distribution of Rembetiko*

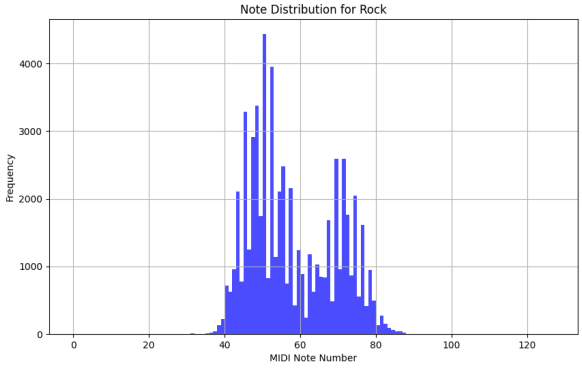


Figure 4.16. *Note Distribution of Rock*

We have extracted and calculated roughly the tempo of the original tracks with the help of essentia [45]. Below we present the tempo distributions of the songs for every genre. We can see that most songs are close to 120 beats per minute (bpm).

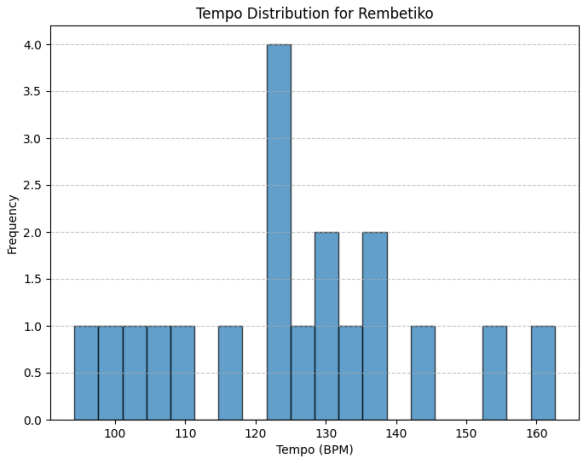


Figure 4.17. *Tempo Distribution of Rembetiko*

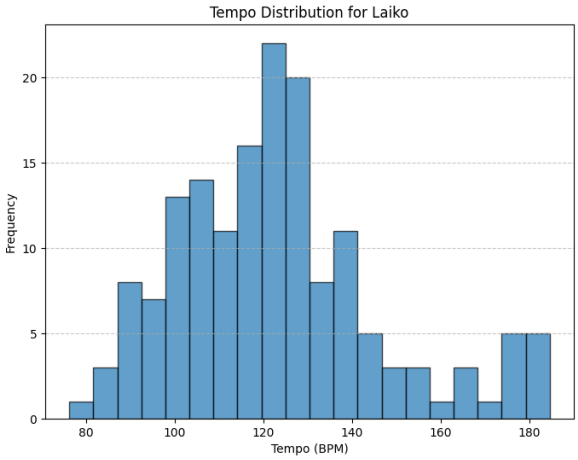


Figure 4.18. *Tempo Distribution of Laiko*

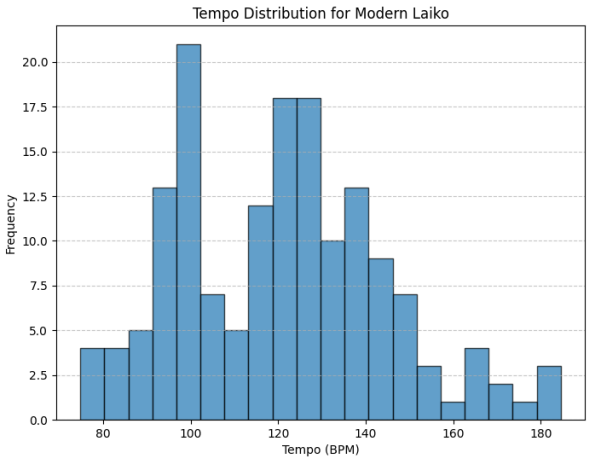


Figure 4.19. *Tempo Distribution of Modern Laiko*

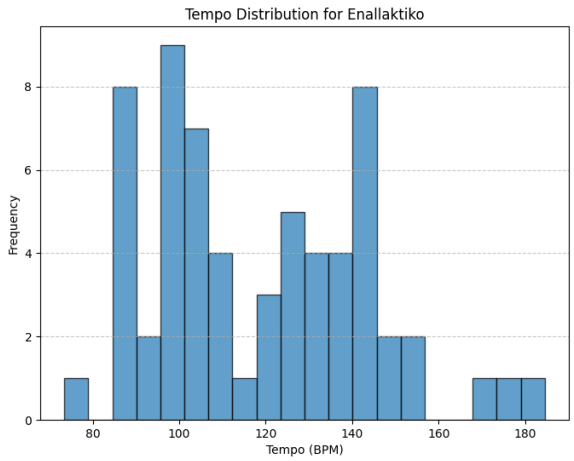


Figure 4.20. *Tempo Distribution of Enallaktiko*

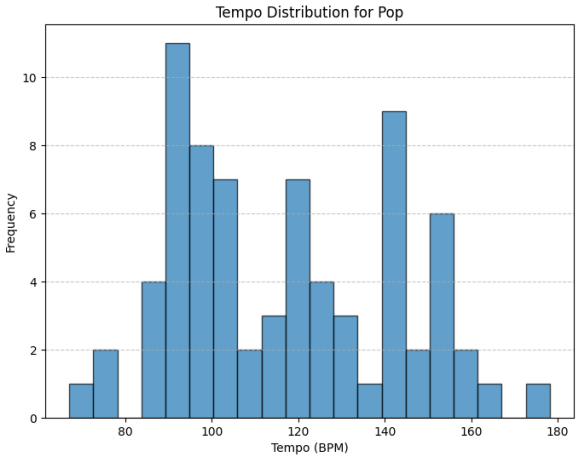


Figure 4.21. *Tempo Distribution of Pop*

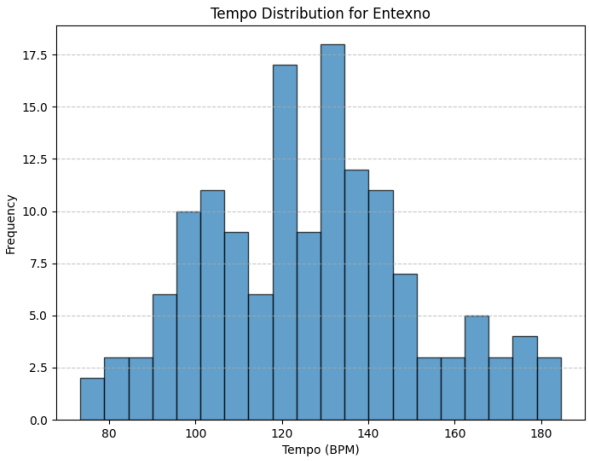


Figure 4.22. *Tempo Distribution of Entexno*

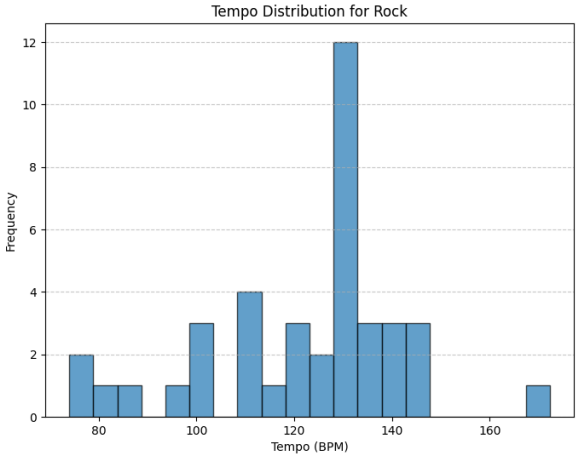


Figure 4.23. *Tempo Distribution of Rock*

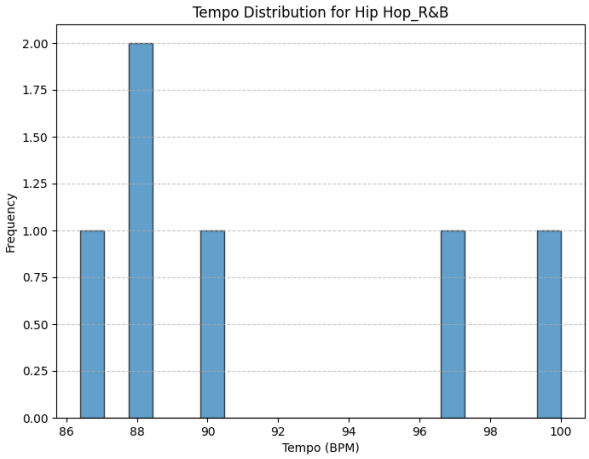


Figure 4.24. *Tempo Distribution of Hip Hop/ R&B*

Lyrics Statistics Our dataset comprises cover-song pairs accompanied by the lyrics of the original songs. A preliminary analysis of the lyrics reveals several interesting statistics:

- **Average Length:** Each song contains approximately **148** words on average.
- **Vocabulary Size:** Across all songs, there are about **10370** unique words.
- **Range of Lengths:** The shortest song comprises of **0** words(small portion of songs are instrumental), while the longest contains up to **509** words.
- **Most Common Words:** If we exclude common stopwords like και, τα, μη, etc., and keep meaningful words we can focus on capturing the emotional and thematic essence of the lyrics.

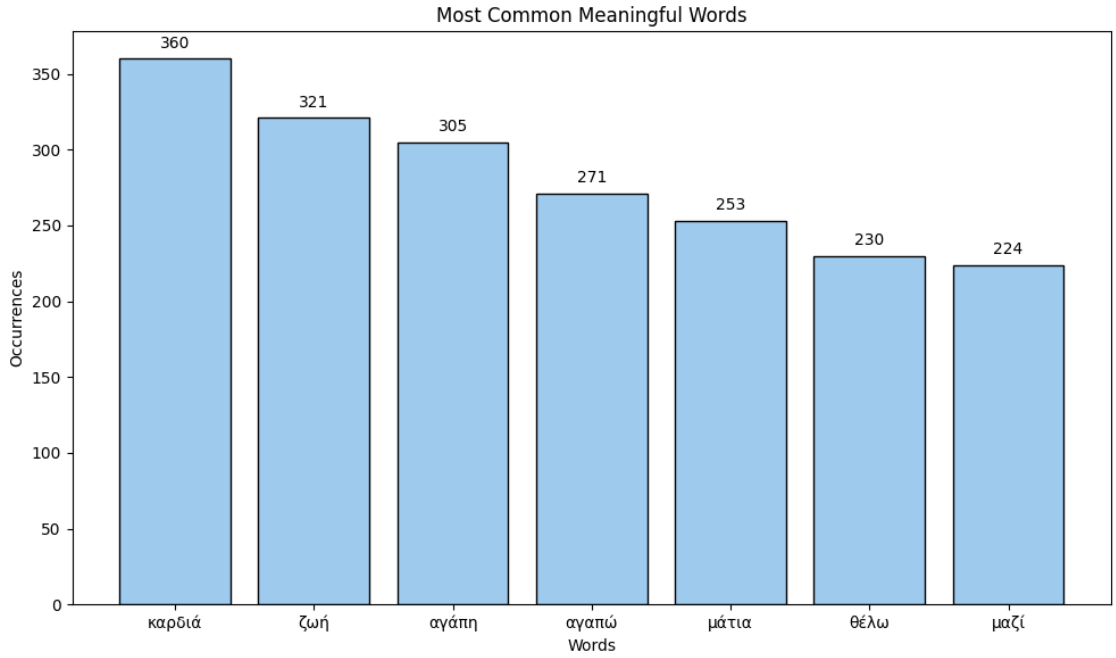


Figure 4.25. *Bar chart showing the frequency of the most common meaningful words: καρδιά (heart) (360), ζωή (life) (321), αγάπη (love) (305), αγαπώ (love) (271), μάτια (eyes) (253), θέλω (want) (230), and μαζί (together) (224) occurrences.*

These words are central to the thematic content of the lyrics, reflecting recurring motifs of love, life, and emotion. It reveals that the corpus is rich in emotion and personal sentiment.

Splitting

The dataset is split into 80-10-10 proportions, resulting in 523 pairs for training, and 64 pairs for validation and 72 for testing. More specifically, we can see the table below.

For all experiments , we use a single train/validation/test split designed to satisfy the following criteria:

- No composition should appear in more than one split.

Split	Performances	Duration (hours)	Size (GB)
Train	523	32.76	34
Validation	66	3.81	3.9
Test	70	4.43	4.1
Total	659	41	42

Table 4.2. *Statistics of the Greek dataset.*

• Train/validation/test should make up roughly 80/10/10 percent of the dataset, respectively. These proportions should be true globally and also within each genre.

• The validation and test splits should contain a variety of pairs. Extremely popular compositions performed should be placed in the training split. For comparison with our results, we recommend using the splits which we have provided. We do not necessarily expect these splits to be suitable for all purposes; future researchers are free to use alternate experimental methodologies.

