
ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ & ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΔΠΜΣ ΕΠΙΣΤΗΜΗ ΔΕΔΟΜΕΝΩΝ & ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

# ΣΥΝΘΕΣΗ ΦΩΝΗΣ ΑΠΟ ΚΕΙΜΕΝΟ ΣΤΑ ΕΛΛΗΝΙΚΑ

Διπλωματική Εργασία

ΑΓΓΕΛΑΚΟΠΟΥΛΟΣ ΧΑΡΑΛΑΜΠΟΣ

**Επιβλέποντες:**

Καθ. ΠΕΤΡΟΣ ΜΑΡΑΓΚΟΣ
Δρ. ΑΘΑΝΑΣΙΟΣ ΚΑΤΣΑΜΑΝΗΣ

ΑΘΗΝΑ, Οκτώβριος 2022

Στους Γονείς μου

# Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέποντα Καθηγητή της Διπλωματικής μου εργασίας κύριο Πέτρο Μαραγκό για την εμπιστοσύνη και το θετικό του ενδιαφέρον να εκπονήσω την εργασία μου μαζί του. Επίσης ευχαριστώ πολύ τον συν-επιβλέποντά μου κύριο Νάσσο Κατσαμάνη που μου πρότεινε να ασχοληθώ με ένα πολύ ενδιαφέρον και σύγχρονο θέμα, καθώς και για την καθοδήγησή του στις διαδικτυακές συναντήσεις που είχαμε κατά τη διάρκεια συγγραφής της εργασίας. Ευχαριστώ επίσης και τον Ευθύμη Γεωργίου για τη βοήθειά του πάνω σε τεχνικά θέματα που προέκυπταν κατά καιρούς. Τέλος θα ήθελα να ευχαριστήσω την οικογένειά μου και κυρίως τους γονείς μου Παναγιώτη και Βασιλική, οι οποίοι με στηρίζουν σε όλες μου τις επιλογές.

# Περίληψη

Η παρούσα διπλωματική εργασία πραγματεύεται το πρόβλημα της σύνθεσης φωνής από κείμενο (text-to-speech) στα ελληνικά. Η σύνθεση φωνής από κείμενο είναι η διαδικασία σύμφωνα με την οποία ένα σύστημα π.χ. ένας υπολογιστής, μετατρέπει ένα κείμενο σε ηχητικό δείγμα που περιλαμβάνει την αντίστοιχη ομιλία. Τα τελευταία χρόνια αναπτύσσονται αρκετές εφαρμογές που στηρίζονται στη σύνθεσης φωνής από κείμενο και χρησιμοποιούνται ακόμα και σε εταιρικά περιβάλλοντα για να βελτιώσουν την αλληλεπίδραση με τους χρήστες και τους πελάτες (π.χ conversational assistants). Επίσης ένα πολύ γνωστό παράδειγμα που βασίζεται στη σύνθεση φωνής είναι το Google Translate μέσω του οποίου μπορεί κανείς να πληκτρολογήσει ένα κείμενο και να ακούσει τον τρόπο με τον οποίο εκφωνείται σε μια συγκεκριμένη γλώσσα. Επιπλέον η υπηρεσία Polly της Amazon μπορεί να χρησιμοποιηθεί για ανάπτυξη εφαρμογών που στηρίζονται στη μετατροπή ενός κειμένου σε ανθρώπινη ομιλία σε αρκετές γλώσσες. Τέλος, τα λεγόμενα audio-books και τα avatars βασίζονται επίσης στη σύνθεση φωνής από κείμενο. Για τέτοιου είδους εφαρμογές, πέραν της φυσικότητας απαιτείται και η ενσωμάτωση χαρακτηριστικών όπως το συναίσθημα και η προσωδία στην παραγόμενη ομιλία. Τα παραπάνω παρουσιάζονται στο πρώτο κεφάλαιο της εργασίας όπου γίνεται μια εισαγωγή στο πρόβλημα που μελετάμε. Αναφέρουμε τις βασικότερες προσεγγίσεις πάνω στο πρόβλημα της σύνθεσης φωνής, οι οποίες είναι η συναθροιστική, η στατιστική-παραμετρική και το neural TTS, εστιάζοντας κυρίως στην τελευταία και πιο σύγχρονη προσέγγιση. Ταυτόχρονα γίνεται αναφορά στον τρόπο εξαγωγής ακουστικών χαρακτηριστικών (mel spectrogram), που χρησιμοποιούνται για την εκπαίδευση και τη συμπερασματολογία ενός συστήματος παραγωγής ομιλίας. Στο δεύτερο κεφάλαιο αναλύουμε τις βασικότερες state-of-the-art αρχιτεκτονικές που στηρίζονται στη χρήση βαθιών νευρωνικών δικτύων. Ως επί το πλείστον τα μοντέλα αυτά αποτελούνται από δύο ξεχωριστά τμήματα. Το πρώτο τμήμα δέχεται το κείμενο και παράγει ενδιάμεσα χαρακτηριστικά όπως το φασματογράφημα στην κλίμακα mel. Στην κατηγορία αυτή ανήκουν μοντέλα όπως το Tacotron, το Tacotron2, ο Transformer TTS κ.ά. Το δεύτερο τμήμα αποτελείται από ένα μοντέλο vocoder όπως το WaveNet ή το WaveGlow, προκειμένου να μετατρέψει τα ενδιάμεσα χαρακτηριστικά στην τελική κυματομορφή ήχου. Τέλος παρουσιάζεται το μοντέλο WaveGrad2 που μπορεί να παράγει απευθείας συνθετική ομιλία από ένα κείμενο χωρίς την εξαγωγή ενδιάμεσων χαρακτηριστικών. Όλα τα παραπάνω μοντέλα πετυχαίνουν πολύ ικανοποιητικά αποτελέσματα με συνθετική φωνή που αγγίζει τα ανθρώπινα επίπεδα, αξιολογώντας τα σύμφωνα με την κλίμακα MOS (Mean Opinion Score). Παρ' όλ' αυτά στα αρνητικά τους συγκαταλέγεται ο μεγάλος χρόνος αλλά και η μεγάλη υπολογιστική ισχύς που απαιτείται κατά την εκπαίδευσή τους. Στο τρίτο κεφάλαιο παρουσιάζουμε ορισμένα ειδικά θέματα στη σύνθεση φωνής από κείμενο. Όπως αναφέραμε, σε τέτοιου είδους συστήματα πέρα από τη φυσικότητα είναι σημαντική η προσθήκη συναισθήματος και προσωδίας στην τελική ομιλία. Ένα μοντέλο που επιτυγχάνει τα παραπάνω είναι το Global Style Tokens, μέσω του οποίου μπορούν να ρυθμιστούν χαρακτηριστικά του παραγόμενου ήχου όπως το στυλ, ο τόνος η ταχύτητα κ.ά. Επιπλέον για γλώσσες που δεν υπάρχουν αρκετά διαθέσιμα δεδομένα για εκπαίδευση ενός μοντέλου (low re-

source languages), μπορούν να αξιοποιηθούν μέθοδοι όπως το LRSpeech ή η επαύξηση δεδομένων (data augmentation) για τη δημιουργία νέων συνθετικών δειγμάτων τα οποία με τη σειρά τους αξιοποιούνται για εκπαίδευση ενός συστήματος TTS στη γλώσσα και το στυλ που επιθυμούμε. Στο τελευταίο κεφάλαιο γίνεται μελέτη του προβλήματος της σύνθεσης φωνής στα ελληνικά. Αρχικά εκπαιδεύουμε τα μοντέλα Tacotron2 και WaveGlow στην ισπανική γλώσσα όπου υπάρχουν αρκετά διαθέσιμα δεδομένα ηχογραφήσεων από έναν ομιλητή. Έπειτα χρησιμοποιούμε την τεχνική της μεταφοράς μάθησης για να εκπαιδεύσουμε τα συγκεκριμένα μοντέλα στην ελληνική γλώσσα, όπου τα διαθέσιμα δεδομένα είναι λιγότερα. Επειδή η ποιότητα των παραγόμενων δειγμάτων δεν ήταν αρκετά ικανοποιητική, προχωρήσαμε σε συλλογή νέων δεδομένων από μια μόνο ομιλήτρια στην ελληνική γλώσσα συνολικής διάρκειας ηχογραφήσεων περίπου 19.5 ώρες. Τα καλύτερα αποτελέσματα προέκυψαν με χρήση της μεταφοράς μάθησης και με την αξιοποίηση του νέου συνόλου δεδομένων στα ελληνικά. Τα παραγόμενα ηχητικά δείγματα αξιολογούνται ως προς τη φυσικότητά τους στην κλίμακα MOS μέσω ερωτηματολογίου. Από τα αποτελέσματα που προκύπτουν, διαπιστώνεται ότι η ποιότητα της συνθετικής φωνής από τα πειράματά μας στα ελληνικά είναι σχετικά καλή, εντούτοις υπάρχει ακόμα περιθώριο βελτίωσης προκειμένου η παραγόμενη φωνή να είναι πιο κοντά στα ανθρώπινα επίπεδα, ώστε να μπορεί να χρησιμοποιηθεί σε μια εφαρμογή. Κλείνοντας, παρουσιάζουμε τα τελικά συμπεράσματα αλλά και ορισμένες μελλοντικές επεκτάσεις όσον αφορά το πρόβλημα της σύνθεσης φωνής από κείμενο στα ελληνικά.

**Λέξεις Κλειδιά** — Σύνθεση φωνής από κείμενο, Ελληνικά, Tacotron2, WaveGlow, Κλίμακα MOS, Νευρωνικά δίκτυα, Βαθιά μάθηση

# Abstract

This thesis deals with the problem of text-to-speech synthesis in greek. Text-to-speech (TTS) synthesis is the process whereby a system e.g. a computer, converts a text into an audio sample that includes the corresponding speech. In recent years, several applications based on text-to-speech synthesis have been developed and are even used in corporate environments to improve interaction with users and customers (e.g. converastional assistants). Also one very well-known example that is based on text-to-speech is Google Translate through which one can type a text and hear how it is pronounced in a certain language. In addition, Amazon's Polly service can be used to develop applications that rely on the conversion of text to human speech in several languages. Finally, so-called audio-books and avatars are also based on text-to-speech synthesis. For such applications, in addition to naturalness, the integration of features such as emotion and prosody into the produced speech is also required. The above are presented in the first chapter of this thesis where we make a general introduction to the problem of text-to-speech. We report the most basic approaches to the problem of speech synthesis, which are concatenative, statistical-parametric and neural TTS, focusing mainly on the last and most modern approach. At the same time, reference is made on how to extract acoustic features (mel spectrogram), which are used for the training and inference of a speech synthesis system. In the second chapter we analyze the most basic state-of-the-art architectures based on the use of deep neural networks. Mostly these models consist of two separate parts. The first part accepts the text as input and produces intermediate features such as the spectrogram at the mel scale. This category includes models such as Tacotron, Tacotron2, Transformer TTS and more. The second part consists of a vocoder model such as WaveNet or WaveGlow, in order to convert the intermediate features into the final audio waveform. Finally, we present the WaveGrad2 model which can directly generate synthetic speech from a text without extracting intermediate features. All the aforementioned models achieve very satisfactory results with synthetic voice reaching human levels, evaluating them according to the MOS (Mean Opinion Score) scale. Nevertheless, their drawbacks include the long time and computing resources required during their training. In the third chapter we present some special topics in text-to-speech synthesis. As we have mentioned, in such systems, in addition to naturalness, it is important to add emotion and prosody to the final speech. A model that achieves that is Global Style Tokens, through which characteristics of the produced sound such as style, pitch, speed, etc. can be set. In addition, for languages where there is not enough data available to train a model (low resource languages), methods such as LRSpeech or data augmentation can be used to create new synthetic samples which in turn are used to train a TTS system in the language and style we want. In the last chapter, we study the problem of speech synthesis in greek. We first train the Tacotron2 and WaveGlow models on the Spanish language where there is enough available recording data from a single speaker. Then we use the transfer learning technique to train these models in the greek language, where the available data is less. Because the

quality of the produced samples was not satisfactory enough, we proceeded to collect new data from a single speaker in the greek language with a total duration of approximately 19.5 hours of recordings. The best results were obtained using transfer learning and leveraging the new data set in greek. The naturalness of the produced audio samples is evaluated using the MOS scale through a questionnaire. From the results we had, it can be seen that the quality of the synthesized speech from our experiments in greek is relatively good, however there is still room for improvement in order for the generated speech to reach human levels and be incorpotated in an application. Finally, we present our conclusions and future work regarding the task of text-to-speech synthesis in greek.

**Keywords** — Text to speech synthesis, Greek, Tacotron2, WaveGlow, Mean Opinion Score, Neural Networks, Deep Learning

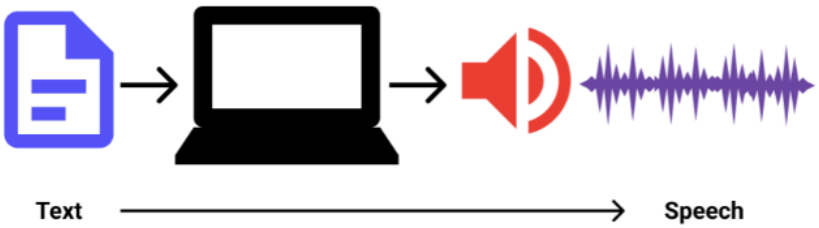
# Περιεχόμενα

# Κεφάλαιο 1

# Εισαγωγή

## 1.1 Το πρόβλημα της σύνθεσης φωνής από κείμενο

Στο παρόν κεφάλαιο κάνουμε μια εισαγωγή στο πρόβλημα της σύνθεσης φωνής από κείμενο. Με τον όρο σύνθεση φωνής από κείμενο ή αλλιώς text-to-speech (TTS) εννοούμε την μετατροπή ενός κειμένου στο αντίστοιχο ηχητικό δείγμα. Παραδείγματος χάριν, αν έχουμε την πρόταση: «Σήμερα έχει ωραία μέρα», στόχος ενός συστήματος TTS είναι η τεχνητή παραγωγή ομιλίας που αντιστοιχεί στη συγκεκριμένη πρόταση. Το ηχητικό δείγμα που παράγεται θα πρέπει να είναι όσο το δυνατόν πιο φυσικό, δηλαδή να ακούγεται σα να έχει προκύψει από ανθρώπινη φωνή. Στο Σχήμα 1.1 παρουσιάζουμε τη διαδικασία παραγωγής ενός τέτοιου ηχητικού δείγματος, όπου αριστερά έχουμε το κείμενο που θέλουμε να μετατρέψουμε και δεξιά την παραγόμενη κυματομορφή ήχου.



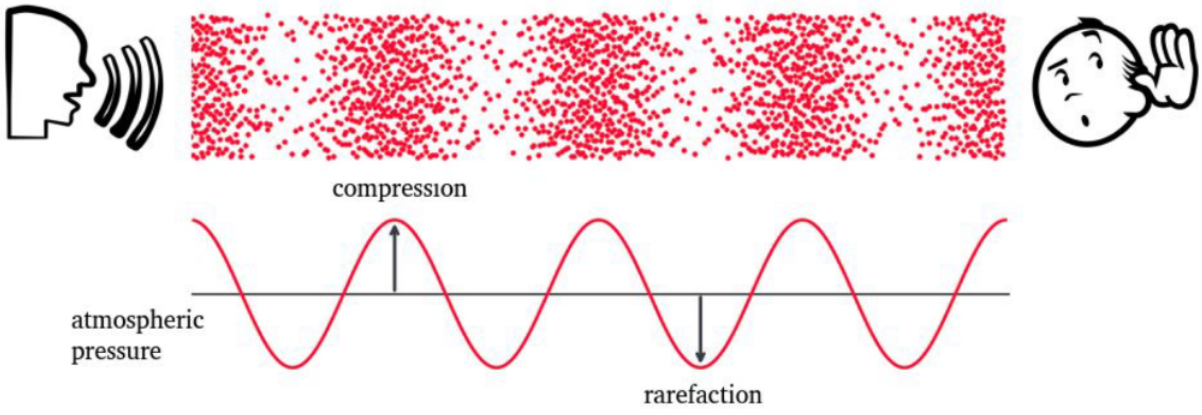

Σχήμα 1.1: Σύστημα σύνθεσης φωνής από κείμενο.