We can see the detailed split by genre in Table 4.3

Split	Rembetiko	Laiko	Modern Laiko	Entexno	Hip Hop/R&B	Pop	Ennalaktiko	Rock	Total
Train	15	128	128	116	4	59	50	32	523
Validation	2	16	16	14	1	7	6	4	66
Test	2	16	16	15	1	8	7	4	70
Total	19	160	160	145	6	74	63	40	659

Table 4.3. *Number of songs in each split by genre.*

4.3 Pop2Guitar Dataset

Our Pop2Guitar Dataset comprises 40 songs paired with their corresponding guitar covers, created by a diverse group of arrangers. The majority of the songs in the dataset belong to the western pop genre. Because of the scarcity of guitar covers in MIDI we could not collect a lot of data. It consist of 2.52 hours of music with a size of 2 GB. The purpose of this dataset is to explore guitar cover generation via domain adaption.

Challenges and Methods

To create the dataset, we aimed to implement a similar pipeline as used for the Greek and Pop2Piano datasets. Leveraging the abundance of guitar covers available on YouTube, we collected a significant number of covers and attempted to transcribe them into guitar MIDI for our task. However, guitar transcription remains a challenging task due to the limited availability of datasets. While the MT3 model [10] showed promise with its improved performance for low-resource instruments like the guitar, it struggled to generalize effectively, often misidentifying the guitar as a variety of other instruments.

As we can see above, the solo guitar recording is transcribed incorrectly(every different colour represents a different instrument). Further evidence about the poor generalization skills are reported here [115]. So instead of using transcribed guitar recordings we collected the guitar covers in MIDI format from Musescore [24], valiable resource for music

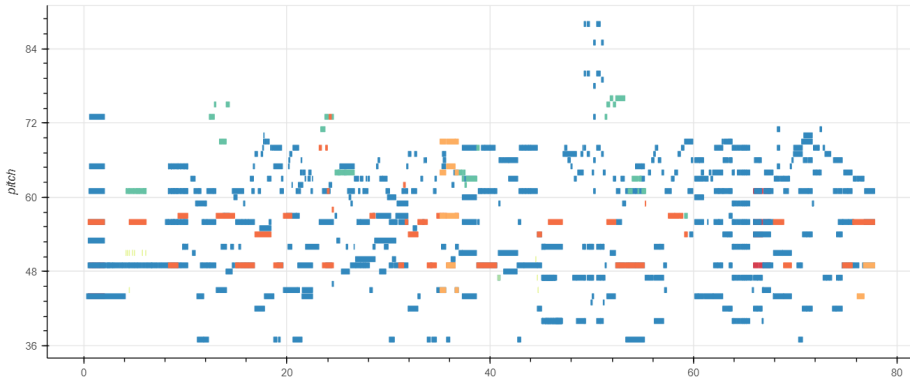


Figure 4.26. *Transcribed example with MT3*

scores. We searched for guitar covers in MIDI format with the keywords: guitar cover and filtered so that only the solo guitar covers would remain. From the 139 pieces, a part of them required payment and another part was missing the MIDI files or did not match with the original song. So in total 40 MIDI files were sourced. The corresponding original tracks were collected from YouTube exactly like the ones for the Greek Dataset.

The data was then synchronized and processed through our pipeline, as described in Section 5.1. This resulted in a set of synchronized song and cover pairs with a total size of approximately 2 GB.

Data Insights

Metadata In addition to the audio and MIDI described in detail above the dataset includes for every one of its tracks a YAML file containing rich information. For both piano cover and the audio track, we collect the uploader, title, YouTube id and duration.

Field	Description
Uploader	The name or identifier of the individual or organization uploading the track.
Title	The title of the music piece as it appears on YouTube or other metadata sources.
YouTube ID	The unique identifier for the track's YouTube video.
Duration	The length of the track in seconds (SSS format).

Splitting

Given the limited size of our Pop2Guitar dataset (40 pairs), we employed 5-fold cross-validation to ensure robust evaluation of our guitar cover generation models. The dataset was randomly partitioned into 5 folds, with each fold serving as the test set once while the remaining 4 folds were used for training. This approach maximizes the use of our limited data while providing more reliable performance estimates than a single train-test split.

Chapter 5

Methodology

In this chapter, we present the methods we followed, regarding our preprocessing pipeline, the model used, our training for our cover generation task.

5.1 Preprocessing Pipeline

Our preprocessing pipeline is split into three parts: Synchronization, Beat Extraction and Filtering of low quality samples. It follows the recipe used in [1].

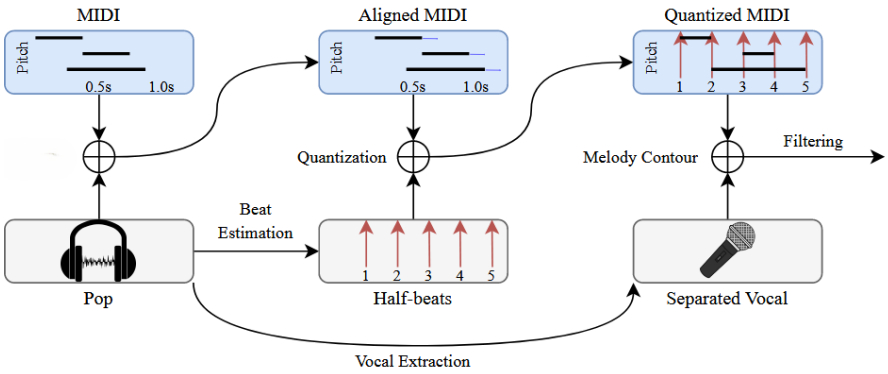


Figure 5.1. *Preprocessing Pipeline. Adopted from [1]*

5.1.1 Synchronization

To create a usable dataset, synchronizing the songs with their covers is a critical step. We utilize *SynctoolBox* [44], a Python package designed for efficient, robust, and precise music synchronization. The process begins by normalizing the audio followed by dynamic time warping (DTW) to align the audio and MIDI accurately, regardless of key or tempo differences. After alignment, we adjust the note timings of MIDI to match the aligned audio with linear interpolation. Lastly, we produce pitch-shifted audio and warped MIDI.

5.1.2 Beat Extraction

The next step involves quantizing the note timings into 8th-note units. Using *Essentia* [45], beats are extracted from audio recordings, serving as the temporal framework for

quantization. The onset and offset of each MIDI note are then snapped to the nearest 8th-note beat. To ensure that no two quantized notes have identical onsets and offsets, any offsets coinciding with their onsets are adjusted to the next beat. This approach reduces data entropy by transforming the timing representation from continuous time (in seconds) to a structured, quantized format (in beats), making the data more manageable and consistent for further processing.

5.1.3 Filtering

The last step is filtering the low quality samples of the data pairs. For example, some covers might have a difference in musical progress or different keys or the synchronization might fail and create unsuitable pairs. This entails both automatic and manual handling. We calculate the Melody Chroma Accuracy (MCA) [46] and discard pairs with 0.10 or less. Melody Chroma Accuracy (MCA) evaluates the similarity between two monophonic melody sequences. The melody line plays a crucial role in deciding whether a song cover resembles the original song. We compute the MCA between the vocals extracted by Spleeter [47] from the audio, and the top melodic line extracted from the cover MIDI using the skyline algorithm. To get the melody contours of music, the f0 sequence of the vocal is calculated using Librosa [83] and pYIN [116]. The analysis is performed with a sample rate of 44,100 Hz and a hop length of 1,024 samples. Additionally, pairs with an audio length difference of 40% or more are rejected. Lastly, we manually verify each extracted pair to ensure alignment and correctness.

5.2 Model

Music cover generation fundamentally involves a sequence-to-sequence task, where a series of audio frames serves as the input, and the output is a sequence of symbolic tokens representing the notes arranged for the cover instrument. So, the use of a generic encoder decoder Transformer architecture where each input position contains a single spectrogram frame and each output position contains an event from a MIDI-like vocabulary is ideal. Figure 5.2 provides an overview of the model along with the input and output configuration.

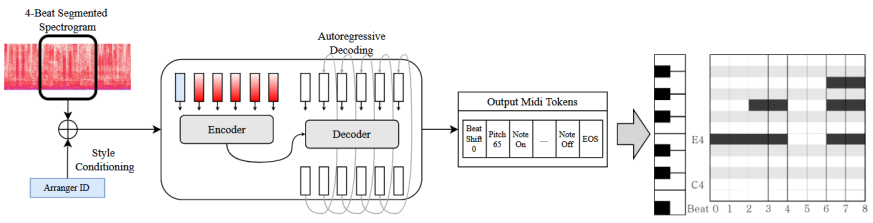


Figure 5.2. *Model Architecture. Source [1]*

5.2.1 Architecture

The architecture is the same as the Pop2Piano model [1]. It is based on the T5 architecture [5], an encoder-decoder Transformer model which closely follows the original form in [4]. The relative positional embeddings are used like in [5] instead of the absolute positional embeddings used in MT3 [10]. The model employs standard Transformer self-attention blocks in both its encoder and decoder. For generating output sequences, greedy autoregressive decoding is used: the input sequence is processed, and the output token with the highest predicted probability is appended iteratively until an end-of-sequence (EOS) token is generated. In our setup, we use the T5 “small” model, which comprises approximately 60 million parameters.

We discussed using a larger model like T5 "large" or a varient of GPT [117] but evidence from [10] and [9] showed that for Automatic Music Transcription (AMT) increasing model size tended to exacerbate overfitting and a comparative small model was sufficient. Since Automatic Music Transcription (AMT) and Automatic Cover Generation are closely related tasks, we determined that the T5 "small" model, was an appropriate choice.

5.2.2 Inputs and Outputs

As shown in Figure 5.2 the model uses log Mel spectrograms as inputs. Also, the arranger token, indicating who arranged the target cover is embedded and appended before the first frame of the log Mel spectrogram. At each step, the model produces a softmax distribution over a discrete vocabulary of events, as outlined below. This vocabulary is heavily inspired by the messages originally defined in the MIDI specification [48] and was first applied in AMT [9] and [10]. The vocabulary consists of the following token types:

Note Pitch [128 values] Indicates a pitch event for one of the MIDI pitches. However, only the 88 pitches corresponding to piano keys are actually used.

Note On/Off [2 values] Determines whether previous Note Pitch events are interpreted as note-on or note-off.

Beat Shift [100 values] Indicates the relative time shift within the segment quantized into 8th-note beats. It applies to all subsequent note-related events until the next Beat Shift event. The vocabulary includes Beat Shifts up to 50 beats, but because time resets for each segment, in practice, only about 10 events of this type are used.

EOS, PAD [2 values] Indicates the end of the sequence and the padding of the sequence.

5.2.3 Sequence Length

Transformers can process all tokens in a sequence at every layer, making them particularly well-suited for transcription tasks that demand precise details about pitch and timing for each event. However, this attention mechanism has a space complexity of

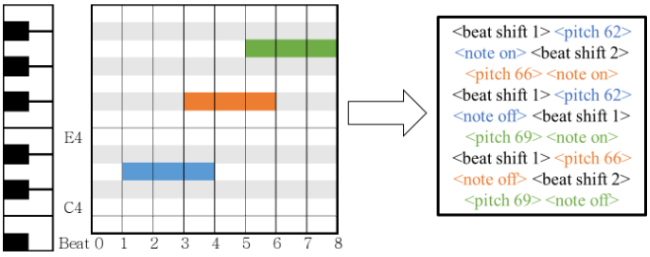


Figure 5.3. *Piano Tokenization. Source [1]*

$O(n^2)$ with respect to the sequence length n. As a result, most audio sequences used for transcription exceed memory limitations. To address this issue, the audio sequence and its corresponding symbolic representation are divided into smaller segments during both training and inference.

During training The following procedure is applied to each sequence in a batch:

1. A random audio segment is selected from the full sequence as the model input. The length of the segment is 4 beats.
2. The corresponding symbolic segment is selected as the training target. Since notes may begin in one segment and end in another, the model is trained to predict note-off events even in cases where the note-on event is not observed within the segment.
3. A spectrogram is computed for the selected audio segment. The sampling rate is 22050, the window size is 4096, and the hop size is 1024.Then, the symbolic sequence is mapped into the defined vocabulary like in Figure 5.3.
4. The spectrogram input and the one-hot-encoded MIDI-like events are provided as a training example to the Transformer architecture.

During inference The following procedure is applied:

1. The audio sequence is divided into non-overlapping segments, using the maximum input length wherever possible, and spectrograms are computed for each segment.
2. Each segment is processed sequentially by providing its spectrogram as input to the Transformer model. The model decodes the sequence by greedily selecting the most probable token at each step based on its output until an EOS token is generated.
3. The decoded events from all segments are concatenated into a single sequence (except for the EOS token).
4. The relative beats of the generated tokens are then mapped to absolute time using the beat timing extracted from the original song.
5. This information is subsequently used to convert the sequence into a standard MIDI file.

5.3 Training Strategies for Greek Song-to-Piano Cover Generation Models

In this section, we present three training strategies: from scratch training, partial fine-tuning, and full fine-tuning. The GreekSong2Piano dataset is used for all training approaches which is described in 4.2.