Δεν είναι λίγες οι εφαρμογές που στηρίζονται σε τέτοιου είδους συστήματα. Για παράδειγμα πολλές επιχειρήσεις τα τελευταία χρόνια χρησιμοποιούν κάποιο σύστημα σύνθεσης φωνής για να βελτιώσουν την αλληλεπίδραση με τους πελάτες τους. Τα λεγόμενα chatbots και οι conversational assistants είναι χαρακτηριστικά παραδείγματα που χρησιμοποιούν επιχειρήσεις, όπως τράπεζες, online shops κτλ., ώστε οι πελάτες τους να λαμβάνουν καλύτερη και ταχύτερη εξυπηρέτηση. Μια εφαρμογή TTS μπορεί για παράδειγμα να χρησιμοποιείται σε ένα τηλεφωνικό κέντρο ώστε να δίνει real-time πληροφορίες στους πελάτες μέσω μιας συνθετικής φωνής που μοιάζει με ανθρώπινη. Επίσης ένα χαρακτηριστικό παράδειγμα είναι η δυνατότητα που προσφέρει το Google Translate για εκφώνηση του κειμένου που θέλουμε να μεταφράσουμε. Έτσι μπορούμε εύκολα να ακούσουμε τον τρόπο με τον οποίο προφέρεται μία λέξη σε μια ξένη γλώσσα. Επιπλέον μέσω ενός συστήματος TTS μπορούν να επωφεληθούν και άτομα με μαθησιακές δυσκολίες ή δυσκολίες στην όραση και την ανάγνωση, δίνοντάς τους πρόσβαση σε μεγαλύτερο εκπαιδευτικό περιεχόμενο. Συγκεκριμένα τα άτομα αυτά μπορούν να ακούν, χωρίς να χρειάζεται να βλέπουν και να διαβάζουν το

κείμενο που φαίνεται σε μια οθόνη ή ένα βιβλίο. Τέλος τα συστήματα text-to-speech μπορούν να μετατρέψουν ψηφιακό περιεχόμενο, όπως άρθρα, blogs κτλ. σε ακουστικό (π.χ. audio books), επεκτείνοντας έτσι τις εφαρμογές της σύνθεσης φωνής στην ενημέρωση και τη ψυχαγωγία. Τα λεγόμενα talking avatars που εμφανίζονται σε ηλεκτρονικά παιχνίδια και κινούμενα σχέδια είναι άλλη μία περιοχή που βασίζεται σε εφαρμογές σύνθεσης φωνής από κείμενο. Λαμβάνοντας υπόψιν όλα τα παραπάνω κατανοούμε τη σημασία και το ενδιαφέρον για την ανάπτυξη τέτοιων εφαρμογών και γενικότερα για τη μελέτη της συγκεκριμένης περιοχής. Όπως θα δούμε και στη συνέχεια, τα περισσότερα συστήματα TTS αποτελούνται συνήθως από δύο επιμέρους τμήματα. Το πρώτο αφορά την επεξεργασία και μετατροπή του κειμένου σε ορισμένα ενδιάμεσα χαρακτηριστικά, όπως είναι το φασματογράφημα στην κλίμακα mel. Το δεύτερο μέρος αφορά την μετατροπή αυτών των χαρακτηριστικών στην αντίστοιχη κυματομορφή ήχου και ονομάζεται vocoder. Προτού όμως εμβαθύνουμε στα επιμέρους αυτά τμήματα είναι χρήσιμο να δούμε ορισμένα βασικά στοιχεία για την βαθύτερη κατανόηση του προβλήματος.

## 1.2 Το ηχητικό σήμα

Το βασικό στοιχείο που πρέπει να κατανοήσουμε είναι ο τρόπος με τον οποίο αντιλαμβανόμαστε και μελετάμε τον ήχο. Ο ήχος είναι ένα μηχανικό κύμα που προκαλείται από τη δόνηση και ταλάντωση των μορίων ενός μέσου, στο οποίο μεταδίδεται. Όσον αφορά τον ήχο που παράγεται από την ανθρώπινη ομιλία, το μέσο που χρησιμοποιείται για τη διάδοσή του είναι κατά βάση ο αέρας. Η ταλάντωση των μορίων του αέρα επιφέρει μεταβολές στην ατμοσφαιρική πίεση, οι οποίες μπορούν να αναπαρασταθούν μέσω μιας κυματομορφής ήχου. Στο Σχήμα 1.2 παρουσιάζεται μια απλή κυματομορφή ήχου. Παρατηρούμε ότι στα σημεία όπου υπάρχει μεγαλύτερη συγκέντρωση στα μόρια του αέρα (compression), η πίεση λαμβάνει την υψηλότερη τιμή πάνω από τη συνήθη ατμοσφαιρική πίεση (οριζόντια γραμμή), ενώ στα σημεία όπου τα μόρια του αέρα έχουν μεγαλύτερη απόσταση μεταξύ τους (rarefaction), η πίεση βρίσκεται στα χαμηλότερα επίπεδα σε σχέση με τη συνήθη ατμοσφαιρική πίεση.



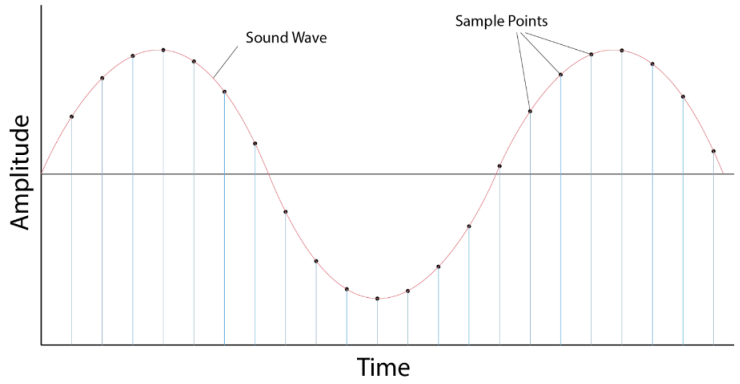

Σχήμα 1.2: Αναπαράσταση μιας κυματομορφής ήχου. Το κάτω γράφημα αντικατοπτρίζει τη μεταβολή στην ατμοσφαιρική πίεση, η οποία προκαλείται από την ταλάντωση των μορίων του αέρα.

Ένα απλό ηχητικό σήμα μπορεί να αναπαρασταθεί από μια ημιτονοειδή συνάρτηση της μορφής:

$$y(t) = A\sin(2\pi ft + \phi), \tag{1.2.1}$$

όπου το $y$ δηλώνει την πίεση σε σχέση με το χρόνο t, $A$ είναι το πλάτος (amplitude), δηλαδή η μέγιστη κατ' απόλυτη τιμή που λαμβάνει η πίεση, $f$ είναι η συχνότητα του κύματος και $\phi$ η φάση, μέσω της οποίας μπορούμε να δούμε την τιμή της πίεσης τη χρονική στιγμή $t = 0$, αφού $y(0) = A\sin\phi$. Η συχνότητα του κύματος μετριέται σε Hertz και ισούται με το αντίστροφο της περιόδου, δηλαδή του ελάχιστου χρόνου που απαιτείται για να εκτελεστεί μια επανάληψη του φαινομένου (π.χ. το χρονικό διάστημα ανάμεσα σε δύο διαδοχικές μέγιστες τιμές που λαμβάνει η πίεση). Είναι λογικό ότι όσο μεγαλύτερο πλάτος έχουμε τόσο πιο έντονα αντιλαμβανόμαστε τον παραγόμενο ήχο αφού τα μόρια του αέρα ταλαντώνονται περισσότερο άρα η πίεση που ασκείται αυξάνεται.

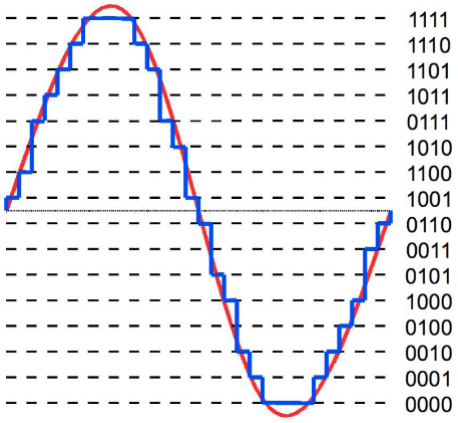
Το ηχητικό σήμα που μελετάμε εμπίπτει στην κατηγορία των αναλογικών σημάτων, αφού λαμβάνει συνεχείς τιμές τόσο στο πεδίο του χρόνου όσο και στο πλάτος. Αυτό καθιστά αδύνατη την αποθήκευση και επεξεργασία του από ψηφιακές μηχανές όπως είναι ο υπολογιστής, αφού η μνήμη που διαθέτουμε είναι περιορισμένη. Συνεπώς το αναλογικό σήμα χρειάζεται να μετατραπεί σε ψηφιακό σήμα (analog-to-digital conversion). Το ψηφιακό σήμα αποτελείται από πεπερασμένες το πλήθος διακριτές τιμές και επομένως μπορεί να αποθηκευτεί στη μνήμη ενός υπολογιστή. Η διαδικασία μετατροπής αναλογικού σε ψηφιακό σήμα αποτελείται από δύο βήματα, τη δειγματοληψία (sampling) και τον κβαντισμό (quantization). Η δειγματοληψία όπως δηλώνει και η λέξη είναι η διαδικασία κατά την οποία λαμβάνουμε δείγματα από το ηχητικό σήμα ανά σταθερά χρονικά διαστήματα, όπως φαίνεται και στο Σχήμα 1.3. Το πλήθος των δειγμάτων που λαμβάνουμε σε διάστημα ενός δευτερολέπτου ονομάζεται ρυθμός δειγματοληψίας (sampling rate). Στα CD ο ρυθμός δειγματοληψίας είναι συνήθως στα 44.1 kHz, ενώ στο μικρόφωνο συνήθως χρησιμοποιείται ρυθμός δειγματοληψίας 16kHz.




Σχήμα 1.3: Δειγματοληψία ενός ηχητικού σήματος.

Οι τιμές που λαμβάνουμε μέσω της δειγματοληψίας αποθηκεύονται στη μνήμη του υπολογιστή ως ακέραιοι και αναπαρίστανται συνήθως από 8 ή 16 bit. Συνεπώς αν ο ρυθμός δειγματοληψίας μας είναι στα 16kHz, τότε για ένα ηχητικό σήμα διάρκειας ενός δευτερολέπτου αποθηκεύονται 16000 τιμές. Ένας ακέραιος που αναπαρίσταται από 16 bit μπορεί να λάβει μία εκ των $2^{16} = 65536$ τιμές ακεραίων στο διάστημα $[-32768, 32767]$. Κατά τον κβαντισμό κάθε πραγματική τιμή που έχουμε λάβει ως δείγμα για το πλάτος αναπαρίσταται από τον «κοντινότερο» ακέραιο που βρίσκεται στο παραπάνω διάστημα. Για παράδειγμα η πραγματική τιμή 5607.76 θα αντιπροσωπεύεται από τον ακέραιο 5608 σύμφωνα με αυτή τη μέθοδο. Η μέθοδος quantization παρουσιάζεται στο Σχήμα 1.4, όπου για χάριν απλότητας κάθε τιμή του πλάτους αναπαρίσταται χρησιμοποιώντας 4 bit.

Συμπεραίνουμε λοιπόν ότι είναι αναγκαία η αποθήκευση των δειγμάτων μιας κυματομορφής ήχου με τρόπο αποδοτικό, αν σκεφτούμε ότι για ρυθμό δειγματοληψίας 16kHz και για ένα λεπτό ήχου χρειάζονται περίπου $60 \cdot 16 \cdot 16000 \cdot 0.000000125 = 1.92$ MB.

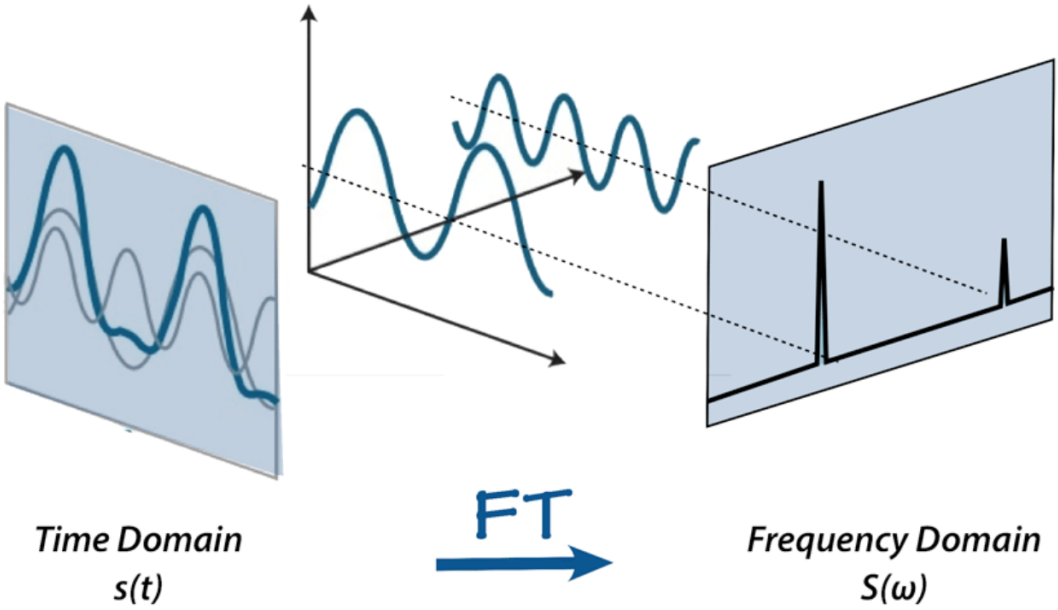



Σχήμα 1.4: Η διαδικασία κβαντισμού (quantization) ενός ηχητικού σήματος.

## 1.3 Επεξεργασία του σήματος ήχου

Αφού μετατρέψουμε το ηχητικό σήμα από αναλογικό σε ψηφιακό μέσω της δειγματοληψίας και του κβαντισμού, σειρά έχει η εξαγωγή κατάλληλων χαρακτηριστικών από αυτό. Για το πρόβλημα της σύνθεσης φωνής από κείμενο και γενικότερα για άλλες εφαρμογές που στηρίζονται στην επεξεργασία του ηχητικού σήματος όπως η αναγνώριση φωνής (speech recognition), είναι σημαντική η χρήση χαρακτηριστικών που περιέχουν ουσιαστική πληροφορία από το ηχητικό σήμα. Ως γνωστόν μια σύνθετη κυματομορφή προκύπτει από το άθροισμα επιμέρους ημιτονοειδών κυμάτων καθένα από τα οποία έχει και διαφορετική συχνότητα. Η συχνότητα ενός κύματος είναι από τα σημαντικότερα μεγέθη που το αντιπροσωπεύουν. Συνεπώς θα πρέπει να γνωρίζουμε τη συνεισφορά κάθε συχνότητας στο τελικό κύμα. Το φάσμα ή αλλιώς spectrum είναι ένας τρόπος αναπαράστασης των συχνοτήτων που απαρτίζουν το τελικό κύμα. Όπως φαίνεται στο Σχήμα 1.5 (δεξιά) στον άξονα x έχουμε τη συχνότητα και στον άξονα y μετράμε τη συνεισφορά κάθε συχνότητας στο κύμα. Παρατηρούμε δύο συχνότητες με σημαντική συνεισφορά στο κύμα, όπου η πρώτη (μικρότερη) συχνότητα, έχει και τη μεγαλύτερη συνεισφορά. Υπολογίζοντας το spectrum μεταβαίνουμε από το πεδίο του χρόνου (time domain) που βρίσκεται η κυματομορφή ήχου, στο πεδίο των συχνοτήτων (frequency domain).

Ο υπολογισμός του φάσματος συχνοτήτων προκύπτει από μία πολύ γνωστή μέθοδο στα μαθηματικά και την επεξεργασία σήματος που ονομάζεται μετασχηματισμός Fourier. Προτού περιγράψουμε τη βασική ιδέα του μετασχηματισμού Fourier, παραθέτουμε ορισμένα αρχικά βήματα που ακολουθούνται κατά την επεξεργασία της κυματομορφής ήχου. Σε πρώτη φάση το ηχητικό σήμα χωρίζεται σε ορισμένα frames σταθερού μήκους, καθένα από τα οποία αποτελείται από μικρό αριθμό δειγμάτων, όπου ο αριθμός αυτός είναι συνήθως κάποια δύναμη του 2 (π.χ. 1024, 2048). Τα χαρακτηριστικά που εξάγουμε δεν προκύπτουν από ολόκληρο το κύμα αλλά από κάθε frame ξεχωριστά. Τα frames λοιπόν μπορούν να θεωρηθούν ως ένα μικρό κομμάτι του ηχητικού σήματος που περιέχει πληροφορία, π.χ. για κάποιο φώνημα. Αν σκεφτούμε ότι ένα δείγμα για ρυθμό δειγματοληψίας 16 kHZ έχει διάρκεια $\frac{1}{16000} = 0.0625$ ms και η ελάχιστη διάρκεια ήχου που μπορεί




Σχήμα 1.5: Αναπαράσταση (δεξιά) του φάσματος των συχνοτήτων που συνεισφέρουν περισσότερο στη σύνθετη κυματομορφή ήχου (αριστερά). Μεταβολή από το πεδίο του χρόνου στο πεδίο των συχνοτήτων μέσω του μετασχηματισμού Fourier (FT).

να αντιληφθεί ο άνθρωπος είναι περίπου στα 10 ms, τότε το κάθε frame είναι ένα μέρος του ηχητικού σήματος που αποτελείται από λίγα δείγματα αλλά μπορεί να γίνει αντιληπτό ακουστικά από τον άνθρωπο. Στη συνέχεια σε κάθε frame εφαρμόζουμε μια συνάρτηση παραθύρου (window) σύμφωνα με τη σχέση:

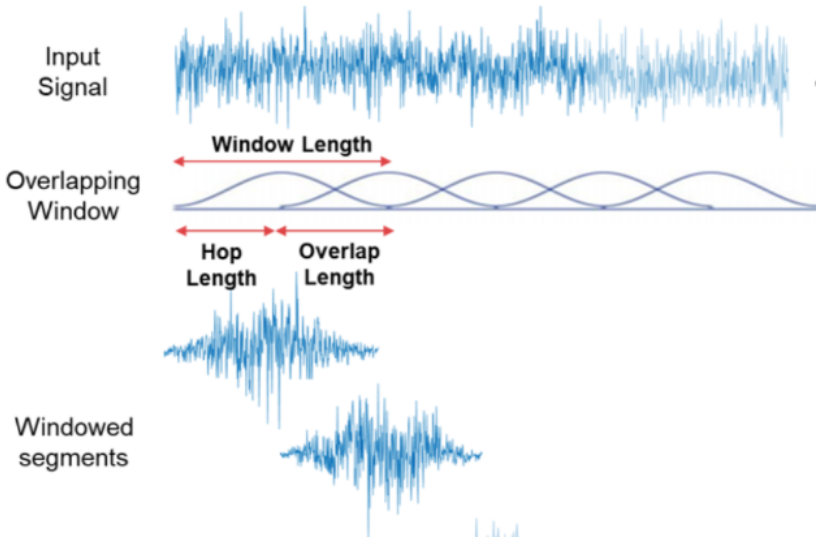
$$y(n) = w(n) \cdot s(n), \tag{1.3.1}$$

όπου $w(n)$ είναι η τιμή της συνάρτησης window στο σημείο n του εκάστοτε frame και $s(n)$ η τιμή του σήματος στο ίδιο σημείο. Συνήθως ως window χρησιμοποιείται η συνάρτηση Hamming που δίνεται από τον τύπο:

$$w(n) = \begin{cases} 0.54 - 0.46\cos\left(\frac{2\pi n}{L-1}\right), & 0 \leq n \leq L-1 \\ 0, & \text{αλλού,} \end{cases}$$

με $L$ να είναι το πλήθος των δειγμάτων σε ένα frame. Ο λόγος που εφαρμόζουμε μια συνάρτηση «παραθύρου» σε κάθε frame είναι για να αποφύγουμε το φαινόμενο της φασματικής διαρροής (spectral leakage). Τις περισσότερες φορές το ηχητικό σήμα δεν αποτελείται από ακέραιο αριθμό περιόδων, οπότε αν δε λάβουμε υπόψιν τις ασυνέχειες που εμφανίζονται στις άκρες του ηχητικού σήματος, καταλήγουμε σε ένα spectrum που περιέχει συχνότητες, οι οποίες φαίνεται να έχουν μεγάλη συνεισφορά στο σήμα μας, αλλά στην πραγματικότητα προκύπτουν λόγω των ασυνεχειών στις άκρες της κυματομορφής. Εφαρμόζοντας μια συνάρτηση window όπως η Hamming, οι τιμές του σήματος που βρίσκονται στα άκρα του παραθύρου τείνουν προς το μηδέν και έτσι αποφεύγουμε το φαινόμενο που αναφέραμε. Όμως εφαρμόζοντας μια τέτοια συνάρτηση παραθύρου, προκύπτει ότι οι τιμές στα άκρα του κάθε frame τείνουν στο μηδέν και κατά συνέπεια χάνουμε πληροφορία

από το αρχικό σήμα. Για να επιλυθεί το συγκεκριμένο πρόβλημα χρησιμοποιούνται επικαλυπτόμενα frames, όπως φαίνεται και στο Σχήμα 1.6. Παρατηρούμε ότι εισάγεται και η μεταβλητή hop length, η οποία αντιπροσωπεύει το πλήθος των δειγμάτων κατά τα οποία μετατοπιζόμαστε προς τα δεξιά κάθε φορά που μεταβαίνουμε στο επόμενο frame. Εδώ να σημειώσουμε ότι το πλήθος των δειγμάτων που επιλέγουμε σε κάθε frame ονομάζεται filter length και είναι πάντα μεγαλύτερο ή ίσο από το window length, δηλαδή το πλήθος των δειγμάτων σε κάθε frame στα οποία εφαρμόζουμε τη συνάρτηση παραθύρου. Στην επόμενη ενότητα περιγράφουμε τον τρόπο εξαγωγής χαρακτηριστικών μέσω του μετασχηματισμού Fourier.




Σχήμα 1.6: Εφαρμογή συνάρτησης παραθύρου στο ηχητικό σήμα χρησιμοποιώντας επικαλυπτόμενα frames.

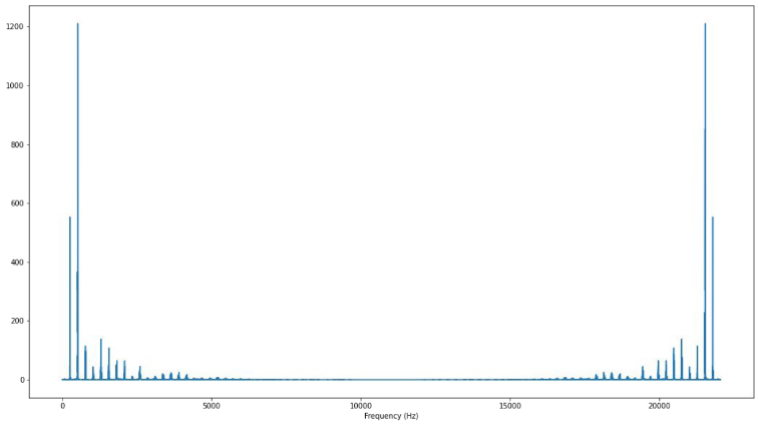
## 1.4 Μετασχηματισμός Fourier

Προηγουμένως είδαμε τον τρόπο με τον οποίο εφαρμόζουμε μια συνάρτηση παραθύρου σε κάθε frame από το ηχητικό σήμα. Στη συνέχεια θα πρέπει από κάθε frame να εξάγουμε κατάλληλα χαρακτηριστικά που σχετίζονται με τις επιμέρους συχνότητες που υπάρχουν στην κυματομορφή. Αυτό επιτυγχάνεται μέσω του μετασχηματισμού Fourier (FT) [BB86]. Όπως έχουμε αναφέρει, ένα κύμα αποτελείται από πολλά επιμέρους κύματα με διαφορετική συχνότητα το καθένα. Μέσω του μετασχηματισμού Fourier υπολογίζουμε το μέγεθος της συνεισφοράς κάθε επιμέρους συχνότητας στο τελικό κύμα. Συγκεκριμένα για ένα κύμα με συχνότητα $f$, ο μετασχηματισμός Fourier παράγει δύο αριθμούς, το magnitude (μέγεθος) και τη φάση, όπου το magnitude δηλώνει τη συνεισφορά της συχνότητας $f$ στο τελικό κύμα. Ο συγκεκριμένος μετασχηματισμός δρα πάνω σε σήματα που λαμβάνουν συνεχείς τιμές. Εφόσον έχουμε μετατρέψει το σήμα μας από αναλογικό σε ψηφιακό, στην πραγματικότητα εφαρμόζουμε το διακριτό μετασχηματισμό Fourier (DFT). Ο αλγόριθμος που χρησιμοποιείται για τον υπολογισμό του DFT είναι ο Fast Fourier Transform (FFT) που λειτουργεί αποδοτικά όταν το πλήθος των τιμών στις οποίες εφαρμόζεται είναι δύναμη του 2. Αυτός είναι και ο λόγος όπου το πλήθος των δειγμάτων που λαμβάνουμε σε κάθε frame του

ηχητικού σήματος είναι συνήθως 512, 1024 ή 2048. Ο DFT ορίζεται σύμφωνα με τη σχέση:

$$\hat{X}(k) = \sum_{n=0}^{N-1} x(n) e^{-i\frac{2\pi k}{N}n}, \tag{1.4.1}$$

όπου το N είναι το συνολικό πλήθος των δειγμάτων στο σήμα, $x(n)$ η τιμή του σήματος στο δείγμα $n$ και $\hat{X}(k)$ το αποτέλεσμα του μετασχηματισμού για τη διακριτή συχνότητα που ορίζεται από το δείκτη $k$. Επειδή και οι συχνότητες θα πρέπει να λαμβάνουν διακριτές τιμές, θεωρούμε ότι ο δείκτης $k$ παίρνει τιμές στο διάστημα φυσικών $[0, N-1]$, δηλαδή το πλήθος των συχνοτήτων για τις οποίες υπολογίζουμε το DFT ισούται με το συνολικό πλήθος των δειγμάτων $N$. Αν θεωρήσουμε ότι $T_s$ είναι η περίοδος δειγματοληψίας, δηλαδή ο χρόνος ανάμεσα σε δύο διαδοχικά δείγματα από το σήμα, τότε η $k-$οστή συχνότητα δίνεται από τον τύπο $f_k = \frac{k}{NT_s} = \frac{k}{N} s_r$, όπου $s_r$ είναι ο ρυθμός δειγματοληψίας. Άρα κάθε $k$ αντιστοιχεί σε μία από τις διακριτές συχνότητες. Το αποτέλεσμα του DFT είναι ένας μιγαδικός αριθμός του οποίου το μέτρο ισούται με το magnitude για μια συγκεκριμένη συχνότητα. Αν εφαρμόσουμε το DFT για κάθε μία διακριτή συχνότητα από το 0 έως το $s_r$ θα παρατηρήσουμε ότι τα magnitudes επαναλαμβάνονται μετά τη συχνότητα $\frac{s_r}{2}$, όπως φαίνεται και στο Σχήμα 1.7. Για το λόγο αυτό κρατάμε μόνο τις συχνότητες που βρίσκονται μέχρι τη συχνότητα με τιμή $\frac{s_r}{2}$.




Σχήμα 1.7: Το φάσμα συχνοτήτων με εφαρμογή του DFT. Παρατηρούμε ότι οι συχνότητες που βρίσκονται μετά τα $\frac{s_r}{2} = 11025$ Hz έχουν ίδιο magnitude με εκείνες που βρίσκονται πριν τα 11025 Hz.

Με το DFT λαμβάνουμε πληροφορία για εκείνες τις συχνότητες που παίζουν σημαντικό ρόλο σε ολόκληρο το σήμα αλλά δε γνωρίζουμε το πόσο συνεισφέρει κάθε μία στη διάρκεια του χρόνου, δηλαδή καθώς εξελίσσεται το σήμα. Προκειμένου λοιπόν να εξάγουμε πληροφορία που συνδυάζει την ένταση μιας συχνότητας με το χρόνο, εφαρμόζουμε το DFT σε κάθε frame του ηχητικού σήματος. Η παραλλαγή του αλγορίθμου ονομάζεται Short Time Fourier Transform (STFT) και δίνεται από τη σχέση:

$$\hat{X}(m,k) = \sum_{n=0}^{N'-1} x(n+mH) \cdot w(n) \cdot e^{-i\frac{2\pi k}{N'}n}, \tag{1.4.2}$$

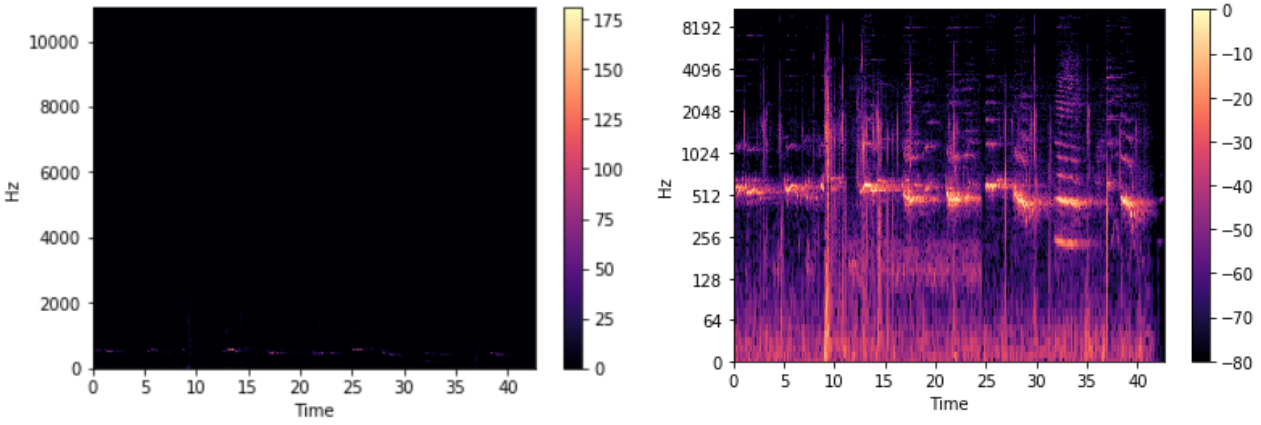
όπου εδώ το $N'$ εκφράζει το πλήθος των δειγμάτων που βρίσκονται σε ένα frame, $m$ είναι ο φυσικός αριθμός που δηλώνει σε ποιο frame εφαρμόζουμε το μετασχηματισμό και $H$ είναι το hop length,