5.3.1 Training from scratch

To develop a piano cover generation model for Greek songs we first tried to train the model from scratch. We use the same model architecture and tokenizer as [1]. The problem with this approach is that it requires a large dataset to achieve good results and our dataset was not large enough. Our dataset is under 1000 songs and more than 5 times smaller than the dataset used to train the Pop2Piano model.

Figure 5.4. *Training from Scratch*

5.3.2 Transfer Learning

To deal with the low-resource task we apply Transfer Learning. We start from the Pop2Piano model [1] which is trained on approximately 5000 songs-piano pairs and fine-tune its parameters on our specific dataset. This approach leverages the knowledge the model has already learned on generating plausible piano covers for the pop genre and tries to transfer it to Greek songs and their discrete genres.

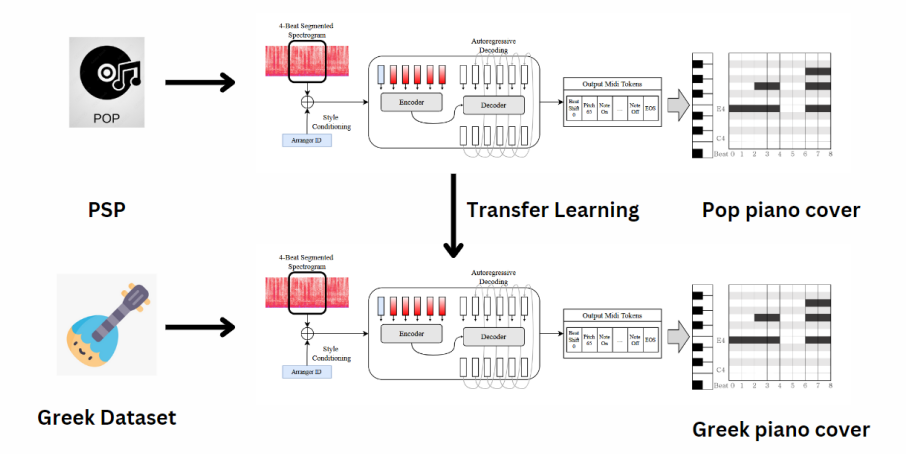


Figure 5.5. *Transfer Learning from Pop2Piano Model*

Partial Fine-tuning

A common approach to transfer learning involves freezing most of the model's layers and training only the final layers. This strategy leverages the fact that the early layers typically learn generic features, such as abstract musical representations, while the later layers capture task-specific features. By keeping the early layers frozen and unfreezing the final layers, as detailed in the following figure 5.6, the model can efficiently adapt to the target genres, such as Entexno. The Pop2Piano model is based on the T5 "small" architecture. It consists of 6 encoder, 6 decoder layers and a language modeling head.

Table 5.1. *T5 Small Model Specifications*

Parameters	**# Layers**	d_{model}	d_{ff}	d_{kv}	**# Heads**
60M	6	512	2048	64	8

We freeze all encoder layers and experiment with fine-tuning parts or all of the decoding layers and the language modeling head.

Full Fine-tuning

Instead of freezing most of the model's layers and training on the final layer, we explore training the whole model from a pre-trained checkpoint. This requires more time and GPU resources, but might give better performance. Full fine-tuning adjusts all layers of the model, allowing it to adapt to the target task. Additionally, it mitigates pre-trained biases more effectively and enhances generalization to new data[49].

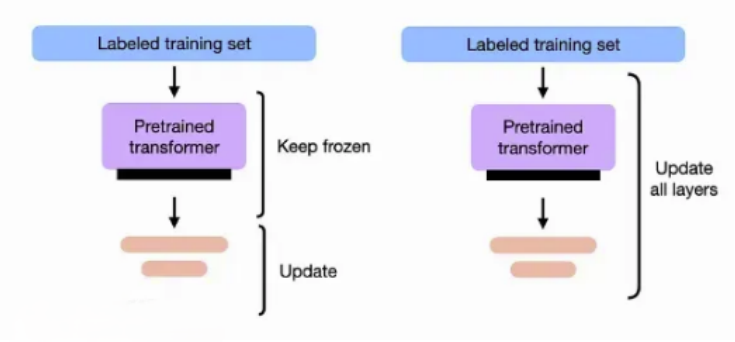


Figure 5.6. *Partial vs Full Fine-tuning*

The combination of transfer learning and full fine-tuning proved to be the most effective strategy for Greek song-to-piano cover generation, solving the challenges of low-resource datasets and genre-specific nuances.

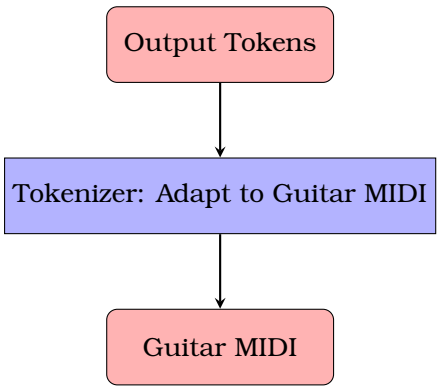
5.4 Training Strategies for Song-to-Guitar Cover Generation Models

In this section, we outline the training strategies employed to develop guitar cover generation models for our Pop2Guitar dataset. As with piano cover generation models, the challenges of limited data and genre-specific nuances make the creation of a from scratch model hard to create. To deal with these difficulties we take advantage of Transfer Learning and Domain Adaptation. Piano cover generation for pop music has achieved plausible results with the use of the Pop2Piano dataset (PSP) but comparable datasets are not yet available for other instruments. We propose the use of a piano cover generation model to train a new guitar cover generation model.

Given the limited size of our Pop2Guitar dataset (40 pairs), we employed 5-fold cross-validation to obtain robust performance estimates. This approach ensures that every song-cover pair serves as both training and test data across different folds, maximizing the utility of our limited dataset while providing statistically reliable evaluation metrics with 95% confidence intervals.

5.4.1 Tokenization

To create the guitar cover generation model we have to make small tweaks to the tokenizer. The input stays the same (spectogram) but the output has to change from piano MIDI to guitar MIDI. A real guitar typically covers a range of MIDI notes corresponding to the physical range of its strings and frets. Here's the breakdown: The lowest note is the open low E string (E2), which corresponds to MIDI note 40. The highest note is typically the 24th fret of the high E string, which corresponds to E6 (MIDI note 88). However, guitars with fewer frets (e.g., 21 or 22 frets) will have a slightly lower upper limit, typically around D6 (MIDI note 86) or Eb6 (MIDI note 87). So, the standard range of a standard guitar is from 40 (E2) to 88 (E6) [48]. Our pre-trained model can output all 128 notes but only 48 are actually used for the guitar generation. We adapt the MIDI rendering process by changing the output instrument from a piano (instrument number 1) to a guitar. Specifically, we use General MIDI instrument numbers 25 (Acoustic Guitar, nylon), aligning the output to a guitar-friendly format. We do not deal with extended playing techniques (pitch bends, harmonics, slides).



5.4.2 Training from scratch

To establish a baseline for guitar cover generation, we first attempted to train the model from scratch using our Pop2Guitar dataset. We employ the same model architecture as [1] with our adapted tokenization schema for guitar MIDI output, as described in Section 5.4.1.

While our dataset contains only 40 song-cover pairs, this training approach serves as an important baseline for comparison with our domain adaptation methods. We acknowledge that this limited dataset size is insufficient for optimal performance, as successful from-scratch training typically requires datasets that are orders of magnitude larger. However, this experiment provides valuable insights into the challenges of low-resource guitar cover generation and establishes a lower bound for model performance in our evaluation framework.

5.4.3 Domain Adaptation

To address the problem of lack of training data for instruments other than piano we adapt a recently proposed model to meet this need. Having created a dataset of aligned guitar tracks and covers presented in 4.3, we now apply this data to the downstream task of training an guitar cover generation model. We approach this task as a domain adaptation task, where we take an existing state-of-the-art piano cover generation model, trained on a large dataset, and fine-tune it using our much smaller guitar dataset . The source model we use is the same model used for Greek piano cover generation in section 5.3.

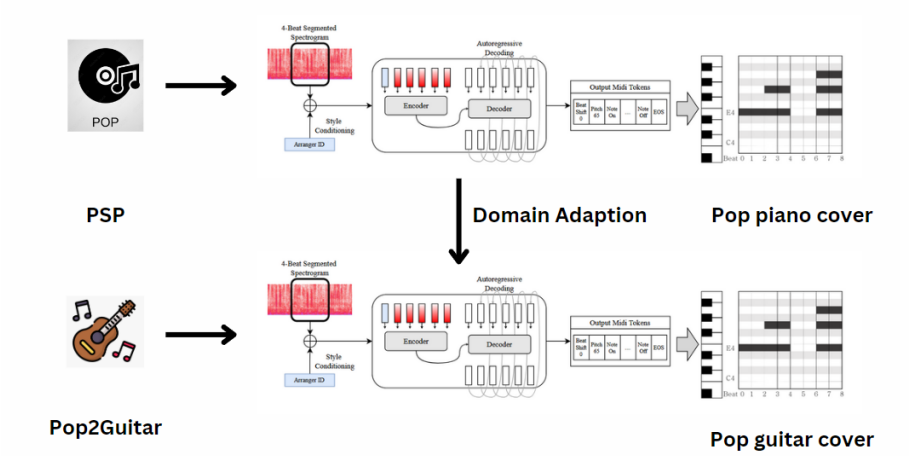


Figure 5.7. *Domain Adaptation from Pop2Piano Model*

Partial Fine-tuning

In partial fine-tuning, we update only specific components of the pre-trained piano cover generation model. We freeze all the encoder layers and the earlier layers of the decoder, as these layers primarily learn general representations of music that remain relevant across domains. We fine-tune only the last few layers of the decoder, which are

responsible for task-specific generation, and the language modelling (LM) head, which maps the decoder outputs to the vocabulary space.

This approach reduces the chance of overfitting, especially since our guitar dataset is small, and it keeps the general knowledge learned from the larger piano dataset. By focusing on the output layers, the model learns to capture the specific style and structure of guitar music without the need to retrain the entire model.

Full Fine-tuning

In full fine-tuning, we unfreeze all layers of the pre-trained model and update all parameters during training. This approach allows the model to adapt both the general representations learned by the encoder and the task-specific generation processes handled by the decoder. While, needing more time and GPU resourses, full fine-tuning can lead to domain-specific adaptation, particularly when the target domain (guitar music) differs significantly from the source domain of piano music.

Full fine-tuning is especially useful when the target dataset is sufficiently large to support such extensive parameter updates, or when the model needs to learn intricate domain-specific details that are not captured by the frozen layers. Care is taken to prevent overfitting, because the dataset is small, using early stopping.

5.5 Sequential Fine-Tuning for Greek Song-to-Guitar Generation

As an exploratory extension, we investigate a multi-stage domain adaptation path that leverages knowledge transfer across both cultural and instrumental boundaries. This approach, which we term "Greek2Guitar," follows a two-stage progression:

Stage 1: Greek2Piano adaptation (Greek songs $\rightarrow$ piano covers)

Stage 2: Greek2Guitar adaptation (Greek songs $\rightarrow$ guitar covers)

This strategy first adapts the model to Greek musical characteristics while maintaining the familiar piano output format, then subsequently adapts the culturally aware model to guitar specific constraints. The hypothesis is that this intermediate adaptation will better preserve Greek musical patterns (such as rhythmic structures, and melodic characteristics) during the final instrumental adaptation phase.

By decomposing the adaptation challenge into cultural adaptation followed by instrumental adaptation, we aim to achieve more effective knowledge transfer than direct cross-domain adaptation, ultimately producing higher-quality guitar covers for Greek songs.

Chapter 6

Experiments & Results

Building on the methods explained in Chapter 5, this chapter describes how we carried out our experiments and shares the results. We start by explaining the setups and training steps for each approach, followed by a clear analysis of the results.

6.1 Implementation Details for Greek Song-to-Piano Cover Generation Models

In this section, we will present our training setup, our specific configurations and implementations focusing on Greek-to-Piano cover generation models following our methodological framework.