δηλαδή το πλήθος των δειγμάτων που μετατοπιζόμαστε κατά τα δεξιά κάθε φορά που αλλάζουμε frame. Υπενθυμίζουμε ότι χρησιμοποιούμε επικαλυπτόμενα frames ώστε να αποφύγουμε την απώλεια πληροφορίας όταν εφαρμόσουμε τη συνάρτηση παραθύρου $w(n)$ στο σήμα. Εφαρμόζοντας λοιπόν STFT λαμβάνουμε για κάθε frame ορισμένους μιγαδικούς αριθμούς που αντιπροσωπεύουν το magnitude και τη φάση για το εκάστοτε διάστημα συχνοτήτων. Το πλήθος των διαστημάτων συχνοτήτων (frequency bins) ισούται με $\frac{N'}{2} + 1$, αφού ο αριθμός των διακριτών συχνοτήτων ισούται με το μισό του πλήθους των δειγμάτων $N'$ σε κάθε frame. Για παράδειγμα αν έχουμε ρυθμό δειγματοληψίας $s_r = 16000$ Hz και κάθε frame αποτελείται από $N' = 1024$ δείγματα, τότε θεωρούμε 513 διαστήματα ξεκινώντας από τη συχνότητα 0 μέχρι και τη συχνότητα 8000 Hz. Επίσης το πλήθος των frames θα ισούται με $\frac{N-N'}{H} + 1$, όπου $N$ το πλήθος των δειγμάτων σε όλο το σήμα και $H$ το hop length. Οπότε αν έχουμε συνολικά $N = 10^5$ δείγματα, $N' = 1024$ δείγματα σε κάθε frame και hop length $H = 200$ δείγματα, τότε το πλήθος των frames του ηχητικού σήματος θα ισούται με 495. Παρατηρούμε ότι υπάρχει ένα trade-off μεταξύ του χρόνου και της συχνότητας, το οποίο εξαρτάται από την επιλογή του πλήθους δειγμάτων $N'$ σε ένα frame. Προφανώς αυξάνοντας τα δείγματα $N'$ λαμβάνουμε και περισσότερες διακριτές συχνότητες εφόσον ο αριθμός τους εξαρτάται από το πλήθος των δειγμάτων σε ένα frame. Όμως επειδή παίρνουμε περισσότερα δείγματα αυτόματα μειώνεται και το συνολικό πλήθος των frames. Αντιθέτως μειώνοντας τον αριθμό των δειγμάτων σε ένα frame, μειώνεται συγχρόνως και το πλήθος των διακριτών συχνοτήτων αλλά αυξάνεται η χρονική ανάλυση, δηλαδή το πλήθος των frames στο σήμα. Συνήθως επιλέγουμε τιμές για τις οποίες ικανοποιείται μια σχετική ισορροπία στο συγκεκριμένο trade-off. Έτσι λοιπόν συμπεραίνουμε ότι με τη μέθοδο STFT μπορούμε να εξάγουμε χρονική πληροφορία για την ένταση των επιμέρους συχνοτήτων του ηχητικού σήματος, εφαρμόζοντας το DFT σε κάθε frame ξεχωριστά. Στη συνέχεια παρουσιάζουμε τον τρόπο με τον οποίο οπτικοποιούμε τη συγκεκριμένη πληροφορία χρησιμοποιώντας το φασματογράφημα ή αλλιώς spectrogram.

## 1.5 Φασματογράφημα

Το φασματογράφημα (spectrogram) είναι ένας οπτικός τρόπος αναπαράστασης του αποτελέσματος που προκύπτει από το STFT. Μέσω αυτού μπορούμε να ερμηνεύσουμε την ενέργεια κάθε συχνότητας στο σήμα για κάθε χρονικό διάστημα (frame). Όπως είδαμε σε κάθε frame το αποτέλεσμα του STFT είναι ορισμένοι μιγαδικοί αριθμοί όπου για κάθε διάστημα διακριτών συχνοτήτων, αντιπροσωπεύουν το magnitude και τη φάση. Για να οπτικοποιήσουμε την ένταση των συχνοτήτων χρησιμοποιούμε το μέτρο αυτών των μιγαδικών αριθμών, δηλαδή το spectrogram προκύπτει από τη σχέση $S = |\hat{X}(m, k)|$. Στο Σχήμα 1.8 (αριστερά) παρουσιάζεται η αναπαράσταση ενός φασματογραφήματος που προκύπτει από την προηγούμενη σχέση. Δυστυχώς δε μπορούμε να διακρίνουμε εύκολα τα σημαντικά χαρακτηριστικά του, παρά μόνο σε ελάχιστες χαμηλές συχνότητες. Ο λόγος γι' αυτό είναι ότι ο άνθρωπος αντιλαμβάνεται τη συχνότητα με λογαριθμικό τρόπο και όχι γραμμικό, άρα είναι αναμενόμενο πως πάνω από ένα κατώφλι οι περισσότερες συχνότητες δε θα έχουν συνεισφορά στο ηχητικό κύμα (βλ. μαύρη εικόνα). Λαμβάνοντας υπόψιν τη συγκεκριμένη παρατήρηση μπορούμε να πάρουμε λογάριθμο στον άξονα των συχνοτήτων και αντί για γραμμική κλίμακα να χρησιμοποιήσουμε την κλίμακα dB που είναι καταλληλότερη για την ένταση των συχνοτήτων. Το αποτέλεσμα φαίνεται στο Σχήμα 1.8 (δεξιά), όπου μπορούμε να διακρίνουμε περισσότερη πληροφορία σε αυτό. Γενικότερα σε ένα φασματογράφημα στον άξονα x βρίσκεται ο χρόνος και στον άξονα y η συχνότητα. Ανάλογα με το πλήθος των frames που έχουμε στον άξονα x παίρνουμε και μία τιμή για κάθε διάστημα διακριτών συχνοτήτων στον άξονα y.

Η τιμή αυτή δηλώνει την ένταση ενός συγκεκριμένου εύρους συχνοτήτων και αναπαρίσταται από τη φωτεινότητα της αντίστοιχης περιοχής στο φασματογράφημα. Επομένως στις περιοχές όπου έχουμε περισσότερη φωτεινότητα, η συνεισφορά εκείνων των συχνοτήτων είναι υψηλότερη.




Σχήμα 1.8: Οπτική αναπαράσταση της έντασης των επιμέρους συχνοτήτων σε ένα ηχητικό κύμα μέσω ενός φασματογραφήματος. Το δεξί φασματογράφημα περιέχει περισσότερη πληροφορία από το αριστερό, αφού έχουμε χρησιμοποιήσει την κλίμακα dB για την ένταση των συχνοτήτων.

Γενικότερα όσον αφορά το πρόβλημα της σύνθεσης φωνής από κείμενο, αλλά και άλλες εφαρμογές που σχετίζονται με εξαγωγή χαρακτηριστικών από το ηχητικό σήμα, είναι σύνηθες αντί για το απλό φασματογράφημα να χρησιμοποιούμε το φασματογράφημα στην κλίμακα mel (mel-spectrogram). Η κλίμακα mel είναι μια κλίμακα συχνοτήτων που είναι κατάλληλη για να μοντελοποιήσουμε τον τρόπο με τον οποίο αντιλαμβάνεται τη συχνότητα ο άνθρωπος. Συγκεκριμένα έχει παρατηρηθεί ότι μπορούμε να αντιληφθούμε ευκολότερα μεταβολές σε χαμηλές συχνότητες συγκριτικά με μεταβολές που συμβαίνουν σε υψηλές συχνότητες. Για παράδειγμα αν έχουμε δύο ηχητικά σήματα με συχνότητες 60 και 260 Hz και άλλα δύο με συχνότητες 1600 και 1800 Hz, τότε μπορούμε να αντιληφθούμε ευκολότερα τη διαφορά στα δύο πρώτα σε σχέση με τα άλλα δύο, παρ' όλο που και τα δύο ζευγάρια έχουν διαφορά συχνότητας 200 Hz. Η κλίμακα mel υπολογίζεται εφαρμόζοντας ένα λογαριθμικό μετασχηματισμό που δίνεται από τη σχέση:

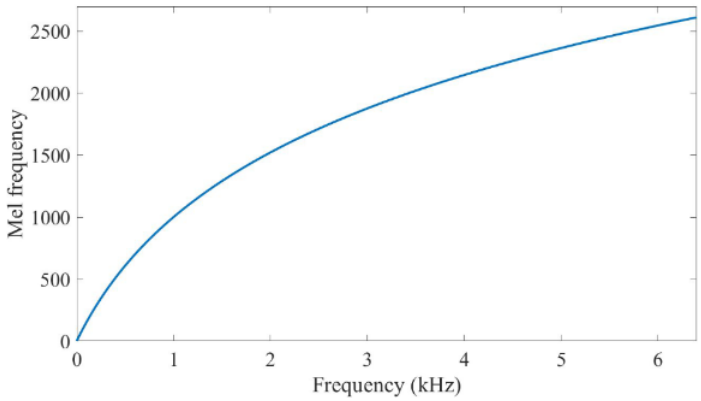
$$mel(f) = 2595 \log_{10}\left(1 + \frac{f}{700}\right), \tag{1.5.1}$$

επομένως η συχνότητα μετρημένη σε Hertz υπολογίζεται εύκολα αντιστρέφοντας την παραπάνω σχέση ως εξής:

$$f = 700(10^{m/2595} - 1). \tag{1.5.2}$$

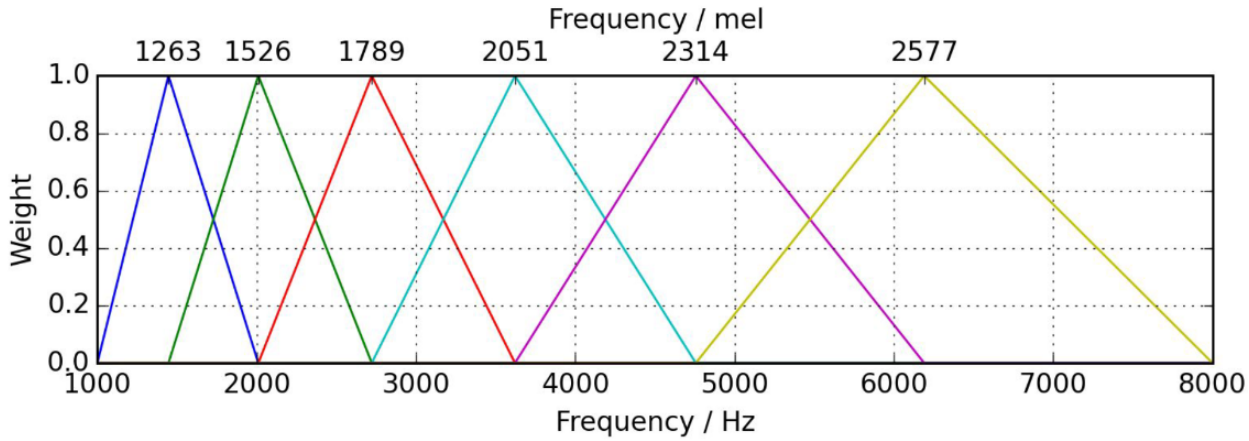
Στο Διάγραμμα 1.9 φαίνεται και η λογαριθμική σχέση που συνδέει την κλίμακα mel με την κλίμακα Hertz.

Για να κατασκευάσουμε τώρα το mel-spectrogram πρέπει να μετατρέψουμε τις συχνότητες στην κλίμακα mel. Αρχικά επιλέγουμε τον αριθμό των mels που θα χρησιμοποιήσουμε. Συνήθεις τιμές που χρησιμοποιούνται στη σύνθεση φωνής είναι 80, αλλά γενικά η τιμή αυτή εξαρτάται από το πρόβλημα που μελετάμε. Στη συνέχεια κατασκευάζουμε τα mel filter banks ακολουθώντας τα παρακάτω βήματα. Αφού επιλέξουμε το πλήθος των mels που θα χρησιμοποιήσουμε υπολογίζουμε τις αντίστοιχες συχνότητες στην κλίμακα mel για την ελάχιστη και τη μέγιστη συχνότητα




Διάγραμμα 1.9: Η λογαριθμική σχέση μεταξύ της κλίμακας mel και της κλίμακας Hertz.

σε Hz. Παραδείγματος χάριν με βάση το Σχήμα 1.10 όπου η ελάχιστη και μέγιστη συχνότητα είναι 1000 και 8000 Hz αντίστοιχα, η ελάχιστη και μέγιστη συχνότητα στην κλίμακα mel θα είναι 1000 και 2840 mels αντίστοιχα σύμφωνα με τη Σχέση 1.5.1. Έπειτα θεωρούμε τις ενδιάμεσες συχνότητες αυτών στην κλίμακα mel, οι οποίες είναι όσες και το πλήθος των mels που επιλέξαμε αρχικά. Εδώ έχουμε επιλέξει 6 mels, οπότε οι αντίστοιχες ισαπέχουσες συχνότητες είναι οι $1263, 1526, \ldots, 2577$. Ύστερα χρησιμοποιούμε τη Σχέση 1.5.2 και παίρνουμε τις αντίστοιχες 6 συχνότητες στην κλίμακα Hz, τις οποίες και στρογγυλοποιούμε στην πλησιέστερη διακριτή συχνότητα. Τέλος κατασκευάζουμε τα filter banks: για παράδειγμα για το πρώτο τριγωνικό φίλτρο σύμφωνα με το Σχήμα 1.10 ξεκινώντας από την πρώτη συχνότητα 1000 Hz ενώνουμε με τον άνω άξονα στην τιμή 1263 που είναι η πρώτη τιμή για τη συχνότητα στην κλίμακα mel και καταλήγουμε περίπου στα 2000 Hz, στο σημείο δηλαδή που βρίσκεται η τρίτη διακριτή συχνότητα στην κλίμακα Hz. Όμοια κατασκευάζονται και τα υπόλοιπα φίλτρα. Τελικά για κάθε ένα από τα




Σχήμα 1.10: Mel filter banks.

6 mels καταλήγουμε σε ορισμένα βάρη που παίρνουν τιμές στο διάστημα $[0, 1]$, όπως φαίνονται και στον άξονα $y$. Το πλήθος αυτών των βαρών ισούται με το πλήθος των διακριτών συχνοτήτων στην κλίμακα Hz. Άρα για κάθε mel παίρνουμε ένα διάνυσμα που περιέχει τα βάρη που προκύπτουν από τα τριγωνικά φίλτρα για κάθε διακριτή συχνότητα. Επομένως καταλήγουμε σε έναν πίνακα με γραμμές ίσες με τον αριθμό των mels και στήλες ίσες με το πλήθος των διακριτών

συχνοτήτων. Πολλαπλασιάζοντας το συγκεκριμένο πίνακα με τον αντίστοιχο πίνακα του γραμμικού φασματογραφήματος, προκύπτει το mel-spectrogram διαστάσεων (mels, frames).

Όπως αναφέραμε, το φασματογράφημα περιέχει σημαντική πληροφορία από το ηχητικό σήμα που αφορά την ένταση των επιμέρους συχνοτήτων σε αυτό. Τα περισσότερα state of the art μοντέλα σύνθεσης φωνής από κείμενο που εξετάζουμε στη συνέχεια χρησιμοποιούν ως ενδιάμεσο χαρακτηριστικό το φασματογράφημα στην κλίμακα mel για να εξάγουν την παραγόμενη κυματομορφή ήχου. Στην επόμενη ενότητα παρουσιάζουμε τις πιο βασικές προσεγγίσεις που έχουν αναπτυχθεί μέχρι σήμερα στο πρόβλημα της σύνθεσης φωνής από κείμενο.

## 1.6 Προσεγγίσεις στη σύνθεση φωνής

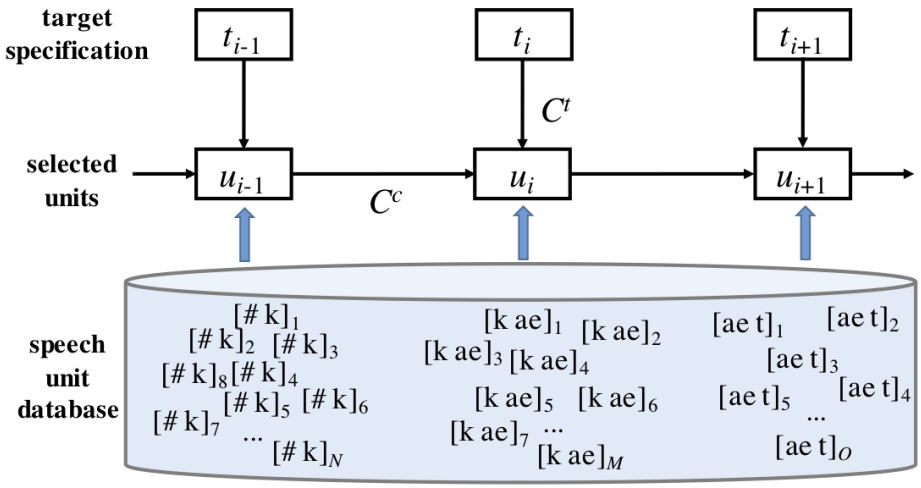
### 1.6.1 Concatenative TTS

Μία από τις παραδοσιακές προσεγγίσεις στο πρόβλημα της σύνθεσης φωνής από κείμενο είναι η συναθροιστική ή αλλιώς concatenative μέθοδος. Σύμφωνα με αυτή προκειμένου να παράγουμε ομιλία από ένα κείμενο μπορούμε να χρησιμοποιήσουμε ηχητικά δείγματα τα οποία είναι αποθηκευμένα σε μια μέγαλη βάση δεδομένων. Η βάση αυτή περιλαμβάνει ηχογραφήσεις όπως για παράδειγμα φωνήματα, συλλαβές, λέξεις κτλ., τις οποίες αν συνενώσουμε μπορούμε να παράγουμε ομιλία με μεγάλη φυσικότητα. Για να μετατρέψουμε λοιπόν ένα συγκεκριμένο κείμενο σε φωνή πρώτα γίνεται αναζήτηση των αντίστοιχων μονάδων ήχου που περιέχονται στη βάση (unit selection) και ύστερα οι μονάδες αυτές ενώνονται ώστε να προκύψει το επιθυμητό αποτέλεσμα. Προφανώς η μέθοδος αυτή απαιτεί μια πολύ μεγάλη βάση δεδομένων ώστε να περιέχει όλες τις δυνατές εκφωνήσεις για κάθε πιθανό συνδυασμό λέξεων ή προτάσεων. Επιπλέον για να υπάρχει φυσικότητα θα πρέπει οι ηχογραφημένες εκφωνήσεις που συνενώνονται να προέρχονται από έναν ομιλητή. Αυτό είναι αρκετά περιοριστικό αν σκεφτούμε ότι για να συλλέξουμε ηχογραφήσεις από όλες τις πιθανές λέξεις σε μια γλώσσα, μαζί με όλους τους δυνατούς τρόπους εκφώνησης αυτών από έναν μόνο ομιλητή, είναι πολύ χρονοβόρο. Επίσης υπάρχει περιορισμός ως προς το στυλ και την προσωδία της παραγόμενης φωνής αφού ενώνοντας απλά ηχητικά δείγματα χάνονται σημαντικά χαρακτηριστικά όπως ο τόνος, ο ρυθμός, το συναίσθημα κτλ. Στην πράξη αντί για λέξεις χρησιμοποιούνται μικρότερες μονάδες όπως είναι τα φωνήματα. Στη συνέχεια περιγράφουμε τον τρόπο λειτουργίας της συναθροιστικής μεθόδου.

#### Μεθοδολογία και εκπαίδευση

Η μεθοδολογία για την επιλογή των κατάλληλων μονάδων ήχου από τη βάση των ηχογραφημένων εκφωνήσεων έγκειται στην ελαχιστοποίηση δύο συναρτήσεων κόστους σύμφωνα με τους Hunt και Black [HB96]. Η πρώτη συνάρτηση $C^t(u_i, t_i)$ ονομάζεται target cost και χρησιμοποιείται προκειμένου η μονάδα $u_i$ που επιλέγεται από τη βάση να συμπίπτει όσο γίνεται περισσότερο με την target μονάδα $t_i$ που ανήκει στην πραγματική εκφώνηση. Η δεύτερη συνάρτηση ελαχιστοποίησης $C^c(u_{i-1}, u_i)$ ονομάζεται concatenation cost και χρησιμοποιείται προκειμένου να υπάρχει ομαλή μετάβαση από τη μονάδα $u_{i-1}$ που επιλέχθηκε στο προηγούμενο βήμα και της υποψήφιας μονάδας $u_i$. Ελαχιστοποιώντας τις δύο συναρτήσεις κόστους επιλέγουμε τελικά τις μονάδες εκείνες, όπου ενώνοντάς τες ο παραγόμενος ήχος είναι όσο το δυνατόν πιο κοντά στον πραγματικό ήχο και επιπλέον υπάρχει ροή μεταξύ των διαδοχικών μονάδων. Στο Σχήμα 1.11 παρατηρούμε τη διαδικασία επιλογής των κατάλληλων μονάδων για το σχηματισμό της λέξης “cat”, από μια βάση που περιέχει ηχογραφήσεις από φωνήματα. Σε κάθε βήμα ελαχιστοποιείται το άθροισμα $C^t(u_i, t_i)+C^c(u_{i-1}, u_i)$.

Μια μονάδα $u_i$ μπορεί να προφέρεται με παραπάνω από έναν τρόπους και σε κάθε περίπτωση εξαρτά-




Σχήμα 1.11: Η διαδικασία επιλογής των βέλτιστων φωνημάτων $u_i$ από μια βάση ηχογραφήσεων. Σε κάθε βήμα ελαχιστοποιείται το target cost $C^t(u_i, t_i)$ και το concatenation cost $C^c(u_{i-1}, u_i)$. [Bäc+22]

ται από χαρακτηριστικά όπως η διάρκεια, η ένταση, ο τόνος κ.ά. Επομένως κάθε κόστος μπορεί να διασπαστεί σε επιμέρους κόστη, όπου καθένα λαμβάνει υπόψιν κάποιο από αυτά τα χαρακτηριστικά. Έτσι το target και concatentation cost μπορούν αντίστοιχα να γραφούν ως

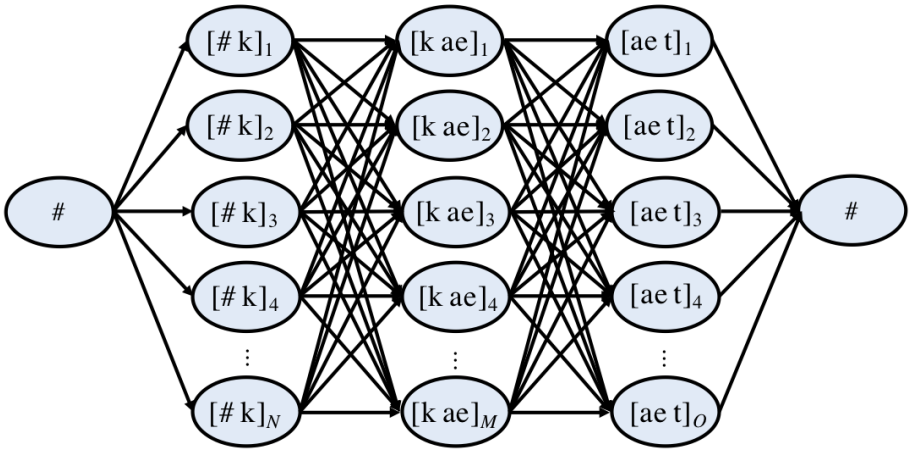
$$C^t(u_i, t_i) = \sum_{j=1}^{P} w_j^t C_j^t(u_i, t_i) \quad \text{και} \quad C^c(u_{i-1}, u_i) = \sum_{j=1}^{Q} w_j^c C_j^c(u_{i-1}, u_i), \tag{1.6.1}$$

με $w_j^t, w_j^c$ να είναι τα βάρη για κάθε επιμέρους κόστος. Τελικά το συνολικό κόστος ελαχιστοποίησης θα είναι

$$C(u, t) = \sum_{i=1}^{n} C^t(u_i, t_i) + \sum_{i=2}^{n} C^c(u_{i-1}, u_i) + C^c(\#, u_1) + C^c(u_n, \#), \tag{1.6.2}$$

όπου το σύμβολο # δηλώνει το silence στην αρχή και στο τέλος της εκφώνησης και $n$ είναι το συνολικό πλήθος των μονάδων που επιλέγουμε για το σχηματισμό της. Όλοι οι πιθανοί συνδυασμοί μπορούν να αναπαρασταθούν μέσω ενός συνόλου «μονοπατιών», όπως φαίνεται στο Σχήμα 1.12. Ξεκινώντας από το silence σε κάθε βήμα επιλέγεται η μονάδα που ελαχιστοποιεί τα κόστη που αναφέραμε. Επειδή το πλήθος όλων των δυνατών συνδυασμών είναι αρκετά μεγάλο χρησιμοποιείται ο αλγόριθμος Viterbi [For73] προκειμένου να υπολογιστεί το μονοπάτι με το ελάχιστο κόστος.

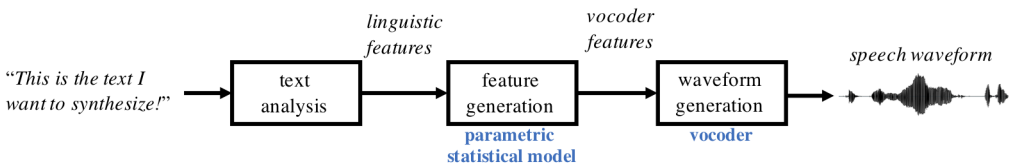
Τέλος για την επιλογή της μορφής των συναρτήσεων κόστους στη Σχέση 1.6.1 αλλά και τα βάρη που σταθμίζουν κάθε επιμέρους κόστος απαιτείται γνώση των χαρακτηριστικών που αντιπροσωπεύει κάθε ένα από αυτά. Ενδεικτικά αναφέρουμε ότι για την επιλογή των βαρών μια απλή προσέγγιση είναι η μέθοδος grid search σύμφωνα με την οποία ελέγχεται ένας μεγάλος συνδυασμός από βάρη και τελικά επιλέγονται εκείνα όπου δίνουν το μικρότερο κόστος συνολικά.




Σχήμα 1.12: Τα μονοπάτια που εκφράζουν όλους τους πιθανούς συνδυασμούς από ηχογραφημένες μονάδες για το σχηματισμό της λέξης "cat". Οι ακμές μεταξύ των διαδοχικών nodes συμβολίζουν το κόστος σε κάθε βήμα. Τελικά επιλέγονται οι μονάδες εκείνες από το μονοπάτι με το μικρότερο συνολικό κόστος. [Bäc+22]

### 1.6.2 Statistical Parametric TTS

Αν και η συναθροιστική μέθοδος μπορεί να παράγει ηχητικά δείγματα τα οποία χαρακτηρίζονται από φυσικότητα, εντούτοις είδαμε ότι έχει ορισμένα μειονεκτήματα. Πρώτον απαιτείται μια μεγάλη βάση ηχογραφήσεων για να καλύψει όλες τις δυνατές περιπτώσεις και συνδυασμούς λέξεων. Επιπλέον υπάρχει περιορισμός στο στυλ και στην προσωδία της παραγόμενης εκφώνησης αφού είναι δύσκολο να μοντελοποιηθούν σημαντικά χαρακτηριστικά αυτής, όπως ο τόνος, η ταχύτητα και το συναίσθημα. Για τους λόγους αυτούς ως εναλλακτική προσέγγιση στο πρόβλημα της σύνθεσης φωνής από κείμενο μπορεί να θεωρηθεί η στατιστική παραμετρική μέθοδος. Σύμφωνα με αυτήν προκειμένου να παράγουμε ομιλία από ένα κείμενο χρησιμοποιούμε τεχνικές και μοντέλα μηχανικής μάθησης τα οποία εκπαιδεύονται σε μεγάλα σύνολα δεδομένων. Τα σύνολα αυτά περιέχουν ηχογραφήσεις, συνήθως με προτάσεις από έναν ή περισσότερους ομιλητές μαζί με τα αντίστοιχα κείμενα (transcriptions), όπως και στο πρόβλημα της αναγνώρισης φωνής (speech recognition). Όπως φαίνεται και στο Σχήμα 1.13, ένα σύστημα που βασίζεται στην παραμετρική μέθοδο αποτελείται από τρία βασικά δίκτυα: το text analysis, ένα ακουστικό μοντέλο (feature generation) και έναν vocoder.



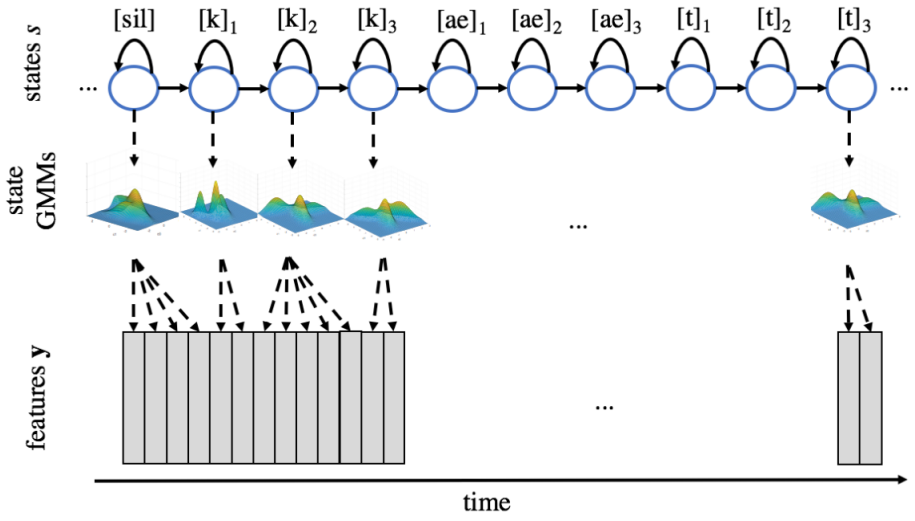

Σχήμα 1.13: Σύστημα σύνθεσης φωνής από κείμενο με τη στατιστική παραμετρική προσέγγιση. [Bäc+22]

### Text analysis

Αρχικά το text analysis module δέχεται το κείμενο εισόδου και παράγει ορισμένα γλωσσικά χαρακτηριστικά όπως είναι τα φωνήματα ή η διάρκεια των χαρακτήρων. Ορισμένες από τις βασικές μεθόδους που χρησιμοποιούνται είναι το text normalization και η μετατροπή χαρακτήρων σε φωνήματα grapheme-to-phoneme (G2P). Το text normalization αφορά βασικά βήματα προεπεξεργασίας ενός κειμένου, όπως το tokenization (αντιστοίχιση των χαρακτήρων σε tokens), μετατροπή των χαρακτήρων σε μικρά γράμματα, μετατροπή των αριθμών σε χαρακτήρες κ.ο.κ. Για παράδειγμα αν έχουμε τη φράση «Αυτό κοστίζει 10$», τότε θα πρέπει αρχικά να γίνει η μετατροπή αυτής ως: «αυτό κοστίζει δέκα δολάρια» και εν συνεχεία η πρόταση να χωριστεί σε tokens που αντιστοιχούν σε κάθε χαρακτήρα. Έπειτα ένας αλγόριθμος G2P παράγει μια ακολουθία φωνημάτων από τους χαρακτήρες του κειμένου. Σημειώνεται ότι η αντιστοίχιση των χαρακτήρων σε φωνήματα δεν είναι ένα προς ένα αφού για παράδειγμα το γράμμα «ξ» αντιστοιχεί στο φώνημα «κσ» που αποτελείται από δύο χαρακτήρες. Αφού προκύψει η ακολουθία των φωνημάτων, σειρά έχει το ακουστικό μοντέλο.

### Acoustic model

Το ακουστικό ή αλλιώς παραμετρικό μοντέλο είναι εκείνο που δέχεται τα γλωσσικά χαρακτηριστικά από το text analysis και παράγει ενδιάμεσα ακουστικά χαρακτηριστικά, όπως είναι τα mel-frequency cepstral coefficients (MFCC) και οι παράγωγοί τους, η θεμελιώδης συχνότητα $F_0$ κ.ά. Τα χαρακτηριστικά αυτά εξάγονται ανά frame και αποθηκεύονται σε διανύσματα. Συνήθως ως ακουστικό μοντέλο χρησιμοποιείται ο συνδυασμός Hidden Markov Model-Gaussian Mixture Model (HMM-GMM). Στο Σχήμα 1.14 βλέπουμε τη λειτουργία ενός τέτοιου μοντέλου για την εξαγωγή ακουστικών χαρακτηριστικών που αντιστοιχούν στη λέξη “cat”.