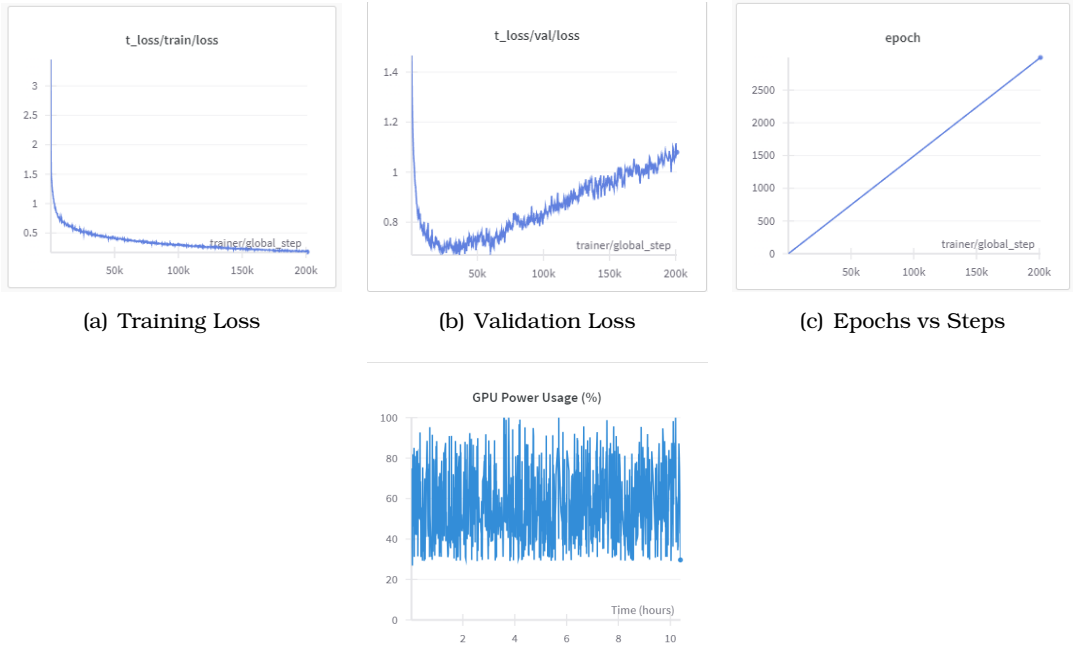
6.1.1 Training Setup

Before training, we establish some basic configurations. We use a fixed seed value of 3407 for all processes to ensure reproducibility. All training is performed on a single NVIDIA GeForce GTX 1080 Ti GPU provided by the SLP-NTUA lab's server. The batch size is set to 8, as higher values resulted in CUDA out-of-memory errors, and the number of workers is also set to 8. Furthermore, the following configurations are employed: `feed_forward_proj` is set to `"gated-gelu"`, `tie_word_embeddings` is disabled (`false`), `tie_encoder_decoder` is also disabled (`false`), the vocabulary size is fixed at 2400, the maximum number of positions (`n_positions`) is set to 1024 (potentially expandable), and `relative_attention_num_buckets` is configured to 32. These settings, together with our training hardware, ensure a stable and reproducible training pipeline.

To monitor our training, we set up the `wandb` framework, which provides insights into the training process. This includes monitoring the training and validation loss, tracking the number of steps corresponding to epochs, observing GPU usage during training, and other relevant metrics. In addition, we implement two callback functions for the monitoring process. The first callback evaluates the validation loss after every epoch and saves the model if it outperforms the previously saved version. The second callback ensures that the latest checkpoint of the model is always saved, providing a fallback in case of unexpected interruptions. These measures ensure effective tracking and safe preservation of our training progress.

6.1.2 Training from Scratch

First we tried to train our model from scratch. The model has 59.1 million trainable parameters, is optimized using AdaFactor [50] and has a learning rate of $1e-3$. Empirical testing and previous studies [10, 1] helped us choose this configuration. The model is trained for 3000 epochs which took roughly 11 hours to complete. Throughout the training we closely monitor the training and validation losses and also listen to pairs of generated covers and original songs to understand if the model is getting better. Below Figure 6.1 provides details about the losses, training steps and GPU usage.



(a) Training Loss (b) Validation Loss (c) Epochs vs Steps

(d) GPU Usage

Figure 6.1. *Training details of the model: (a) Training loss, (b) Validation loss, (c) Epochs vs Steps, and (d) GPU usage during training.*

The above figure gives us deeper insights about our training. We can see that the model overfits the training data and even though it has a very low training loss, the validation loss skyrockets. To deal with this problem we keep checkpoints of the model when the validation loss is at its lowest. The best model checkpoint is from epoch 544 which gives balance between understanding the training data and being able to generalize to data it has never seen. We can estimate that about 158 hours of music were used to train the best model assuming an average bpm of 120.

6.1.3 Transfer Learning

We continue our experiments by applying transfer learning to the pre-trained model in our Greek Dataset. This way we will leverage the ability of the model to generate piano covers for pop songs and refine its parameters for our specific dataset. We will explore two strategies: partial and full fine-tuning to figure out which one is best for our case.

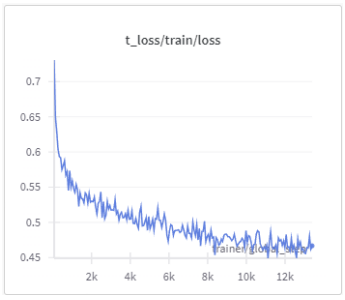
Partial Fine-tuning

In our partial fine-tuning approach, we freeze the early layers of the model, and only update the later layers. The model architecture consists of 6 encoder-decoder layers and a language model head. We experiment with unfreezing different layers and determine that unfreezing the last two decoder layers suits best.

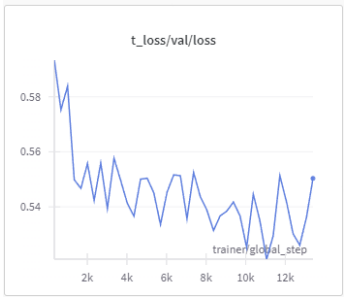
Configuration	Trainable Params	Non-trainable Params	Total Params
Two Last Decoder Layers + LM Head	11.7 M	47.4 M	59.1 M
Last Decoder + LM Head	6.5 M	52.6 M	59.1 M
All Layer + LM Head	59.1 M	0	59.1 M

Table 6.1. *Parameter counts for different fine-tuning configurations.*

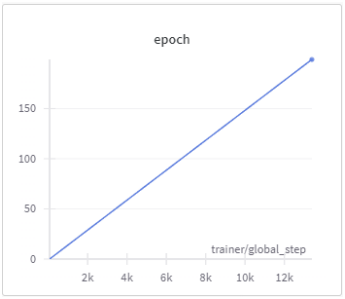
We keep the same configurations we had for training from scratch. The model is trained for 200 epochs which took close to 20 minutes to complete. We keep the best checkpoint (epoch 165) which has the lowest validation loss. We can estimate that about 48 hours of music were used to train the best model assuming an average bpm of 120. The figure below 6.2 provides details about losses, training steps, and GPU usage. The training loss consistently decreases. However, after approximately 8–10k steps, the validation loss no longer follows this downward trend and instead fluctuates.



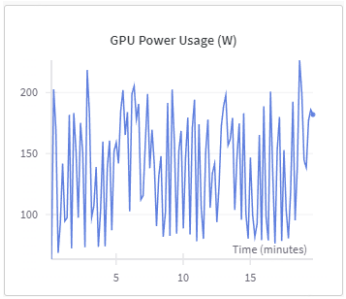
(a) Training Loss



(b) Validation Loss



(c) Epochs vs Steps

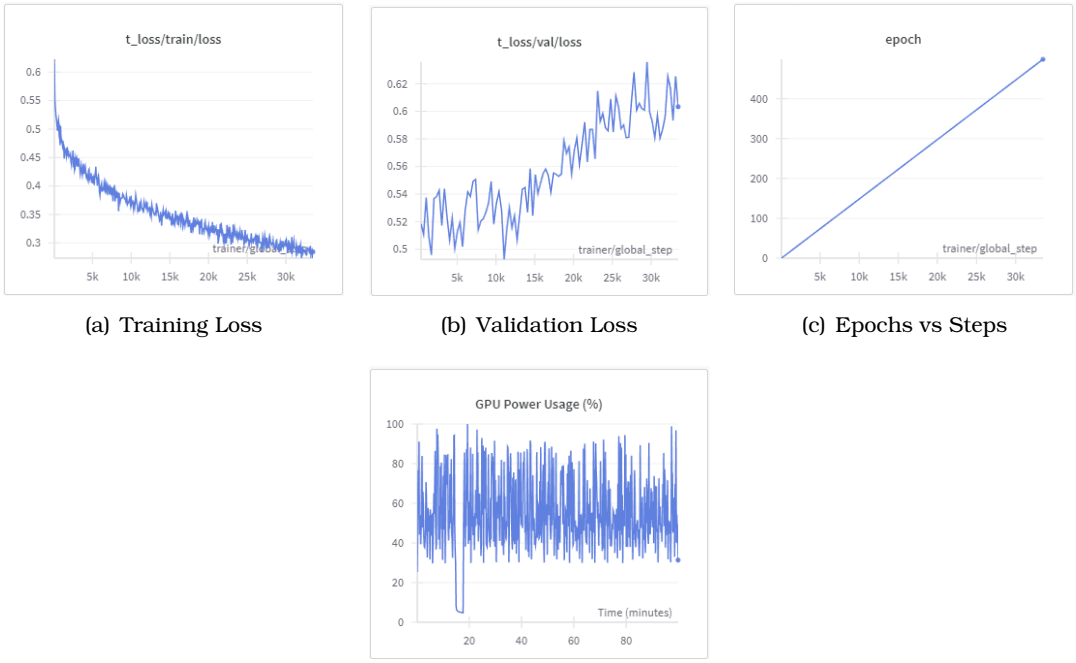


(d) GPU Usage

Figure 6.2. *Training details of the model: (a) Training loss, (b) Validation loss, (c) Epochs vs Steps, and (d) GPU usage during training.*

Full Fine-tuning

In contrast to partial fine-tuning, we allow all layers to update their weights, letting the model to fully adapt to the new data distribution. We keep the same configurations we had for training from scratch. The model is trained for 500 epochs, which took close to 2 hours to complete. We keep the best checkpoint (epoch 164) which has the lowest validation loss. We can estimate that about 48 hours of music were used to train the best model assuming an average bpm of 120. The figure below 6.3 provides details about losses, training steps, and GPU usage.



(a) Training Loss (b) Validation Loss (c) Epochs vs Steps

(d) GPU Usage

Figure 6.3. *Training details of the model: (a) Training loss, (b) Validation loss, (c) Epochs vs Steps, and (d) GPU usage during training.*

6.2 Implementation Details for Song-to-Guitar Cover Generation Models

In this section, we will present our training setup, our specific configurations and implementations focusing on Song-to-Guitar cover generation models following our methodological framework.

6.2.1 Training Setup

Prior to initiating training, we establish several fundamental configurations to ensure a stable and reproducible pipeline. A fixed seed value of 3407 is used across all processes to guarantee reproducibility. Training is conducted on a single NVIDIA GeForce GTX 1080 Ti GPU provided by the SLP-NTUA lab's server. We set the batch size to 8 because larger

batches led to CUDA out-of-memory errors, and we also configure the number of workers to 8. Additionally, the same configurations are employed as in the training setup of the piano cover generation models.

To monitor our training, we use the already set up `wandb` framework, which provides insights into the training process. The same metrics are monitored and callback functions are implemented ensuring effective tracking and safe preservation of our training progress.

Given the limited size of our Pop2Guitar dataset (40 pairs), we employed 5-fold cross-validation to ensure robust evaluation of our guitar cover generation models. The dataset was randomly partitioned into 5 folds, with each fold serving as the test set once while the remaining 4 folds were used for training. This approach maximizes the use of our limited data while providing more reliable performance estimates than a single train-test split.

For consistency in presentation, the training curves and GPU usage statistics shown in the following figures correspond to the first fold of our cross-validation setup. The final evaluation metrics reported in Section 6.4 represent the mean and 95% confidence intervals across all five folds.

6.2.2 Training from Scratch

For the training from scratch approach, we trained the guitar cover generation model for 2000 epochs using our Pop2Guitar dataset. The model achieved its best performance at epoch 1974, demonstrating the challenges of learning from limited data. Throughout training, we monitored both training and validation losses to track the model's learning progress and identify optimal stopping points.

The model exhibited clear signs of overfitting due to the severely limited dataset size of only 40 song-cover pairs. The small dataset forces the model to memorize training examples rather than learn generalizable patterns for guitar arrangement, resulting in poor performance on unseen data.

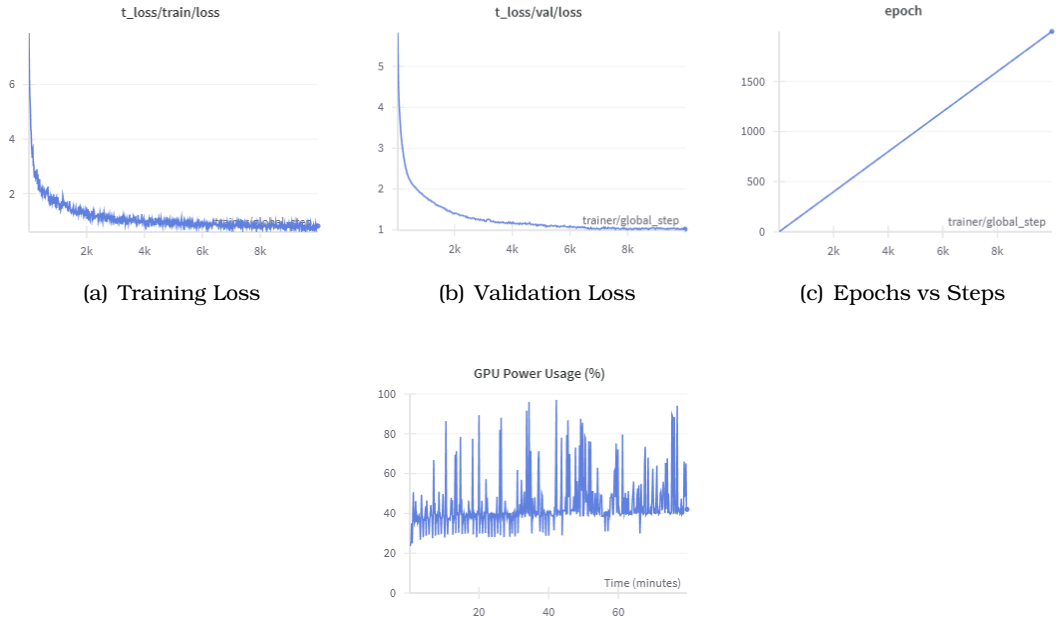
The training process required approximately one hour and 20 minutes to complete on our hardware setup. While the from-scratch approach provides a valuable baseline for comparison, the results confirm that domain adaptation techniques are essential for achieving reasonable performance in low-resource scenarios like guitar cover generation.

6.2.3 Domain Adaptation

We approach this task as a domain adaptation task, where we have the Pop2Piano pretrained model for piano cover generation trained on close to 5000 pairs and fine-tune it using our much smaller Pop2Guitar dataset. We will explore two strategies: partial and full fine-tuning to figure out which one is best for our case.

Partial Fine-tuning

In our partial fine-tuning approach, we freeze a subset of the model layers while updating the others. Our model architecture comprises six encoder-decoder layers and a language model head, and after experimenting with various freezing strategies, we found



(a) Training Loss (b) Validation Loss (c) Epochs vs Steps

(d) GPU Usage

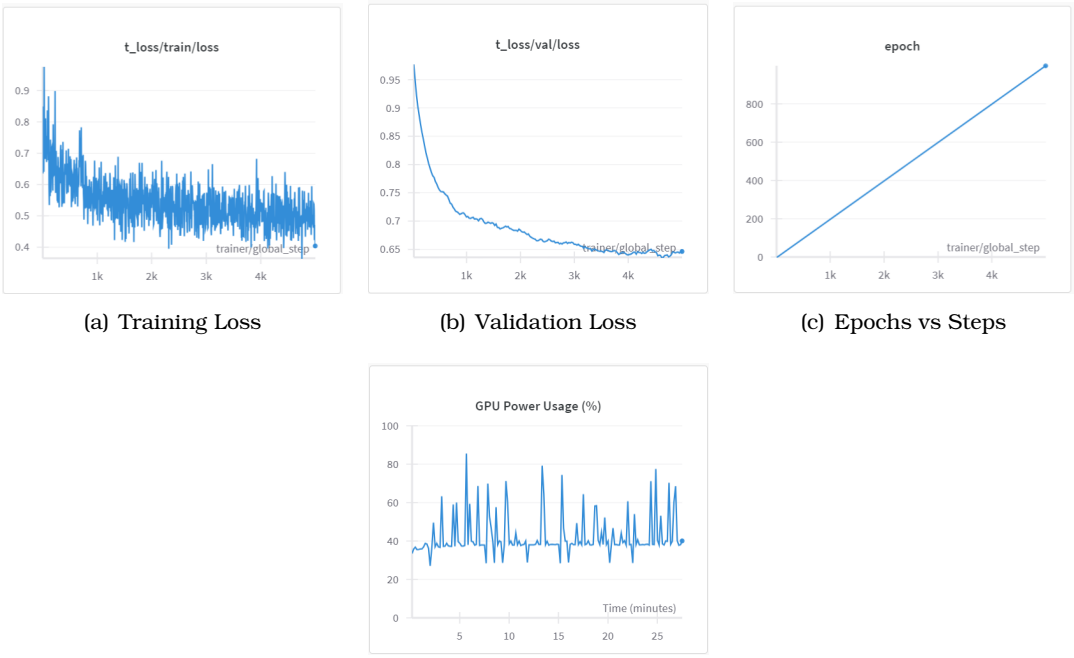
Figure 6.4. *Training details of the model: (a) Training loss, (b) Validation loss, (c) Epochs vs Steps, and (d) GPU usage during training.*

that unfreezing the final two decoder layers provides the optimal balance. The same observation was made for the piano generation model. We can see the parameter count for different fine-tuning configurations in Figure 6.1 . Also, we experiment with the learning rate (10^{-3}, 10^{-4}, 10^{-5}) and find best results with 10^{-4} .

The model is trained for 1000 epochs which took close to 25 minutes to complete. We keep the best checkpoint (epoch 929) which has the lowest validation loss. We can estimate that about 20 hours of music were used to train the best model assuming an average bpm of 120. The figure below 6.5 provides details about losses, training steps, and GPU usage. We observe that both the training and validation losses steadily decline during the early stages of training and then stabilize after approximately 3,000 steps.

Full Fine-tuning

In contrast to partial fine-tuning, we allow all layers to update their weights, letting the model to fully adapt to the new data distribution. We keep the same configurations we had for partial fine-tuning training. We experiment with the learning rate and choose 10^{-4} as the best fit. The model is trained for 1000 epochs, which took close to 35 minutes to complete. We keep the best checkpoint (epoch 174) which has the lowest validation loss. We can estimate that about 20 hours of music were used for the whole training and 3.5 hours to train the best model assuming an average bpm of 120. The figure below 6.6 provides details about losses, training steps, and GPU usage. We can see that even though the training loss decreases, the validation loss after close to 1k steps start to increase. This can be attributed to the small size of the dataset, so the model overfits



(a) Training Loss (b) Validation Loss (c) Epochs vs Steps

(d) GPU Usage

Figure 6.5. *Training details of the model: (a) Training loss, (b) Validation loss, (c) Epochs vs Steps, and (d) GPU usage during training.*

it after some time.

6.3 Implementation Details for Greek Song-to-Guitar Cover Generation Models

6.3.1 Training Setup

Before beginning the training process, we configure several key parameters to ensure pipeline stability and reproducibility. We set a fixed random seed of 3407 throughout all operations to guarantee consistent results across runs. The training is performed on a single NVIDIA GeForce GTX 1080 Ti GPU from the SLP-NTUA lab's server infrastructure. Due to GPU memory constraints that caused CUDA out-of-memory errors with larger configurations, we limit the batch size to 8 and set the number of workers to 8. We maintain consistency by applying the same parameter settings used in the piano cover generation model training phase.

6.3.2 Training Pipeline

Our sequential approach for Greek song-to-guitar generation follows the established two-stage progression:

Stage 1: Fine-tune on GreekSong2Piano dataset using the optimal configuration from Section 6.1, creating a Greek-culturally-aware piano generation model.

(a) Training Loss (b) Validation Loss (c) Epochs vs Steps

(d) GPU Usage

Figure 6.6. *Training details of the model: (a) Training loss, (b) Validation loss, (c) Epochs vs Steps, and (d) GPU usage during training.*

Stage 2: Using the Greek2Piano model as initialization, perform final adaptation on Pop2Guitar dataset with guitar covers, applying guitar-specific tokenization.

We begin with the best Greek2Piano model from full fine-tuning and further adapt it to the Pop2Guitar dataset using both partial and full fine-tuning approaches. In both cases, we train for 1000 epochs with a learning rate of 10^{-4}, selecting the best validation checkpoints at epoch 929 for partial fine-tuning and epoch 169 for full fine-tuning.

6.4 Evaluation Methodology

In this section, we present the evaluation methods used to assess the performance of our cover generation models. Our evaluation comprises both objective and subjective measures. The objective evaluation is based on several quantitative criteria, while the subjective evaluation relies on a user study.

6.4.1 Performance Analysis Across Evaluation Metrics

We adopt the following metrics to assess the quality of the generated covers from multiple perspectives. These metrics evaluate both the similarity of the covers to the original song and their adherence to the stylistic conventions and internal coherence characteristic of the human arrangements.

Melody Chroma Accuracy

Melody Chroma Accuracy (MCA) [46] evaluates the similarity between two monophonic melody sequences. The melody line plays a crucial role in deciding whether a song cover resembles the original song. Following the directions from [1] we compute the MCA between the vocals extracted by Spleeter [47] from the audio, and the top melodic line extracted from the cover MIDI using the skyline algorithm. To get the melody contours of music, the f0 sequence of the vocal is calculated using Librosa [83] and pYIN [116]. The analysis is performed with a sample rate of 44,100 Hz and a hop length of 1,024 samples.

Cover Song Identification

In order to assess the similarity between the original track and its generated cover, we employ a metric inspired by cover song identification, namely the metric $Q_{\max}$ [36]. The $Q_{\max}$ metric assesses the similarity of harmonic content between the generated cover and the reference, with lower values indicating a closer match. To calculate it, we first convert the generated MIDI to audio using the synthesizer FluidSynth with its basic soundfont. We then calculated the similarity between the original track and the generated audio using the Python implementation of ChromaCoverID [118]. Lastly, we average the $Q_{\max}$ value for all the pairs.

In addition, we employed a state-of-the-art CSI model which took part and ranked third in the MIREX 2024 Cover Song Identification challenge. We use CoverHunter [40] to extract rich, discriminative embeddings from the original recordings and covers. Then we calculate the cosine distance between the embeddings of the original and generated covers. This metric quantifies the degree of similarity in content between the original track and its various generated versions. Note that a pretrained CoverHunter model (CoverHunter-128) is employed. The bigger model with 256 dimensions was not available.

Embedding-based Similarity

Building upon the idea that we can calculate the similarity with the use of embeddings we employed MERT (Music undERstanding model with large-scale self-supervised Training) [51] a a large-scale, self-supervised learning (SSL) model designed for acoustic music understanding. It achieves comparable results across a wide range of MIR tasks while using significantly smaller parameter size and generates high-dimensional audio embeddings by aggregating hidden states across time and layers, capturing both fine-grained acoustic details and broader harmonic structure. We take advantage of these embeddings to calculate to cosine similarity between the original songs and different covers.

In our experiments, we employ the MERT-v1-95M model available from Hugging Face, selecting this smaller variant due to our compute limitations and its faster inference times. While we initially experimented with comparing whole track embeddings, we found that a segment-based approach yielded more reliable similarity measures. Therefore, we divide both the original song and its corresponding cover into non-overlapping 4-second segments. MERT comprises 12 representation layers, and to obtain a unified,

high-dimensional embedding vector for each segment, we first average the hidden states over the time dimension within each layer and then further average these results across all layers. This process effectively condenses the temporal dynamics and hierarchical information of each 4-second audio segment into a single robust representation. We then compute the cosine similarity between corresponding 4-second segments from the original track and its cover (i.e., segment 1 vs. segment 1, segment 2 vs. segment 2, etc.). Finally, we average all pairwise similarity scores across segments to obtain the overall similarity measure between the original song and its generated cover.

Piano Cover Evaluation

Table 6.2 summarizes the performance of our piano cover generation models across several objective metrics. The table lists each model alongside its corresponding scores for Melody Chroma Accuracy (MCA), Cover Song Identification performance as measured by $Q_{\max}$, CoverHunter embedding distance, and MERT embedding similarity. MCA evaluates how effectively the generated cover preserves the melodic characteristics of the original song, while $Q_{\max}$ reflects the harmonic similarity between the generated cover and the reference. The embedding distance metrics derived from both CoverHunter and similarity from MERT capture the overall similarity in learned representations, providing complementary perspectives on the quality of the generated covers.

Table 6.2. *Evaluation Metrics for Generated Piano Covers. Values are mean ± 95 % confidence interval. Higher MCA and embedding similarity (MERT) values are preferable, whereas lower CSI ($Q_{\max}$) and embedding distances (CoverHunter) indicate better performance.*

Model	MCA ↑	CSI ($Q_{\max}$) ↓	CoverHunter Distance ↓	MERT Similarity↑
Pop2Piano [1]	0.363 ± 0.019	0.075 ± 0.017	0.159 ± 0.015	0.808 ± 0.007
Greek2Piano-Scratch	0.372 ± 0.020	0.100 ± 0.020	0.175 ± 0.012	0.802 ± 0.008
Greek2Piano-Partial	**0.443 ± 0.021**	0.068 ± 0.017	0.155 ± 0.013	0.809 ± 0.007
Greek2Piano-Full	0.439 ± 0.022	**0.064 ± 0.013**	**0.146 ± 0.013**	**0.811 ± 0.009**
Human Piano	0.389 ± 0.029	0.093 ± 0.028	0.142 ± 0.014	0.794 ± 0.017
Human Piano (Audio)	–	0.087 ± 0.026	0.134 ± 0.013	0.834 ± 0.007

For piano covers, fine-tuning strategies demonstrate clear improvements over both the Pop2Piano baseline and from-scratch training. The partial fine-tuning approach achieves the best performance with an MCA of 0.443 ± 0.021 and the partial fine-tuning with a $Q_{\max}$ of 0.064 ± 0.013 (lower is better), and a CoverHunter distance of 0.146, representing improvements of 21.0% in MCA and 14.7% in $Q_{\max}$ compared to the baseline. Even the from-scratch model trained exclusively on our GreekSong2Piano dataset (0.372 ± 0.020 MCA) outperforms the Pop2Piano baseline in melodic accuracy, demonstrating the value of domain-specific training for Greek music adaptation.