Σχήμα 1.14: Παραμετρικό μοντέλο της μορφής HMM-GMM για την παραγωγή ακουστικών χαρακτηριστικών που αντιστοιχούν στη λέξη “cat”. [Bäc+22]

Παρατηρούμε ότι κάθε φώνημα της λέξης αντιστοιχεί σε τρία states. Ξεκινώντας από το state που αντιστοιχεί σε silence [sil], η πιθανότητα $P(s_{i+1}|s_i)$ από το state $s_i$ στο state $s_{i+1}$ εκφράζει το πόσο πιθανό είναι να μεταβούμε από ένα φώνημα στο επόμενο ή να παραμείνουμε στο ίδιο τη χρονική στιγμή $i$. Τα ακουστικά χαρακτηριστικά μοντελοποιούνται από το μοντέλο GMM

σύμφωνα με τη σχέση

$$P(\mathbf{y}|s) = \sum_{i=1}^{K} w_{s,i}\, \mathcal{N}(\mathbf{y}|\boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i}), \tag{1.6.3}$$

όπου τα βάρη $w_{s,i}$ αθροίζουν στη μονάδα και σταθμίζουν κάθε κανονική κατανομή με μέση τιμή $\boldsymbol{\mu}_{s,i}$ και πίνακα διασποράς $\boldsymbol{\Sigma}_{s,i}$. Η Σχέση 1.6.3 εκφράζει το πόσο πιθανό είναι να προκύψουν τα χαρακτηριστικά $\mathbf{y}$ δεδομένου ότι βρισκόμαστε στο state $s$.

### Vocoder

Ως τελευταίο μέρος ενός παραμετρικού συστήματος χρησιμοποιείται ένα μοντέλο vocoder. Ο vocoder δέχεται ως είσοδο τα χαρακτηριστικά που προκύπτουν από το ακουστικό μοντέλο και εξάγει την κυματομορφή ήχου. Συνηθισμένοι vocoders που χρησιμοποιούνται στα παραμετρικά συστήματα είναι ο STRAIGHT [Kaw06] και ο WORLD [MYO16]. Επίσης ο Griffin-Lim [GL84] είναι ένας απλός vocoder που δέχεται το φασματογράφημα σε γραμμική κλίμακα και παράγει την αντίστοιχη κυματομορφή ήχου. Πιο σύνθετοι vocoders στηρίζονται σε μοντέλα βαθιάς μάθησης και δίνουν καλύτερα αποτελέσματα όσον αφορά το τελικό ηχητικό δείγμα.

Η εκπαίδευση ενός παραμετρικού συστήματος στοχεύει στην εκμάθηση των παραμέτρων του ακουστικού μοντέλου HMM-GMM. Σε πρώτη φάση εξάγονται τα ακουστικά χαρακτηριστικά από τα ηχητικά δείγματα του συνόλου εκπαίδευσης καθώς επίσης και τα γλωσσικά χαρακτηριστικά (π.χ. φωνήματα) από το text analysis module. Το ακουστικό μοντέλο εκπαιδεύεται με τον αλγόριθμο EM [DLR77] ώστε τα χαρακτηριστικά που προκύπτουν σε κάθε state να συμπίπτουν με τα ακουστικά χαρακτηριστικά (observations) που έχουμε εξάγει από τα δεδομένα εκπαίδευσης. Κατανοούμε λοιπόν ότι υπάρχουν ορισμένα πλεονεκτήματα της στατιστικής παραμετρικής προσέγγισης σε σχέση με τη συναθροιστική. Ένα στατιστικό μοντέλο ουσιαστικά μαθαίνει να παράγει ηχητικά δείγμα μέσω των παραμέτρων που το ορίζουν και έτσι υπάρχει μεγαλύτερη ευελιξία στο τελικό ηχητικό δείγμα που παράγεται. Επίσης σημαντικό πλεονέκτημα είναι ότι δε χρειάζεται μια μεγάλη βάση δεδομένων για να αποθηκεύονται ηχογραφήσεις, αφού χρησιμοποιώντας μικρότερο πλήθος δεδομένων μπορούμε να παράγουμε ομιλία με μεγάλη φυσικότητα.

## 1.6.3 Neural TTS

Η πιο σύγχρονη προσέγγιση που μελετάμε στο πρόβλημα της σύνθεσης φωνής από κείμενο είναι το neural text-to-speech. Ως βασικό στοιχείο αυτής της μεθόδου είναι η χρήση μοντέλων βαθιάς μάθησης για την παραγωγή ομιλίας όσο το δυνατόν πιο κοντά στην ανθρώπινη. Ακόμα και στα παραμετρικά συστήματα υπάρχουν προσεγγίσεις σύμφωνα με τις οποίες το μοντέλο HMM-GMM αντικαθίσταται από αναδρομικά νευρωνικά δίκτυα LSTM [HS97], τα οποία μοντελοποιούν κατάλληλα χρονικές εξαρτήσεις. Επίσης για την παραγωγή της κυματομορφής χρησιμοποιούνται πιο σύνθετοι vocoders, των οποίων η αρχιτεκτονική στηρίζεται στα νευρωνικά δίκτυα. Για παράδειγμα το WaveNet [Oor+16] που βασίζεται σε συνελικτικά δίκτυα, είναι ο πρώτος σύγχρονος vocoder που μπορεί να μετατρέψει τα γλωσσικά χαρακτηριστικά απευθείας στο αντίστοιχο ηχητικό δείγμα χωρίς τη χρήση ενδιάμεσων ακουστικών χαρακτηριστικών. Ακόμα η επεξεργασία του κειμένου και η παραγωγή γλωσσικών χαρακτηριστικών μπορεί να γίνει απευθείας με τα λεγόμενα end-to-end μοντέλα όπως το Tacotron2. Τα end-to-end μοντέλα αποτελούνται από δύο επιμέρους τμήματα. Το πρώτο δέχεται ως είσοδο το κείμενο (ακολουθία χαρακτήρων ή φωνημάτων) και εξάγει (στην πλειοψηφία των περιπτώσεων) το φασματογράφημα στην κλίμακα mel. Εν συνεχεία ένας vocoder επεξεργάζεται το φασματογράφημα και παράγει την κυματομορφή ήχου. Τα state

of the art μοντέλα που βασίζονται στο neural TTS ξεπερνούν τις προηγούμενες δύο προσεγγίσεις (παραμετρική και συναθροιστική) τόσο στη φυσικότητα των συνθετικών εκφωνήσεων όσο και στη δυνατότητα προσαρμογής των μοντέλων ώστε να λαμβάνουν υπόψιν και χαρακτηριστικά όπως το συναίσθημα, ο τόνος, η ταχύτητα κτλ. Επίσης δε χρειάζεται να υπάρχει αρκετή γνώση ως προς το ποια είναι τα κατάλληλα χαρακτηριστικά που θα επιλεγούν εφόσον η διαδικασία αυτή πραγματοποιείται από το μοντέλο. Για την εκπαίδευσή τους απαιτείται ένα μεγάλο πλήθος ηχογραφήσεων μαζί με τα αντίστοιχα κείμενα. Ενδεικτικά αναφέρουμε ότι το μοντέλο Tacotron2 σε συνδυασμό με το WaveNet εκπαιδεύτηκε σε ένα σύνολο δεδομένων με 24 ώρες ηχογράφησης. Επιπλέον απαιτείται μεγάλη υπολογιστική δύναμη (GPUs) αλλά και μεγάλος χρόνος εκπαίδευσης προκειμένου η συνθετική φωνή να έχει την επιθυμητή ποιότητα. Ειδικότερα όσον αφορά τους vocoders γίνεται προσπάθεια (π.χ. [Son+20]), προκειμένου να μειωθεί το υπολογιστικό κόστος και ο χρόνος εκπαίδευσής τους, αλλά συγχρόνως να παραμείνει υψηλή η ποιότητα της παραγόμενης φωνής.

## 1.7 Συμπέρασμα

Στο κεφάλαιο αυτό έγινε μια πρώτη εισαγωγή στο πρόβλημα της σύνθεσης φωνής από κείμενο. Αρχικά είδαμε ορισμένα βασικά βήματα που αφορούν την επεξεργασία των δεδομένων ήχου για την εξαγωγή των κατάλληλων χαρακτηριστικών όπως είναι το φασματογράφημα στην κλίμακα mel, και στη συνέχεια παρουσιάσαμε τις βασικότερες προσεγγίσεις που χρησιμοποιούνται για τη σύνθεση φωνής από κείμενο. Εξετάσαμε το βασικό τρόπο λειτουργίας τους και είδαμε ότι η πιο σύγχρονη προσέγγιση, το neural TTS, οδηγεί σε βελτιωμένα αποτελέσματα και έχει ορισμένα σημαντικά πλεονεκτήματα σε σχέση με τις άλλες δύο προσεγγίσεις. Στο επόμενο κεφάλαιο εστιάζουμε στις state of the art αρχιτεκτονικές που ανήκουν στην κατηγορία του neural TTS.

# Κεφάλαιο 2

# State of the art σύνθεση φωνής από κείμενο

## 2.1 Εισαγωγή

Στο κεφάλαιο αυτό μελετάμε τις βασικότερες αρχιτεκτονικές που ανήκουν στην κατηγορία του neural text-to-speech. Οι αρχιτεκτονικές αυτές βασίζονται κυρίως σε μοντέλα βαθιάς μάθησης προκειμένου να παράγουν ομιλία από ένα συγκεκριμένο κείμενο. Τα μοντέλα αυτά ονομάζονται end-to-end και ως επί το πλείστον αποτελούνται από δύο τμήματα. Το πρώτο τμήμα είναι υπεύθυνο για την εξαγωγή του φασματογραφήματος στην κλίμακα mel από τους χαρακτήρες ή τα φωνήματα του κειμένου που θέλουμε να μετατρέψουμε σε φωνή. Στην κατηγορία αυτή ανήκουν μοντέλα όπως το Tacotron [Wan+17], το Tacotron2 [She+18], ο Transformer TTS [Li+19] κ.ά. Το δεύτερο τμήμα ενός end-to-end συστήματος είναι ένα μοντέλο που ονομάζεται vocoder και επεξεργάζεται το φασματογράφημα στην κλίμακα mel προκειμένου να παράγει τη συνθετική κυματομορφή ήχου. Το WaveNet [Oor+16] και το WaveGlow [PVC19] είναι χαρακτηριστικά παραδείγματα από state of the art vocoders, τους οποίους αναλύουμε στη συνέχεια του κεφαλαίου. Επίσης υπάρχουν και περιπτώσεις μοντέλων που στηρίζονται στα λεγόμενα Generative Adversarial Networks (GANs) [Goo+14], όπως είναι το MelGAN [Kum+19] το οποίο έχει πολύ λιγότερες παραμέτρους από έναν vocoder και μπορεί να παράγει φωνή με καλή ποιότητα. Τέλος, κλείνουμε το κεφάλαιο παρουσιάζοντας το μοντέλο WaveGrad2 [Che+21], το οποίο στηρίζεται στα diffusion models [HJA20] και μπορεί να παράγει απευθείας φωνή από ένα κείμενο χωρίς τη χρήση ενδιάμεσων χαρακτηριστικών. Τα αποτελέσματα των μοντέλων είναι αρκετά ικανοποιητικά με την έννοια ότι η συνθετική φωνή που παράγουν μοιάζει πολύ με ανθρώπινη.

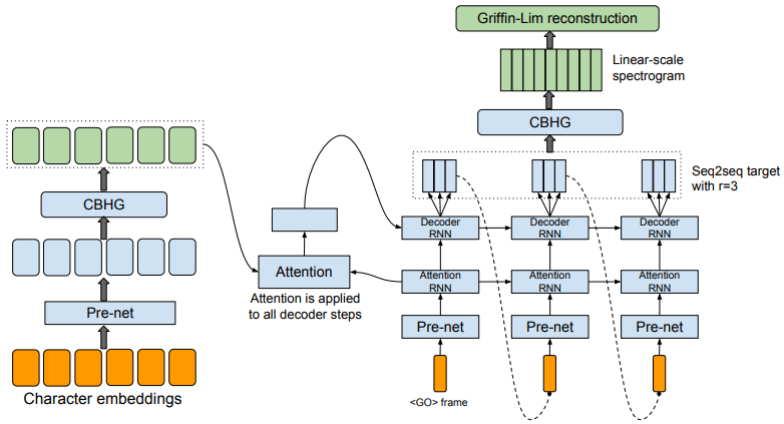
## 2.2 Tacotron

Το Tacotron [Wan+17] αποτελεί ένα end-to-end μοντέλο για τη σύνθεση φωνής από κείμενο. Το μοντέλο αυτό μπορεί να αντικαταστήσει τεχνικές που βασίζονται είτε σε συναθροιστικές (concatenative) [HB96] είτε σε παραμετρικές (parametric) [ZTB09] μεθόδους για την απευθείας παραγωγή φωνής από ένα κείμενο. Τα πλεονεκτήματά του σε σχέση με αυτές τις τεχνικές αφορούν κυρίως τη διαδικασία επιλογής κατάλληλων χαρακτηριστικών για την εκπαίδευση, αφού συνήθως στις συναθροιστικές ή τις παραμετρικές μεθόδους χρειάζεται ανθρώπινη γνώση για την εξαγωγή χαρακτηριστικών. Επίσης σε αυτές τις περιπτώσεις χρησιμοποιούνται μοντέλα τα οποία εκπαιδεύονται ανεξάρτητα. Για παράδειγμα μια στατιστική παραμετρική μέθοδος μπορεί να αποτελείται από ένα

text frontend για την εξαγωγή γλωσσικών χαρακτηριστικών, ένα μοντέλο διάρκειας (duration model), ένα μοντέλο πρόβλεψης ακουστικών χαρακτηριστικών καθώς και έναν vocoder για την μετατροπή των ακουστικών χαρακτηριστικών στη τελική κυματομορφή ήχου. Είναι σαφές λοιπόν ότι υπάρχει πολυπλοκότητα σε τέτοιου είδους συστήματα και έτσι αξιοποιώντας ένα end-to-end μοντέλο όπως το Tacotron, μπορεί να απλοποιηθεί αρκετά η διαδικασία εκπαίδευσης και συμπερασματολογίας. Το Tacotron εκπαιδεύεται πάνω σε ζεύγη κειμένου και ήχου. Συγκεκριμένα στην είσοδό του δέχεται μια ακολουθία χαρακτήρων που αποτελούν το κείμενο και παράγει το αντίστοιχο φασματογράφημα. Στη συνέχεια γίνεται ανακατασκευή του φασματογραφήματος μέσω του αλγορίθμου Griffin-Lim [GL84], ώστε να παραχθούν τα τελικά δείγματα ήχου. Εναλλακτικά μπορεί να χρησιμοποιηθεί ένα μοντέλο vocoder, όπως το WaveNet [Oor+16] για την παραγωγή της κυματορφής από το φασματογράφημα. Στη συνέχεια παρουσιάζουμε τα βασικά στοιχεία που δομούν την αρχιτεκτονική του μοντέλου.

### 2.2.1 Αρχιτεκτονική του μοντέλου

Η αρχιτεκτονική του Tacotron, η οποία παρουσιάζεται στο Σχήμα 2.1 βασίζεται σε ένα seq2seq μοντέλο με μηχανισμό προσοχής [BCB14]. Συγκεκριμένα αποτελείται από έναν encoder ο οποίος εξάγει τα κύρια χαρακτηριστικά της ακολουθίας εισόδου και στη συνεχεία ο decoder σε κάθε βήμα χρησιμοποιεί τα χαρακτηριστικά αυτά για να παράγει ορισμένα frames από το φασματογράφημα. Η αρχική είσοδος στο μοντέλο είναι ένα κείμενο όπου κάθε χαρακτήρας του αναπαρίσταται από ένα one-hot διανύσμα. Στη συνέχεια χρησιμοποιούνται embeddings μήκους 256, ώστε το embedded διάνυσμα κάθε χαρακτήρα να λαμβάνει συνεχείς τιμές.