Guitar Cover Evaluation

Table 6.3 summarizes the performance of our guitar cover generation models across several objective metrics. We employ the same evaluation metrics as in piano cover

generation: Melody Chroma Accuracy (MCA), Cover Song Identification performance as measured by $Q_{\max}$, CoverHunter embedding distance, and MERT embedding similarity.

Table 6.3. *Evaluation metrics for generated guitar covers (5-fold experiment). Values are mean ± 95 % confidence interval. Higher MCA and MERT indicate better quality; lower CSI ($Q_{\max}$) and CoverHunter distance indicate better quality.*

Model	MCA ↑	CSI ($Q_{\max}$) ↓	CoverHunter Distance ↓	MERT Similarity ↑
Pop2Guitar-Scratch	0.189 ± 0.016	0.576 ± 0.105	0.181 ± 0.030	0.735 ± 0.007
Pop2Guitar-Partial	0.358 ± 0.043	**0.152 ± 0.050**	**0.153 ± 0.010**	0.781 ± 0.016
Pop2Guitar-Full	**0.363 ± 0.042**	0.169 ± 0.062	0.156 ± 0.014	**0.783 ± 0.024**
Human Guitar	0.288 ± 0.018	0.211 ± 0.053	0.168 ± 0.022	0.777 ± 0.014

In the guitar generation task, the performance differences between strategies are substantial. Both fine-tuning approaches achieve strong results, with the full fine-tuned model reaching the highest MCA (0.363 ± 0.042) and MERT similarity (0.783 ± 0.024), while the partial fine-tuned model achieves the best CSI performance (0.152 ± 0.050 $Q_{\max}$ $Q_{\max}$) and lowest CoverHunter distance (0.153 ± 0.010). In contrast, the from-scratch model trained on only 40 guitar cover pairs performs poorly across all metrics (0.189 ± 0.016 MCA, 0.576 ± 0.105 $Q_{\max}$ $Q_{\max}$), highlighting the importance of transfer learning when working with limited datasets.

It is worth noting that our fine-tuned models outperform human-created covers across multiple objective metrics, including MCA (0.363 ± 0.042 vs. 0.288 ± 0.018 for guitar), despite human covers receiving superior subjective ratings. This apparent contradiction reflects a fundamental difference in approach: while our models optimize for melodic fidelity, human arrangers prioritize artistic interpretation over literal reproduction, introducing creative variations and instrument-specific techniques that enhance musical expressiveness but reduce measurable similarity. This pattern is consistent with findings in the original Pop2Piano paper [1], where human arrangements similarly scored lower on computational metrics yet received higher subjective ratings. The higher MERT similarity scores for human covers suggest that these creative deviations, while reducing note-level accuracy, ultimately contribute to the overall musical quality that listeners value.

Interestingly, when the same human performances are evaluated directly in their recorded audio form (*Human Piano (Audio)*), they achieve not only higher MERT similarity but also improved CoverHunter distance and CSI (Qmax), outperforming both the MIDI-rendered human covers and all model-generated outputs. This discrepancy highlights how the transcription–preprocessing–re-rendering pipeline introduces degradations that suppress metric scores.

6.4.2 User Perception and Subjective Quality Assessment

For subjective evaluation we conduct a user study. Our user study with 26 non-professional participants evaluated 10-second excerpts from test set songs across three dimensions: Similarity to Original (SI), Musical Coherence (CO), and Listener Enjoyment (LE). Excerpts were presented anonymously in randomized order to ensure unbiased assessment.

Subjects are asked to listen to these audio clips and provide ratings on a 5-point Likert scale for the following aspects:

- **Similarity to Original (SI):** The degree of similarity between the piano/guitar performances and the original song.
- **Music Coherence (CO):** The degree of perceived fluency in the music, representing the smoothness and coherence of the piano/guitar performances.
- **Listener Enjoyment (LE) :** How much the participants like the piano/guitar cover in their overall listening experience.

User Study

Table 6.4. *Evaluation Metrics for Generated Piano Covers. Higher values for Similarity, Music Fluency, and Overall indicate better performance.*

Model	Similarity to Original (↑)	Music Coherence (↑)	Listener Enjoyment (↑)
Pop2Piano [1]	2.29 ± 0.20	2.60 ± 0.26	2.40 ± 0.25
Greek2Piano-Scratch	1.81 ± 0.21	2.42 ± 0.26	1.97 ± 0.21
Greek2Piano-Partial	2.67 ± 0.22	2.60 ± 0.23	2.46 ± 0.24
Greek2Piano-Full	$\mathbf{2.94 \pm 0.21}$	$\mathbf{2.91 \pm 0.23}$	$\mathbf{2.72 \pm 0.25}$
Human Piano	4.06 ± 0.23	3.94 ± 0.25	3.78 ± 0.28

Table 6.5. *Evaluation Metrics for Generated Guitar Covers. Higher values for Similarity, Music Fluency, and Overall indicate better performance.*

Model	Similarity to Original(↑)	Music Coherence (↑)	Listener Enjoyment(↑)
Pop2Guitar-Scratch	1.54 ± 0.28	1.77 ± 0.27	1.54 ± 0.24
Pop2Guitar-Partial	$\mathbf{2.56 \pm 0.29}$	$\mathbf{2.50 \pm 0.28}$	$\mathbf{2.29 \pm 0.31}$
Pop2Guitar-Full	2.27 ± 0.25	2.35 ± 0.26	2.06 ± 0.26
Human Guitar	3.13 ± 0.26	2.87 ± 0.34	2.71 ± 0.33

The subjective evaluation results in Table 6.4 demonstrate strong correspondence with our objective metrics. For piano covers, the full fine-tuning approach received the highest ratings across all dimensions (reported as average with 95% confidence intervals: SI: 2.94 ± 0.21, CO: 2.91 ± 0.23, LE: 2.72 ± 0.25) approaching the human benchmark and outperforming both the Pop2Piano baseline and the from-scratch model. This confirms that fine-tuning enhances not only technical metrics but also perceived musicality and enjoyment.

The guitar models in Table 6.5 show a similar pattern but with more pronounced differences. The from-scratch model scored poorly on all measures (SI: 1.54 ± 0.28, CO: 1.77 ± 0.27, LE: 1.54 ± 0.24), while fine-tuned models achieved substantially higher ratings, with the partial fine-tuning approach receiving particularly strong ratings for similarity (2.56 ± 0.29) and coherence (2.50 ± 0.28).

Most notably, our exploratory sequential fine-tuning experiment (Greek2Guitar) demonstrates the potential of layered domain adaptation. The partial fine-tuned sequential

Table 6.6. *Evaluation Metrics for Greek-to-Guitar Covers. Higher values for Similarity Index (SI), Coherence (CO), and Listener Enjoyment (LE) indicate better performance.*

Model	**Similarity to Original(↑)**	**Music Coherence (↑)**	**Listener Enjoyment (↑)**
Base (No Fine-tuning)	2.37 ± 0.32	2.10 ± 0.31	1.90 ± 0.29
Sequential-Partial	$\mathbf{3.31} \pm 0.33$	$\mathbf{3.00} \pm 0.33$	$\mathbf{3.00} \pm 0.33$
Sequential-Full	3.19 ± 0.28	2.67 ± 0.29	2.50 ± 0.29
Human (Greek Guitar)	4.17 ± 0.28	3.85 ± 0.31	3.65 ± 0.38

model achieved the highest similarity rating (3.31±0.33) among all guitar models, approaching the human benchmark (4.17±0.28). This suggests that the knowledge transfer path from Western pop piano to Greek piano to guitar effectively captures important musical characteristics that enhance the perceived quality of generated covers.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

Cover song generation represents a challenge in Music Information Retrieval, requiring systems to preserve the musical essence of original compositions while adapting them to specific instruments and styles. This thesis addressed two fundamental limitations in the field: the scarcity of training data for non-Western musical traditions and the lack of cover generation models for instruments beyond piano. Through systematic investigation of transfer learning approaches and the creation of specialized datasets, we demonstrated strategies for automatic cover generation in low-resource scenarios.

Our primary dataset contribution, the GreekSong2Piano dataset, consists of 659 Greek songs paired with their corresponding piano covers, totaling 41 hours of music across eight distinct Greek genres including Rembetiko, Laiko, and Entexno. This dataset captures the unique characteristics of Greek musical traditions, providing the first synchronized collection specifically designed for Greek music cover generation. Additionally, we created the Pop2Guitar dataset with 40 song-guitar pairs, enabling exploration of cross-instrument domain adaptation beyond the piano-centric approaches that have dominated the field.

Our systematic analysis of training strategies revealed clear performance advantages for transfer learning approaches over training from scratch. When comparing from-scratch training, partial fine-tuning, and full fine-tuning on our Greek dataset, both fine-tuning approaches achieved higher performance than the Pop2Piano baseline, with the partial fine-tuning approach reaching the highest MCA of 0.443 ± 0.021, representing a 21.0% improvement over the baseline. Even our from-scratch model, trained exclusively on Greek music, competed closely with the original Pop2Piano model on Greek songs, demonstrating the value of domain-specific training. For guitar generation, transfer learning proved even more critical, with fine-tuned models achieving substantially higher performance than from-scratch approaches given the limited training data.

We introduced a novel sequential fine-tuning strategy that performs multi-step domain adaptation: from Western pop piano covers to Greek piano covers to guitar covers. This approach achieved promising results, with the partial fine-tuned sequential model receiving the highest similarity ratings (3.31 ± 0.33) among guitar models in our user study, approaching human performance (4.17 ± 0.28). This suggests that knowledge can be ef-

fectively transferred across both cultural and instrumental boundaries through carefully designed adaptation paths.

Our evaluation framework combined objective metrics with subjective assessment to provide comprehensive quality assessment. We employed Melody Chroma Accuracy (MCA), cover song identification metrics and embedding-based approaches using state-of-the-art models like CoverHunter and MERT. The subjective evaluation through user studies with 26 participants validated our objective findings, showing strong correspondence between computational metrics and human perception of cover quality.

In summary, our key contributions are:

- **Novel datasets for underexplored domains:** Created the first synchronized Greek song-piano cover dataset (GreekSong2Piano dataset) and expanded cover generation to guitar arrangements with the Pop2Guitar dataset.
- **Systematic analysis of transfer learning strategies:** Demonstrated that partial and full fine-tuning consistently outperforms from-scratch training in low-resource scenarios enabling adaptation across stylistic and instrumental domains.
- **Multi-step domain adaptation approach::** Introduced sequential fine-tuning across cultural and instrumental boundaries, showing effective knowledge transfer from pop piano to Greek piano to guitar covers.
- **Comprehensive evaluation framework:** Established objective metrics combining melodic and embedding-based similarity measures with subjective user assessment, demonstrating strong correlation between computational measures and human perceptual judgments.

7.2 Limitations and Future Work

While this work demonstrates the potential of transfer learning for cross-cultural and cross-instrumental cover generation, several limitations highlight areas for improvement. Our datasets face inherent quality constraints. Unlike professionally recorded datasets such as MAESTRO [52], which provides virtuosic piano performances with fine-grained alignment $\approx 3\,\text{ms}$ between note labels and audio waveforms, or GuitarSet [53] with its hexaphonic pickup recordings, our data is sourced from YouTube and synchronized using computational methods that, while effective, cannot achieve the same level of precision. This introduces potential timing discrepancies that may affect model training quality. Additionally, our subjective evaluation, while providing valuable insights, involved 26 non-professional participants, suggesting that larger-scale studies incorporating diverse listener populations and professional musicians could provide more robust validation of our findings.

Also, technical limitations constrained our experimental scope. Most notably, our reliance on pre-existing MIDI files from MuseScore for the Pop2Guitar dataset was necessitated by the poor performance of current automatic music transcription models on guitar recordings. Despite MT3's design for multi-instrument transcription [10], it frequently

misidentified guitar performances as other instruments, making it unsuitable for creating the synchronized audio-MIDI pairs essential to our approach. This limitation forced us to work with a substantially smaller guitar dataset (40 pairs) compared to our piano dataset (659 pairs). Furthermore, computational resource limitations constrained our our experimental scope. All experiments were conducted on the SLP-NTUA lab's server equipped with two 12GB GPUs (NVIDIA GeForce GTX 1080 Ti and GeForce GTX TITAN X), which constrained our ability to experiment with larger context lengths and batch sizes in our training process. Similarly, during evaluation, we were limited to using MERT-95M [51] rather than the more capable MERT-330M model due to memory constraints, potentially affecting the quality of our embedding-based similarity assessments.

Future research should explore expanded cultural domains and musical styles beyond the Greek and Western pop traditions examined in this work. Greek music, with its distinctive rhythmic patterns and unique structural characteristics [42], provided an effective test case for cross-cultural adaptation, but the principles demonstrated here could extend to other traditions with unique characteristics, such as Indian classical music [54], Arabic maqam systems [55], or East Asian musical forms [56]. Such extensions would further test the generalizability of transfer learning approaches across more diverse musical parameters.

Just as automatic music transcription has successfully expanded from piano-focused systems to encompass diverse instrumental families, cover generation could follow a similar trajectory given appropriate dataset development. Recent advances in AMT have demonstrated successful transcription across strings (violin, cello) [57], woodwinds (flute) [58], drums [59] and traditional ethnic instruments like the Arabian flute [60]. Cover generation could similarly expand to these instruments, though this would require developing larger-scale synchronized datasets and more sophisticated tokenization schemes to handle instrument-specific techniques.