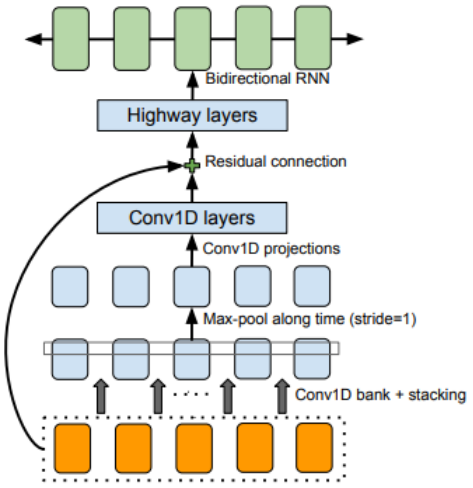



Σχήμα 2.1: Αρχιτεκτονική του μοντέλου Tacotron. (Αριστερά) Ο encoder εξάγει την αναπαράσταση της ακολουθίας εισόδου, δηλ. των embeddings κάθε χαρακτήρα στο κείμενο. (Δεξιά) Σε κάθε χρονικό βήμα ο decoder παράγει ορισμένα frames του φασματογραφήματος. Τέλος ο αλγόριθμος Griffin-Lim επεξεργάζεται το φασματογράφημα και παράγει τα δείγματα της κυματομορφής. [Wan+17]

#### Encoder

Ο encoder δέχεται τα embeddings κάθε χαρακτήρα και παράγει την «κρυφή» αναπαράσταση αυτών (hidden representation) μέσω δύο modules, του Prenet και του CBHG [LCH17]. Το Prenet αποτελείται από δύο fully connected επίπεδα με 256 και 128 νευρώνες αντίστοιχα, συνάρτηση

ενεργοποίησης της ReLU, καθώς και Dropout [Sri+14] με πιθανότητα 0.5 για την ομαλοποίηση και τη βελτίωση γενίκευσης του μοντέλου. Έπειτα ακολουθεί ένα δίκτυο που ονομάζεται CBHG. Το δίκτυο αυτό αποτελείται από τέσσερα επιμέρους modules, όπως φαίνεται και στο Σχήμα 2.2.




Σχήμα 2.2: Η δομή του δικτύου CBHG που χρησιμοποιείται στον encoder του Tacotron. Αποτελείται από τέσσερα επιμέρους modules: 1d-convolution bank, 1d convolution projections, δίκτυο Highway και ένα αμφίδρομο αναδρομικό δίκτυο GRU. [Wan+17]

Συγκεκριμένα η έξοδος από το Prenet τροφοδοτείται ανεξάρτητα σε $K = 16$ συνελικτικά επίπεδα μιας διάστασης (1d-convolution bank), όπου το καθένα έχει συνάρτηση ενεργοποίησης τη ReLU, μέγεθος πυρήνα (kernel size) $k = 1, \ldots, 16$ και 128 κανάλια εξόδου. Οι έξοδοι από τα όλα τα συνελικτικά επίπεδα συνενώνονται (ως προς τη διάσταση των καναλιών) και ακολουθεί ένα επίπεδο max pooling με μέγεθος πυρήνα 2 και βήμα (stride) 1. Έπειτα ακολουθούν άλλα δύο συνελικτικά επίπεδα με μέγεθος πυρήνα ίσο με 3 και 128 κανάλια εξόδου το καθένα (στο Σχήμα 2.2 αναφέρονται ως Conv1D projections). Η έξοδος από αυτά τα δύο επίπεδα αθροίζεται με την αρχική είσοδο στο CBHG module μέσω ενός residual connection [He+16]. Ακολουθεί ένα δίκτυο που αποτελείται από τέσσερα fully connected επίπεδα με 128 νευρώνες το καθένα και συνάρτηση ενεργοποίησης τη ReLU. Το δίκτυο αυτό ονομάζεται Highway [SGS15] και η έξοδός του τροφοδοτείται σε ένα αμφίδρομο (bidirectional) αναδρομικό νευρωνικό δίκτυο GRU [Chu+14] με διάσταση εξόδου 128, το οποίο εξάγει την αναπαράσταση της ακολουθίας εισόδου. Κάθε στοιχείο αυτής της αναπαράστασης έχει διάσταση ίση με 256 (128 από κάθε κατεύθυνση του αμφίδρομου GRU). Κατά την εφαρμογή των επιπέδων στο δίκτυο CBHG διατηρείται η αρχική διάσταση της εισόδου, δηλαδή το μήκος της ακολουθίας των χαρακτήρων, χρησιμοποιώντας όπου χρειάζεται ανάλογο padding. Επίσης σε όλα τα συνελικτικά επίπεδα χρησιμοποιείται και batch normalization [IS15]. Η επιλογή της δομής του CBHG module στον encoder μειώνει την υπερεκπαίδευση (overfitting) και βελτιώνει την ποιότητα των παραγόμενων ηχητικών αποτελεσμάτων, όπως αναφέρεται και από τους συγγραφείς. Αφού έχει προκύψει η αναπαράσταση της ακολουθίας των χαρακτήρων από τον encoder, σειρά έχει ο decoder.

### Decoder

Ο decoder αναλαμβάνει την αποκωδικοποίηση της «κρυφής» αναπαράστασης προκειμένου να εξάγει ορισμένα frames από το φασματογράφημα. Συγκεκριμένα σε κάθε βήμα παράγει $r = 2$ frames αντί για ένα. Η τεχνική αυτή μειώνει στο μισό το πλήθος των βημάτων που χρειάζεται ο decoder για

να παράγει ολόκληρο το φασματογράφημα. Αν υποθέσουμε ότι το φασματογράφημα αποτελείται από 100 frames, τότε δίνοντας στον decoder ένα frame σε κάθε βήμα θα εκτελούσε συνολικά 100 βήματα. Με την εξαγωγή δύο frames σε κάθε βήμα, ο συνολικός χρόνος μειώνεται στα 50 βήματα. Η βασική ιδέα αυτής της μεθόδου είναι ότι η διάρκεια εκφώνησης ενός χαρακτήρα αντιστοιχεί σε πολύ κοντινά frames του φασματογραφήματος. Έτσι λοιπόν αρχικά δίνεται ως είσοδος στον decoder ένα frame με μηδενικές τιμές που υποδηλώνει την έναρξη της αποκωδικοποίησης (<GO> frame). Έπειτα ακολουθεί ένα δίκτυο Prenet με δύο fully connected επίπεδα ίδιας δομής με αυτό του encoder. Στη συνέχεια η έξοδος του Prenet τροφοδοτείται σε ένα αναδρομικό δίκτυο GRU, όπου στο Σχήμα 2.1 ονομάζεται Attention RNN και έχει διάσταση εξόδου ίση με 256. Η έξοδος του Attention RNN χρησιμοποιείται ως query σε ένα μηχανισμό προσοχής (content-based tanh attention [Vin+15]).

Ο μηχανισμός αυτός λειτουργεί ως εξής. Αν υποθέσουμε ότι $(\mathbf{h}_1, \ldots, \mathbf{h}_{T_A})$ με $\mathbf{h}_i \in \mathbb{R}^{256}$ είναι η αναπαράσταση που έχει προκύψει από τον encoder και $\mathbf{d}_t \in \mathbb{R}^{256}$ η έξοδος του Attention RNN στο βήμα t, τότε το αντίστοιχο διάνυσμα προσοχής $\mathbf{d}'_t$ υπολογίζεται σύμφωνα με τις παρακάτω σχέσεις:

$$u_i^t = \mathbf{v}^T \tanh(\mathbf{W}_1 \mathbf{h}_i + \mathbf{W}_2 \mathbf{d}_t), \quad i = 1, \ldots, T_A$$

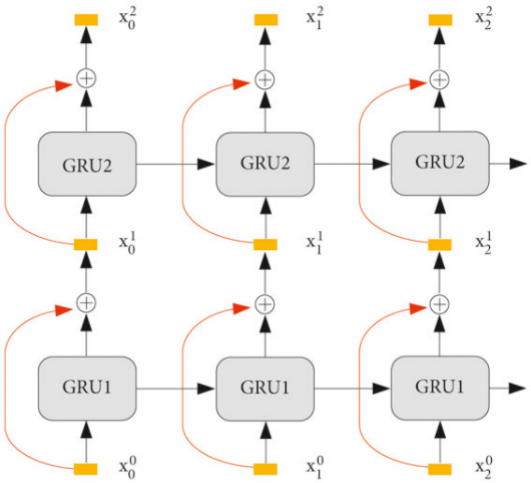
$$\alpha_i^t = \text{softmax}(u_i^t), \quad i = 1, \ldots, T_A$$

$$\mathbf{d}'_t = \sum_{i=1}^{T_A} \alpha_i^t \mathbf{h}_i.$$

Από τις δύο πρώτες σχέσεις προκύπτουν τα βάρη $\alpha_i^t$ με τα οποία σταθμίζονται τα στοιχεία της αναπαράστασης $(\mathbf{h}_1, \ldots, \mathbf{h}_{T_A})$. Οι πίνακες $\mathbf{W}_1, \mathbf{W}_2$ και το διάνυσμα $\mathbf{v}$ «μαθαίνονται» κατά την εκπαίδευση του δικτύου. Κάθε αρχικό βάρος $u_i^t$ υποδηλώνει το βαθμό στον οποίο ο decoder τη χρονική στιγμή t «εστιάζει την προσοχή» του στο στοιχείο $\mathbf{h}_i$ της κρυφής αναπαράστασης του encoder. Τα τελικά βάρη $\alpha_i^t$ προκύπτουν ύστερα από εφαρμογή της συνάρτησης softmax ώστε να παίρνουν θετικές τιμές και να αθροίζουν στη μονάδα. Έτσι λοιπόν σε κάθε βήμα t το διάνυσμα προσοχής $\mathbf{d}'_t \in \mathbb{R}^{256}$ προκύπτει ως ένας γραμμικός συνδυασμός των στοιχείων της αναπαράστασης που προκύπτει από τον encoder με συντελεστές τα βάρη $\alpha_i^t$.

Στη συνέχεια η έξοδος του Attention RNN $\mathbf{d}_t$ συνενώνεται με το διάνυσμα προσοχής $\mathbf{d}'_t$ και δίνονται ως είσοδος σε ένα δίκτυο GRU δύο επιπέδων που ονομάζεται Decoder RNN. Στο Decoder RNN χρησιμοποιούνται και residual connections, όπως φαίνεται στο Σχήμα 2.3. Τελικά η έξοδος από το Decoder RNN διέρχεται από ένα απλό fully connected επίπεδο για να προκύψουν δύο frames από το φασματογράφημα. Κατά την εκπαίδευση ο decoder σε κάθε βήμα δέχεται ως είσοδο το ground truth frame από το φασματογράφημα και σταματά την παραγωγή όταν όλα τα frames έχουν «περάσει» από τον decoder. Αντιθέτως κατά την συμπερασματολογία (inference) όπου δεν είναι γνωστό το ground truth φασματογράφημα, ο decoder σε κάθε βήμα t δέχεται ως είσοδο το δεύτερο από τα $r = 2$ frames που προέκυψαν στο προηγούμενο βήμα $t - 1$. Στο πρώτο βήμα ο decoder δέχεται ως είσοδο ένα frame με μηδενικές τιμές. Το inference σταματά όταν ο decoder ξεπεράσει ένα προκαθορισμένο μέγιστο πλήθος βημάτων. Εναλλακτικά σε κάθε βήμα μπορεί να χρησιμοποιηθεί ένα απλό fully connected επίπεδο που παράγει έναν αριθμό στο $[0, 1]$, το stop token. Όταν το stop token ξεπεράσει ένα προκαθορισμένο κατώφλι, π.χ. την τιμή 0.7 τότε ο decoder σταματά να παράγει άλλα frames.

Εφόσον έχουν προκύψει όλα τα frames από το φασματογράφημα, την τελική επεξεργασία τους αναλαμβάνει ένα δίκτυο CBHG ίδιας μορφής με αυτό του encoder. Σκοπός του είναι να μετατρέψει




Σχήμα 2.3: Αναδρομικό δίκτυο GRU με residual connections. [Xie+21]

το φασματογράφημα ώστε η συχνότητά του να βρίσκεται σε γραμμική κλίμακα προκειμένου ο αλγόριθμος Griffin-Lim να δώσει τα τελικά δείγματα της κυματομορφής. Επίσης ένα πλεονέκτημα του δικτύου CBHG μετά τον decoder, είναι ότι μπορεί να λάβει πληροφορία από ολόκληρη την αποκωδικοποιημένη ακολουθία των frames και να βελτιώσει το τελικό αποτέλεσμα.

### 2.2.2 Αποτελέσματα

Το μοντέλο Tacotron εκπαιδεύτηκε πάνω σε ένα (εσωτερικό) σύνολο δεδομένων στην Αγγλική γλώσσα που περιέχει συνολικά περίπου 24.6 ώρες ηχογραφημένης ομιλίας μαζί με τα αντίστοιχα κείμενα. Ως συνάρτηση ελαχιστοποίησης χρησιμοποιήθηκε το $l_1$-loss. Το $l_1$-loss υπολογίστηκε και για το τελικό φασματογράφημα του δικτύου CBHG, αλλά και για το φασματογράφημα που προκύπτει από τα frames που εξάγει ο decoder. Για την εκμάθηση των παραμέτρων του μοντέλου χρησιμοποιήθηκε ο βελτιστοποιητής Adam [KB14] με βήμα εκμάθησης 0.001 που μειωνόταν σταδιακά έπειτα από ορισμένο αριθμό εποχών. Τα αποτελέσματα του μοντέλου Tacotron είναι αρκετά ικανοποιητικά αν λάβουμε υπόψιν τη μετρική του MOS (Mean Opinion Score). Σύμφωνα με αυτή τη μετρική ορισμένοι βαθμολογητές αξιολογούν τις συνθετικές εκφωνήσεις του μοντέλου αλλά και τις πραγματικές, δίνοντας μια βαθμολογία στην κλίμακα 1 έως 5, όπου όσο υψηλότερη είναι η βαθμολογία τόσο καλύτερη είναι και η ποιότητα του ηχητικού δείγματος. Συγκεκριμένα το Tacotron πετυχαίνει MOS περίπου 3.82 που είναι υψηλότερο από ενός παραμετρικού συστήματος [Zen+16] με τιμή περίπου 3.69 και είναι κοντά στο MOS ενός συναθροιστικού συστήματος [Gon+16] με τιμή περίπου 4.09. Συμπεραίνουμε λοιπόν ότι το πρόβλημα της σύνθεσης φωνής από κείμενο μπορεί να αντιμετωπιστεί αποτελεσματικά από ένα end-to-end μοντέλο όπως το Tacotron, το οποίο με περισσότερη βελτίωση μπορεί να αντικαταστήσει άλλες παραμετρικές ή συναθροιστικές προσεγγίσεις. Στην επόμενη ενότητα παρουσιάζουμε μια βελτιωμένη εκδοχή αυτού του μοντέλου.