A particularly promising extension involves creating comprehensive end-to-end systems that transform audio input directly into performance-ready notation. Building upon our current audio-to-MIDI pipeline, such systems could integrate post-processing modules for musical notation generation. Recent work on MIDI-to-score transformation, such as the MIDI2ScoreTransformer [61] , demonstrates the feasibility of converting symbolic music into readable sheet music. Similarly, guitar-specific systems could leverage MIDI-to-tablature conversion models [62] to produce instrument-appropriate notation. An end-to-end system could potentially unify three stages: (1) audio-to-symbolic conversion using our trained cover generation models, (2) symbolic-to-notation transformation using specialized rendering models, and (3) joint optimization across the entire pipeline.

Finally, another direction of this thesis involves conditional cover generation with textual or multi-modal inputs. By incorporating various types of input conditions, such as chords, melody tracks, lyrics, and text descriptions, not only can users interact with the music generation process more dynamically, but they also gain higher and more fine-grained control over the output [63]. By leveraging models like ChatMusician [64], Llark [65] or expanding the T5 model [5] future models could accept natural language descriptions like "create a melancholic piano cover in the style of a classical ballad" or

"generate an upbeat guitar cover arrangement suitable for a folk festival."

We hope that our work will help attract more attention to the challenging problem of music cover generation and inspire new research on musical cross-cultural exchange through computational approaches. By bridging diverse musical traditions and instrumental forms, systems like ours could ultimately contribute to both creative applications and deeper computational understanding of musical translation across cultural boundaries.

Bibliography

[1] Jongho Choi και Kyogu Lee. *Pop2Piano: Pop audio-based piano cover generation*. *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, σελίδες 1–5. IEEE, 2023.

[2] Ian Goodfellow. *Deep learning*, τόμος 196. MIT press, 2016.

[3] Laith Alzubaidi, Jinglan Zhang, Amjad J Humaidi, Ayad Al-Dujaili, Ye Duan, Omran Al-Shamma, José Santamaría, Mohammed A Fadhel, Muthana Al-Amidie και Laith Farhan. *Review of deep learning: concepts, CNN architectures, challenges, applications, future directions*. *Journal of big Data*, 8:1–74, 2021.

[4] A Vaswani. *Attention is all you need*. *Advances in Neural Information Processing Systems*, 2017.

[5] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li και Peter J Liu. *Exploring the limits of transfer learning with a unified text-to-text transformer*. *Journal of machine learning research*, 21(140):1–67, 2020.

[6] Emmanouil Benetos, Simon Dixon, Zhiyao Duan και Sebastian Ewert. *Automatic music transcription: An overview*. *IEEE Signal Processing Magazine*, 36(1):20–30, 2018.

[7] Curtis Hawthorne, Erich Elsen, Jialin Song, Adam Roberts, Ian Simon, Colin Raffel, Jesse Engel, Sageev Oore και Douglas Eck. *Onsets and frames: Dual-objective piano transcription*. *arXiv preprint arXiv:1710.11153*, 2017.

[8] Qiuqiang Kong, Bochen Li, Xuchen Song, Yuan Wan και Yuxuan Wang. *High-resolution piano transcription with pedals by regressing onset and offset times*. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3707–3717, 2021.

[9] Curtis Hawthorne, Ian Simon, Rigel Swavely, Ethan Manilow και Jesse Engel. *Sequence-to-sequence piano transcription with transformers*. *arXiv preprint arXiv:2107.09142*, 2021.

[10] Josh Gardner, Ian Simon, Ethan Manilow, Curtis Hawthorne και Jesse Engel. *MT3: Multi-task multitrack music transcription*. *arXiv preprint arXiv:2111.03017*, 2021.

[11] Ethan Manilow, Gordon Wichern, Prem Seetharaman και Jonathan Le Roux. *Cutting music source separation some Slakh: A dataset to study the impact of training data quality and quantity*. *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, σελίδες 45–49. IEEE, 2019.

[12] John Thickstun, Zaid Harchaoui και Sham Kakade. *Learning features of music from scratch*. *arXiv preprint arXiv:1611.09827*, 2016.

[13] Donald P Pazel. *An Introduction to Music Transformations*. *Music Representation and Transformation in Software: Structure and Algorithms in Python*, σελίδες 175–178. Springer, 2022.

[14] Ondřej Cífka, Umut Şimşekli και Gaël Richard. *Groove2groove: One-shot music style transfer with supervision from synthetic data*. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2638–2650, 2020.

[15] Shih Chuan Chiu, Man Kwan Shan και Jiun Long Huang. *Automatic system for the arrangement of piano reductions*. *2009 11th IEEE International Symposium on Multimedia*, σελίδες 459–464. IEEE, 2009.

[16] Graham Percival, Satoru Fukayama και Masataka Goto. *Song2Quartet: A System for Generating String Quartet Cover Songs from Polyphonic Audio of Popular Music*. *ISMIR*, σελίδες 114–120, 2015.

[17] Shunya Ariga, Satoru Fukayama και Masataka Goto. *Song2Guitar: A Difficulty-Aware Arrangement System for Generating Guitar Solo Covers from Polyphonic Audio of Popular Music*. *ISMIR*, σελίδες 568–574. Suzhou, 2017.

[18] Tom M Mitchell και Tom M Mitchell. *Machine learning*, τόμος 1. McGraw-hill New York, 1997.

[19] Trevor Hastie, Robert Tibshirani, Jerome Friedman, Trevor Hastie, Robert Tibshirani και Jerome Friedman. *Overview of supervised learning*. *The elements of statistical learning: Data mining, inference, and prediction*, σελίδες 9–41, 2009.

[20] Zoubin Ghahramani. *Unsupervised learning*. *Summer school on machine learning*, σελίδες 72–112. Springer, 2003.

[21] Richard S Sutton, Andrew G Barto και others. *Reinforcement learning: An introduction*, τόμος 1. MIT press Cambridge, 1998.

[22] Yann LeCun, Léon Bottou, Yoshua Bengio και Patrick Haffner. *Gradient-based learning applied to document recognition*. *Proceedings of the IEEE*, 86(11):2278–2324, 2002.

[23] Sepp Hochreiter και Jürgen Schmidhuber. *Long short-term memory*. *Neural computation*, 9(8):1735–1780, 1997.

[24] *MuseScore: Free music composition and notation software*. https://musescore.org. Ημερομηνία πρόσβασης:07-01-2025.

[25] Aaron Van Den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, Koray Kavukcuoglu και others. *Wavenet: A generative model for raw audio*. *arXiv preprint arXiv:1609.03499*, 12, 2016.

[26] Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford και Ilya Sutskever. *Jukebox: A generative model for music*. *arXiv preprint arXiv:2005.00341*, 2020.

[27] Matija Marolt. *A connectionist approach to automatic transcription of polyphonic piano music*. *IEEE Transactions on Multimedia*, 6(3):439–449, 2004.

[28] Kristy Choi, Curtis Hawthorne, Ian Simon, Monica Dinculescu και Jesse Engel. *Encoding musical style with transformer autoencoders*. *International conference on machine learning*, σελίδες 1899–1908. PMLR, 2020.

[29] *MuseNet: AI-generated music from multiple styles and instruments*. https://openai.com/index/musenet/. Ημερομηνία πρόσβασης: 07-01-2025.

[30] Sho Onuma και Masatoshi Hamanaka. *Piano Arrangement System Based On Composers' Arrangement Processes*. *ICMC*, 2010.

[31] Yuki Hoshi, Ryohei Orihara, Yuichi Sei, Yasuyuki Tahara και Akihiko Ohsuga. *Versatile Automatic Piano Reduction Generation System by Deep Learning*. *2022 2nd International Conference on Advanced Research in Computing (ICARC)*, σελίδες 66–71. IEEE, 2022.

[32] Joan Serra, Emilia Gómez, Perfecto Herrera και Xavier Serra. *Chroma binary similarity and local alignment applied to cover song identification*. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(6):1138–1151, 2008.

[33] Colin Larkin. *The encyclopedia of popular music*. Omnibus Press, 2011.

[34] Joan Serra, Emilia Gómez και Perfecto Herrera. *Audio cover song identification and similarity: background, approaches, evaluation, and beyond*. *Advances in music information retrieval*, σελίδες 307–332, 2010.

[35] Joan Serra. *Identification of versions of the same musical composition by processing audio descriptions*. *Department of Information and Communication Technologies*, 2011.

[36] Joan Serra, Xavier Serra και Ralph G Andrzejak. *Cross recurrence quantification for cover song identification*. *New Journal of Physics*, 11(9):093017, 2009.

[37] Zhesong Yu, Xiaoshuo Xu, Xiaoou Chen και Deshun Yang. *Temporal Pyramid Pooling Convolutional Neural Network for Cover Song Identification*. *IJCAI*, σελίδες 4846–4852, 2019.

[38] Xingjian Du, Zhesong Yu, Bilei Zhu, Xiaoou Chen και Zejun Ma. *Bytecover: Cover song identification via multi-loss training*. *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, σελίδες 551–555. IEEE, 2021.

[39] Ken O'Hanlon, Emmanouil Benetos και Simon Dixon. *Detecting cover songs with pitch class key-invariant networks*. *2021 IEEE 31st International Workshop on Machine Learning for Signal Processing (MLSP)*, σελίδες 1–6. IEEE, 2021.

[40] Feng Liu, Deyi Tuo, Yinan Xu και Xintong Han. *CoverHunter: Cover Song Identification with Refined Attention and Alignments*. *2023 IEEE International Conference on Multimedia and Expo (ICME)*, σελίδες 1080–1085. IEEE, 2023.

[41] Ziyu Wang, Ke Chen, Junyan Jiang, Yiyi Zhang, Maoran Xu, Shuqi Dai, Xianbin Gu και Gus Xia. *Pop909: A pop-song dataset for music arrangement generation*. *arXiv preprint arXiv:2008.07142*, 2020.

[42] Dimos Makris, Katia Lida Kermanidis και Ioannis Karydis. *The greek audio dataset*. *Artificial Intelligence Applications and Innovations: AIAI 2014 Workshops: CoPA, MHDW, IIVC, and MT4BD, Rhodes, Greece, September 19-21, 2014. Proceedings 10*, σελίδες 165–173. Springer, 2014.

[43] Charilaos Papaioannou, Ioannis Valiantzas, Theodoros Giannakopoulos, Maximos Kaliakatsos-Papakostas και Alexandros Potamianos. *A Dataset for Greek Traditional and Folk Music: Lyra*. *arXiv preprint arXiv:2211.11479*, 2022.

[44] Meinard Müller, Yigitcan Özer, Michael Krause, Thomas Prätzlich και Jonathan Driedger. *Sync Toolbox: A Python package for efficient, robust, and accurate music synchronization*. *Journal of Open Source Software*, 6(64):3434, 2021.

[45] Dmitry Bogdanov, Nicolas Wack, Emilia Gómez Gutiérrez, Sankalp Gulati, Herrera Boyer, Oscar Mayor, Gerard Roma Trepat, Justin Salamon, José Ricardo Zapata González, Xavier Serra και others. *Essentia: An audio analysis library for music information retrieval*. *Britto A, Gouyon F, Dixon S, editors. 14th Conference of the International Society for Music Information Retrieval (ISMIR); 2013 Nov 4-8; Curitiba, Brazil.[place unknown]: ISMIR; 2013. p. 493-8*. International Society for Music Information Retrieval (ISMIR), 2013.

[46] Colin Raffel, Brian McFee, Eric J Humphrey, Justin Salamon, Oriol Nieto, Dawen Liang, Daniel PW Ellis και C Colin Raffel. *MIR_EVAL: A Transparent Implementation of Common MIR Metrics*. *ISMIR*, τόμος 10, σελίδα 2014, 2014.

[47] Romain Hennequin, Anis Khlif, Felix Voituret και Manuel Moussallam. *Spleeter: a fast and efficient music source separation tool with pre-trained models*. *Journal of Open Source Software*, 5(50):2154, 2020.

[48] MIDI Manufacturers Association και others. *The Complete MIDI 1.0 Detailed Specification*. The MIDI Manufacturers Association, Los Angeles, CA, 1996.

[49] Ana Davila, Jacinto Colan και Yasuhisa Hasegawa. *Comparison of fine-tuning strategies for transfer learning in medical image classification*. *Image and Vision Computing*, 146:105012, 2024.

[50] Noam Shazeer και Mitchell Stern. *Adafactor: Adaptive learning rates with sublinear memory cost*. *International Conference on Machine Learning*, σελίδες 4596–4604. PMLR, 2018.

[51] Yizhi Li, Ruibin Yuan, Ge Zhang, Yinghao Ma, Xingran Chen, Hanzhi Yin, Chenghao Xiao, Chenghua Lin, Anton Ragni, Emmanouil Benetos και others. *Mert: Acoustic music understanding model with large-scale self-supervised training*. *arXiv preprint arXiv:2306.00107*, 2023.

[52] Curtis Hawthorne, Andriy Stasyuk, Adam Roberts, Ian Simon, Cheng Zhi Anna Huang, Sander Dieleman, Erich Elsen, Jesse Engel και Douglas Eck. *Enabling factorized piano music modeling and generation with the MAESTRO dataset*. *arXiv preprint arXiv:1810.12247*, 2018.

[53] Qingyang Xi, Rachel M Bittner, Johan Pauwels, Xuzhou Ye και Juan Pablo Bello. *GuitarSet: A Dataset for Guitar Transcription*. *ISMIR*, σελίδες 453–460, 2018.

[54] Wikipedia contributors. *Indian classical music*. `https://en.wikipedia.org/wiki/Indian_classical_music`, 2025. Ημερομηνία πρόσβασης: 07-01-2025.

[55] Wikipedia contributors. *Arabic maqam*. `https://en.wikipedia.org/wiki/Arabic_maqam`, 2025. Ημερομηνία πρόσβασης: 07-01-2025.

[56] Nan Nan και Xiaohong Guan. *Common and distinct quantitative characteristics of Chinese and Western music in terms of modes, scales, degrees and melody variations*. *Journal of New Music Research*, 52(2-3):227–244, 2023.

[57] Yu Te Wu, Yin Jyun Luo, Tsung Ping Chen, I Wei, Jui Yang Hsu, Yi Chin Chuang, Li Su και others. *Omnizart: A general toolbox for automatic music transcription*. *arXiv preprint arXiv:2106.00497*, 2021.

[58] Yihao Wang, Yuwen Chen και Tianyi Peng. *Automatic Transcription of Ornamented Irish Flute Music*.

[59] Chih Wei Wu, Christian Dittmar, Carl Southall, Richard Vogl, Gerhard Widmer, Jason Hockman, Meinard Müller και Alexander Lerch. *A review of automatic drum transcription*. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(9):1457–1483, 2018.

[60] Majid A Al-Taee, Mohammad S Al-Rawi και Fadi M Al-Ghawanmeh. *Time-frequency analysis of the Arabian flute (Nay) tone applied to automatic music transcription*. *2008 IEEE/ACS International Conference on Computer Systems and Applications*, σελίδες 891–894. IEEE, 2008.

[61] Tim Beyer και Angela Dai. *End-to-end Piano Performance-MIDI to Score Conversion with Transformers*. *arXiv preprint arXiv:2410.00210*, 2024.

[62] Drew Edwards, Xavier Riley, Pedro Sarmento και Simon Dixon. *MIDI-to-Tab: Guitar tablature inference via masked language modeling*. *arXiv preprint arXiv:2408.05024*, 2024.

[63] Yinghao Ma, Anders Øland, Anton Ragni, Bleiz MacSen Del Sette, Charalampos Saitis, Chris Donahue, Chenghua Lin, Christos Plachouras, Emmanouil Benetos, Elona Shatri και others. *Foundation models for music: A survey*. *arXiv preprint arXiv:2408.14340*, 2024.

[64] Ruibin Yuan, Hanfeng Lin, Yi Wang, Zeyue Tian, Shangda Wu, Tianhao Shen, Ge Zhang, Yuhang Wu, Cong Liu, Ziya Zhou και others. *Chatmusician: Understanding and generating music intrinsically with llm*. *arXiv preprint arXiv:2402.16153*, 2024.

[65] Joshua P Gardner, Simon Durand, Daniel Stoller και Rachel M Bittner. *Llark: A multimodal foundation model for music*. 2023.

[66] Arthur L Samuel. *Some studies in machine learning using the game of checkers*. *IBM Journal of research and development*, 3(3):210–229, 1959.

[67] Zehra Cataltepe, Yusuf Yaslan και Abdullah Sonmez. *Music genre classification using MIDI and audio features*. *EURASIP Journal on Advances in Signal Processing*, 2007:1–8, 2007.

[68] Hareesh Bahuleyan. *Music genre classification using machine learning techniques*. *arXiv preprint arXiv:1804.01149*, 2018.

[69] Jan Wülfing και Martin A Riedmiller. *Unsupervised Learning of Local Features for Music Classification*. *ISMIR*, σελίδες 139–144, 2012.

[70] Matthew C McCallum. *Unsupervised learning of deep features for music segmentation*. *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, σελίδες 346–350. IEEE, 2019.

[71] Sylvain Le Groux και PFMJ Verschure. *Towards adaptive music generation by reinforcement learning of musical tension*. *proceedings of the 6th sound and music conference, Barcelona, Spain*, τόμος 134, 2010.

[72] Nan Jiang, Sheng Jin, Zhiyao Duan και Changshui Zhang. *Rl-duet: Online music accompaniment generation using deep reinforcement learning*. *Proceedings of the AAAI conference on artificial intelligence*, τόμος 34, σελίδες 710–718, 2020.

[73] Jean Pierre Briot, Gaëtan Hadjeres και François David Pachet. *Deep learning techniques for music generation-a survey*. *arXiv preprint arXiv:1709.01620*, 2017.

[74] Thierry Bertin-Mahieux, Daniel PW Ellis, Brian Whitman και Paul Lamere. *The million song dataset*. *Ismir*, τόμος 2, σελίδα 10, 2011.

[75] Matt McVicar, Raúl Santos-Rodríguez, Yizhao Ni και Tijl De Bie. *Automatic chord estimation from audio: A review of the state of the art*. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(2):556–575, 2014.

[76] David Temperley. *What's key for key? The Krumhansl-Schmuckler key-finding algorithm reconsidered*. *Music Perception*, 17(1):65–100, 1999.

[77] Daniel PW Ellis. *Beat tracking by dynamic programming*. *Journal of New Music Research*, 36(1):51–60, 2007.

[78] Michael Good. *MusicXML for notation and analysis*. *The virtual score: representation, retrieval, restoration*, 12(113-124):160, 2001.

[79] Chris Walshaw. *The ABC music standard*. `http://abcnotation.com/`, 2011. Ημερομηνία πρόσβασης: 07-01-2025.

[80] David Brian Huron. *The humdrum toolkit: Reference manual*. Center for Computer Assisted Research in the Humanities, 1994.

[81] Yu Siang Huang και Yi Hsuan Yang. *Pop music transformer: Beat-based modeling and generation of expressive pop piano compositions*. *Proceedings of the 28th ACM international conference on multimedia*, σελίδες 1180–1188, 2020.

[82] Yi Ren, Jinzheng He, Xu Tan, Tao Qin, Zhou Zhao και Tie Yan Liu. *Popmag: Pop music accompaniment generation*. *Proceedings of the 28th ACM international conference on multimedia*, σελίδες 1198–1206, 2020.

[83] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg και Oriol Nieto. *librosa: Audio and music signal analysis in python*. *SciPy*, σελίδες 18–24, 2015.

[84] M. Müller. *Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications*. Springer, 2015.

[85] J. B. Allen και L. R. Rabiner. *A unified approach to short-time Fourier analysis and synthesis*. *Proceedings of the IEEE*, 65(11):1558–1564, 1977.

[86] J. C. Brown. *Calculation of a constant Q spectral transform*. *Journal of the Acoustical Society of America*, 89(1):425–434, 1991.

[87] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. Weiss και K. Wilson. *CNN Architectures for Large-Scale Audio Classification*. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, σελίδες 131–135, 2017.

[88] J. Cramer, H. Wu, J. Salamon και J. P. Bello. *Look, Listen, and Learn More: Design Choices for Deep Audio Embeddings*. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, σελίδες 3852–3856, 2019.

[89] A. Guzhov, F. Elezi, R. M. Ferrer και B. Ommer. *AudioCLIP: Extending CLIP to Audio*. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, σελίδες 976–980, 2022.

[90] Lejaren Arthur Hiller και Leonard M Isaacson. *Experimental Music; Composition with an electronic computer*. Greenwood Publishing Group Inc., 1979.

[91] Gaëtan Hadjeres, François Pachet και Frank Nielsen. *Deepbach: a steerable model for bach chorales generation*. *International conference on machine learning*, σελίδες 1362–1371. PMLR, 2017.

[92] Hao Wen Dong, Wen Yi Hsiao, Li Chia Yang και Yi Hsuan Yang. *Musegan: Multi-track sequential generative adversarial networks for symbolic music generation and accompaniment*. *Proceedings of the AAAI conference on artificial intelligence*, τόμος 32, 2018.

[93] Aashiq Muhamed, Liang Li, Xingjian Shi, Suri Yaddanapudi, Wayne Chi, Dylan Jackson, Rahul Suresh, Zachary C Lipton και Alex J Smola. *Symbolic music generation with transformer-gans*. *Proceedings of the AAAI conference on artificial intelligence*, τόμος 35, σελίδες 408–417, 2021.

[94] Andrea Agostinelli, Timo I Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi και others. *Musiclm: Generating music from text*. *arXiv preprint arXiv:2301.11325*, 2023.

[95] Fatemeh Jamshidi, Gary Pike, Amit Das και Richard Chapman. *Machine learning techniques in automatic music transcription: A systematic survey*. *arXiv preprint arXiv:2406.15249*, 2024.

[96] Sebastian Ewert και Mark Sandler. *Piano transcription in the studio using an extensible alternating directions framework*. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(11):1983–1997, 2016.

[97] Sebastian Böck και Markus Schedl. *Polyphonic piano note transcription with recurrent neural networks*. *2012 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, σελίδες 121–124. IEEE, 2012.

[98] Siddharth Sigtia, Emmanouil Benetos και Simon Dixon. *An end-to-end neural network for polyphonic piano music transcription*. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(5):927–939, 2016.

[99] Valentin Emiya, Roland Badeau και Bertrand David. *Multipitch estimation of piano sounds using a new probabilistic spectral smoothness principle*. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(6):1643–1654, 2009.

[100] Valentin Emiya, Nancy Bertin, Bertrand David και Roland Badeau. *MAPS-A piano database for multipitch estimation and automatic transcription of music*. 2010.

[101] Colin Raffel. *Learning-based methods for comparing sequences, with applications to audio-to-midi alignment and matching*. Columbia University, 2016.

[102] Bochen Li, Xinzhao Liu, Karthik Dinesh, Zhiyao Duan και Gaurav Sharma. *Creating a multitrack classical music performance dataset for multimodal music analysis: Challenges, insights, and applications*. *IEEE Transactions on Multimedia*, 21(2):522–535, 2018.

[103] Wei Tsung Lu, Li Su και others. *Transferring the Style of Homophonic Music Using Recurrent Neural Networks and Autoregressive Model*. *ISMIR*, σελίδες 740–746, 2018.

[104] Aaron van den Oord. *WaveNet: A Generative Model for Raw Audio*. *arXiv preprint arXiv:1609.03499*, 2016.

[105] *Reduction (music) — Wikipedia, The Free Encyclopedia*. `https://en.wikipedia.org/wiki/Reduction_(music)`, 2025. Ημερομηνία πρόσβασης: 07-01-2025.

[106] Zhesong Yu, Xiaoshuo Xu, Xiaoou Chen και Deshun Yang. *Learning a representation for cover song identification using convolutional neural network*. *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, σελίδες 541–545. IEEE, 2020.

[107] Xingjian Du, Ke Chen, Zijie Wang, Bilei Zhu και Zejun Ma. *Bytecover2: Towards dimensionality reduction of latent embedding for efficient cover song identification*. *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, σελίδες 616–620. IEEE, 2022.

[108] Shichao Hu, Bin Zhang, Jinhong Lu, Yiliang Jiang, Wucheng Wang, Lingcheng Kong, Weifeng Zhao και Tao Jiang. *WideResNet with Joint Representation Learning and Data Augmentation for Cover Song Identification*. *Interspeech*, σελίδες 4187–4191, 2022.

[109] R Oguz Araz, Xavier Serra και Dmitry Bogdanov. *Discogs-VI: A musical version identification dataset based on public editorial metadata*. *arXiv preprint arXiv:2410.17400*, 2024.

[110] Stixoi. *Stixoi*. `https://stixoi.info`. Ημερομηνία πρόσβασης:07-01-2025.

[111] Nikos Ordoulidis. *The greek laiko (popular) rhythms: Some problematic issues*. *Proceedings 2nd Annual International Conference on Visual and Performing Arts*, 2011.

[112] Lampros Liavas. *The Greek Song from 1821 to the decade of 1950*. *Athens: Emporiki Bank of Greece*, 2009.

[113] A. Sideras. *The sung poetry*. *Musicology*, 3:89–106, 1985.

[114] Alandi Perna. *Brave Noise—The History of Alternative Rock Guitar*. *Guitar World*, 1995.

[115] Xavier Riley, Drew Edwards και Simon Dixon. *High resolution guitar transcription via domain adaptation*. *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, σελίδες 1051–1055. IEEE, 2024.

[116] Matthias Mauch και Simon Dixon. *pYIN: A fundamental frequency estimator using probabilistic threshold distributions*. *2014 ieee international conference on acoustics, speech and signal processing (icassp)*, σελίδες 659–663. IEEE, 2014.

[117] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell και others. *Language models are few-shot learners*. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[118] *Chromacoverid*. `https://github.com/albincorreya/ChromaCoverId`. Ημερομηνία πρόσβασης: 07-01-2025.