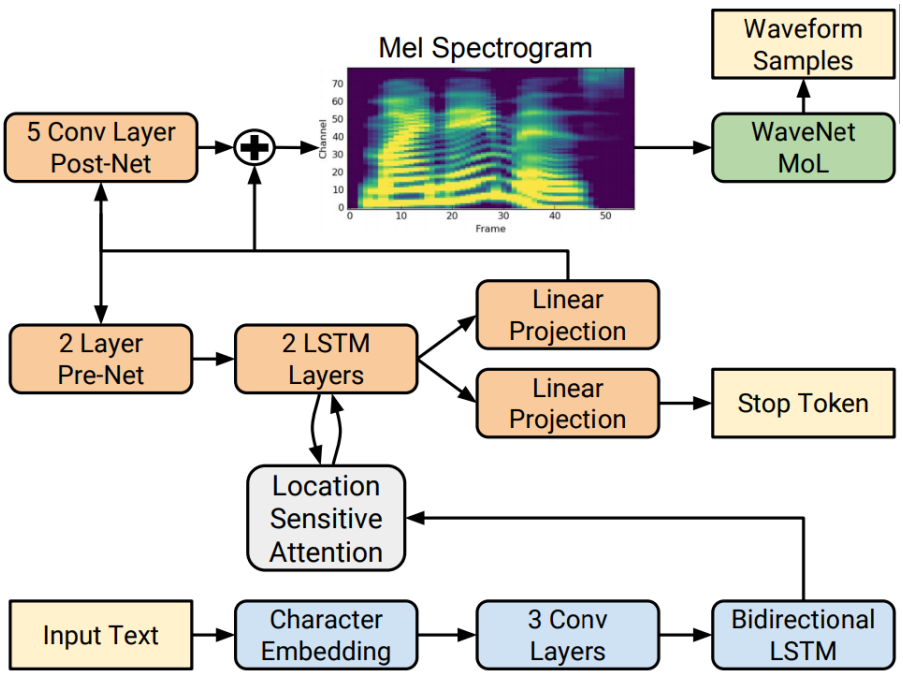
## 2.3 Tacotron2

Το μοντέλο Tacotron2 [She+18] είναι μια end-to-end προσέγγιση για τη σύνθεση φωνής από κείμενο και αποτελεί μια βελτιωμένη εκδοχή του απλού μοντέλου Tacotron. Αποτελείται από δύο βασικά επιμέρους τμήματα. Το πρώτο είναι ένα νευρωνικό δίκτυο της μορφής sequence-to-

sequence [SVL14] το οποίο έχει στόχο να μετατρέψει το κείμενο εισόδου σε ένα φασματογράφημα στη κλίμακα mel. Το δεύτερο τμήμα αποτελείται από ένα μοντέλο vocoder που επεξεργάζεται το φασματογράφημα του πρώτου δικτύου και παράγει δείγματα από την κυματομορφή του ηχητικού σήματος. Ως neural vocoder χρησιμοποιείται μια τροποποιημένη εκδοχή του μοντέλου WaveNet [Oor+16].

### 2.3.1 Αρχιτεκτονική του μοντέλου

Η αρχιτεκτονική του μοντέλου Tacotron2, η οποία παρουσιάζεται στο Σχήμα 2.4, έχει παρόμοια δομή με αυτή του Tacotron. Όπως αναφέραμε, το πρώτο μέρος είναι ένα sequence-to-sequence δίκτυο που μετατρέπει ένα κείμενο (σύνολο χαρακτήρων) σε ένα φασματογράφημα στην κλίμακα mel. Το δίκτυο αυτό χρησιμοποιεί έναν encoder που δέχεται το κείμενο εισόδου, έστω $\mathbf{x} = (x_1, \ldots, x_L)$ και το μετατρέπει σε μια ακολουθία «κρυφών» (hidden) αναπαραστάσεων $\mathbf{h} = (h_1, \ldots, h_L)$. Για να το επιτύχει χρησιμοποιεί αρχικά ένα embedding layer, ώστε οι χαρακτήρες που αποτελούν το κείμενο να γίνουν διανύσματα σταθερού μήκους με συνεχείς τιμές. Οι χαρακτήρες σε ένα κείμενο μπορούν να αναπαρασταθούν από μια ακολουθία one-hot διανυσμάτων, όπου καθένα έχει μήκος $N_{symbols}$. Το $N_{symbols}$ υποδηλώνει το πλήθος των μοναδικών χαρακτήρων στο σύνολο δεδομένων. Επίσης το μήκος των διανυσμάτων embedding έχει την τιμή 512.

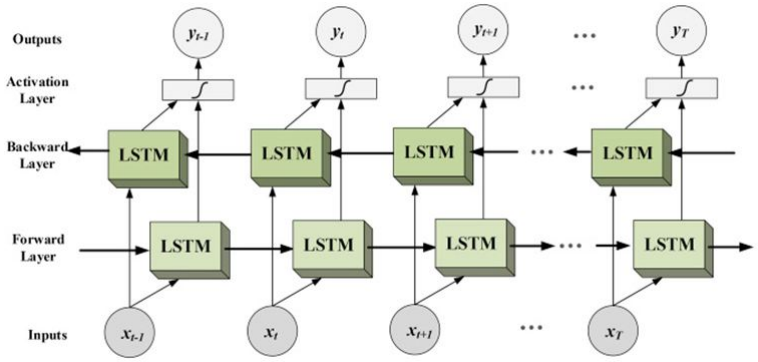



Σχήμα 2.4: Αρχιτεκτονική του μοντέλου Tacotron2. [She+18]

Ύστερα από το embedding layer χρησιμοποιούνται τρία stacked συνελικτικά (convolutional) επίπεδα, όπου το καθένα έχει μέγεθος πυρήνα (kernel size) 5, βήμα (stride) 1 και δρα κατά μήκος της ακολουθίας των χαρακτήρων, ώστε να μοντελοποιηθούν χρονικές εξαρτήσεις σε αυτήν. Κάθε συνελικτικό στρώμα χρησιμοποιεί ένα convolutional layer με 512 φίλτρα καθώς επίσης και padding, ώστε να διατηρείται το αρχικό μήκος της ακολουθίας εισόδου. Έπειτα εφαρμόζεται ένα Batch Normalization layer [IS15] και η συνάρτηση ενεργοποίησης ReLU.

Εν συνεχεία ακολουθεί το βασικό μέρος του encoder που είναι ένα αμφίδρομο (bidirectional) αναδρομικό νευρωνικό δίκτυο και συγκεκριμένα ένα LSTM [HS97]. Ένα απλό (unidirectional)

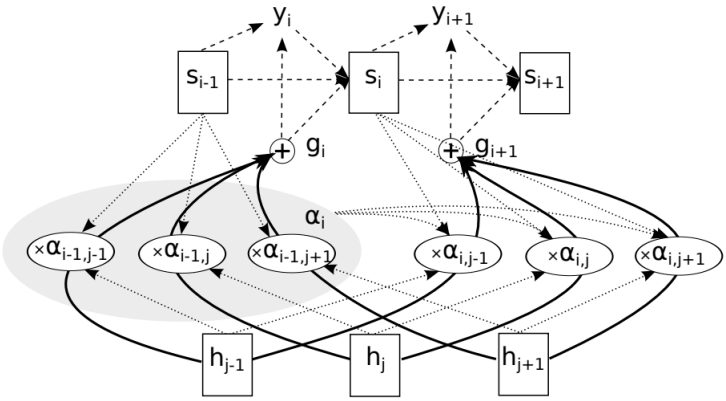
LSTM επεξεργάζεται την είσοδό του μόνο προς τα εμπρός, οπότε τα hidden features που παράγονται σε κάθε βήμα έχουν πληροφορία μόνο από τα προηγούμενα χρονικά βήματα. Αντιθέτως σε ένα bidirectional LSTM, η έξοδος σε κάθε χρονικό βήμα προκύπτει από τη συνένωση των εξόδων από το forward και το backward δίκτυο, έχοντας έτσι πληροφορία και από τα προηγούμενα αλλά και από τα επόμενα βήματα και κατά συνέπεια μπορεί να παράγει πιο αντιπροσωπευτικά hidden features της ακολουθίας εισόδου. Στο Σχήμα 2.5 φαίνεται η δομή ενός bidirectional LSTM. Στο μοντέλο Tacotron2, το bidirectional LSTM του encoder χρησιμοποιεί 512 κρυφούς νευρώνες, 256 για το forward και 256 για το backward LSTM.




Σχήμα 2.5: Αμφίδρομο (bidirectional) δίκτυο LSTM. Η έξοδος του δικτύου σε κάθε βήμα προκύπτει από τη συνένωση των εξόδων του forward και του backward pass, χρησιμοποιώντας έτσι πληροφορία από όλη την ακολουθία εισόδου.

Ένα βασικό στοιχείο που χρησιμοποιείται στο μοντέλο είναι ο μηχανισμός Attention [BCB14]. Ο decoder σε κάθε βήμα παράγει ένα frame από το φασματογράφημα. Συγκεκριμένα για να προκύψει η έξοδος στο βήμα $i$, έστω $\mathbf{s}_i$, ο decoder εκμεταλλεύεται την πληροφορία των κρυφών αναπαραστάσεων $\mathbf{h} = (h_1, \ldots, h_L)$ του encoder, καθώς επίσης και την έξοδο του decoder $\mathbf{s}_{i-1}$ του προηγούμενου βήματος. Για να το επιτύχει αυτό χρησιμοποιεί έναν μηχανισμό προσοχής, προκειμένου να παράγει ένα διάνυσμα προσοχής (attention context vector) $\mathbf{w}_i$ που θα χρησιμοποιηθεί και αυτό ως είσοδος στον decoder. Πιο συγκεκριμένα, έχοντας τα hidden features $\mathbf{h} = (h_1, \ldots, h_L)$ του encoder θα πρέπει να δημιουργηθούν τα βάρη $\boldsymbol{\alpha}_i = (\alpha_{i,1}, \ldots, \alpha_{i,L})$ (alignment), τα οποία αφού πολλαπλασιαστούν με τα hidden features θα δώσουν το διάνυσμα προσοχής. Το διάνυσμα αυτό θα εμπεριέχει πληροφορία από όλη την ακολουθία των κρυφών αναπαραστάσεων (άρα και από την ακολουθία χαρακτήρων), έχοντας «εστιάσει την προσοχή» στα σημεία της εισόδου που παίζουν το σημαντικότερο ρόλο για την παραγωγή του frame $\mathbf{s}_i$ τη χρονική στιγμή $i$. Ο μηχανισμός Attention που χρησιμοποιείται στο μοντέλο Tacotron2 είναι μια παραλλαγή του additive ή Bahdanau μηχανισμού προσοχής και ονομάζεται location-sensitive attention [Cho+15] (βλ. Σχήμα 2.6). Για το frame $\mathbf{s}_i$ χρησιμοποιούμε το συνδυασμό (concatentation) του alignment $\boldsymbol{\alpha}_{i-1}$ και των αθροιστικών (cumulative) alignments που έχουν προκύψει μέχρι και το βήμα $i-1$, έστω $\boldsymbol{\alpha}_{cum,i-1}$ καθώς επίσης και το frame $\mathbf{s}_{i-1}$ και ορισμένα location features $\mathbf{f}_i$. Τα location features προκύπτουν ύστερα από συνέλιξη του (concatenated) alignment $\boldsymbol{\alpha}_{cat,i-1} = [\boldsymbol{\alpha}_{i-1}; \boldsymbol{\alpha}_{cum,i-1}]$ με έναν πίνακα $\mathbf{F} \in \mathbb{R}^{k \times r}$, δηλαδή $\mathbf{f}_i = F * \boldsymbol{\alpha}_{cat,i-1}$. Τελικά το alignment $\boldsymbol{\alpha}_i = (\alpha_{i,1}, \ldots, \alpha_{i,L})$ του βήματος $i$ προκύπτει σύμφωνα με τη σχέση:

$$\alpha_{i,j} = \text{softmax}(e_{i,j}), \text{ για } j = 1, \ldots, L, \tag{2.3.1}$$




Σχήμα 2.6: Location sensitive attention [Cho+15]. Για τον υπολογισμό του alignment $\boldsymbol{\alpha}_i$ χρησιμοποιείται το διάνυσμα $\mathbf{s}_{i-1}$, τα hidden features $\mathbf{h}$ και το προηγούμενο alignment $\boldsymbol{\alpha}_{i-1}$. Ύστερα υπολογίζεται το διάνυσμα προσοχής (εδώ $\mathbf{g}_i$) από το $\boldsymbol{\alpha}_i$ και τα hidden features $\mathbf{h}$, το οποίο με τη σειρά του χρησιμοποιείται για την έξοδο $\mathbf{s}_i$.

με

$$e_{i,j} = \mathbf{v}^T \tanh(\mathbf{W}\mathbf{s}_{i-1} + \mathbf{V}h_j + \mathbf{U}f_{i,j} + \mathbf{b}), \text{ για } j = 1, \ldots, L, \tag{2.3.2}$$

όπου οι $\mathbf{W}, \mathbf{V}, \mathbf{U}$ είναι πίνακες και τα $\mathbf{v}, \mathbf{b}$ διανύσματα. Για τον υπολογισμό των location features χρησιμοποιούνται 32 φίλτρα με μέγεθος πυρήνα 31 και για τον υπολογισμό των ενεργειών $e_{i,j}$ οι πίνακες $\mathbf{W}, \mathbf{V}, \mathbf{U}$ προβάλουν τα διανύσματα $\mathbf{s}_{i-1}, h_j, f_{i,j}$ αντίστοιχα, ώστε να έχουν διάσταση 128. Τελικά το διάνυσμα προσοχής που θα χρησιμοποιηθεί από τον decoder στο βήμα $i$ προκύπτει από τη σχέση $\mathbf{w}_i = \boldsymbol{\alpha}_i\mathbf{h}$ και έχει διάσταση ίση με 512 (η k-συνιστώσα του διανύσματος είναι $\mathbf{w}_{i,k} = \sum_{j=1}^{L} \alpha_{i,j}h_{j,k}$ για $k = 1, \ldots, 512$).

Ο decoder με τη σειρά του είναι ένα δίκτυο όπου σε κάθε βήμα παράγει ένα frame από το φασματογράφημα στην κλίμακα mel, το οποίο είναι ένα διάνυσμα με 80 τιμές (50ms μέγεθος frame, με βήμα 12ms). Αρχικά το frame που προκύπτει στο προηγούμενα βήμα, έστω $\mathbf{s}_{i-1}$, περνά από ένα fully connected δίκτυο που ονομάζεται pre-net, και αποτελείται από δύο fully connected layers με 256 νευρώνες και συνάρτηση ενεργοποίησης τη ReLU. Η έξοδος από το δίκτυο αυτό συνενώνεται με το διάνυσμα προσοχής (attention context) $\mathbf{w}_i$ και εισέρχονται σε δύο stacked LSTMs όπου το καθένα έχει 1024 κρυφούς νευρώνες. Ύστερα το διάνυσμα προσοχής συνενώνεται με την έξοδο από το πρώτο LSTM και εισέρχονται στο δεύτερο LSTM. Η έξοδος από το δεύτερο LSTM συνενώνεται εκ νέου με το attention context διάνυσμα και εισέρχεται σε ένα fully connected επίπεδο που παράγει ένα frame του φασματογραφήματος. Για να βελτιωθεί το frame που παράχθηκε χρησιμοποιείται επιπλέον ένα δίκτυο με πέντε συνελικτικά στρώματα, όπου το καθένα έχει 512 φίλτρα και το μέγεθος του πυρήνα είναι 5. Τα συνελικτικά αυτά στρώματα ακολουθούνται από ένα επίπεδο Batch Normalization και συνάρτηση ενεργοποίησης την υπερβολική εφαπτομένη tanh (σε όλα εκτός από το τελευταίο στρώμα). Το δίκτυο αυτό ονομάζεται Post-net και το αποτέλεσμά του προστίθεται στην έξοδο του προηγούμενου fully connected επιπέδου για να υπάρξει βελτίωση του τελικού frame. Η τεχνική αυτή είναι γνωστή και ως residual connection [He+16]. Κατά το inference χρησιμοποιείται και ένα επιπλέον fully connected επίπεδο με σιγμοειδή συνάρτηση ενεργοποίησης, που λαμβάνει ως είσοδο την έξοδο από το δεύτερο LSTM μαζί με το διάνυσμα προσοχής και παράγει έναν αριθμό που εκφράζει την πιθανότητα τερματισμού ("stop token"). Το μοντέλο σταματά να παράγει άλλα frames όταν ο αριθμός αυτός ξεπεράσει την τιμή 0.5. Για να γίνει ομαλοποίηση του δικτύου κατά την εκπαίδευση χρησιμοποιείται dropout

[Sri+14] με πιθανότητα 0.5 στα συνελικτικά επίπεδα και zoneout [Kru+16] με πιθανότητα 0.1 στα LSTMs.

### 2.3.2 Εκπαίδευση και αξιολόγηση του μοντέλου

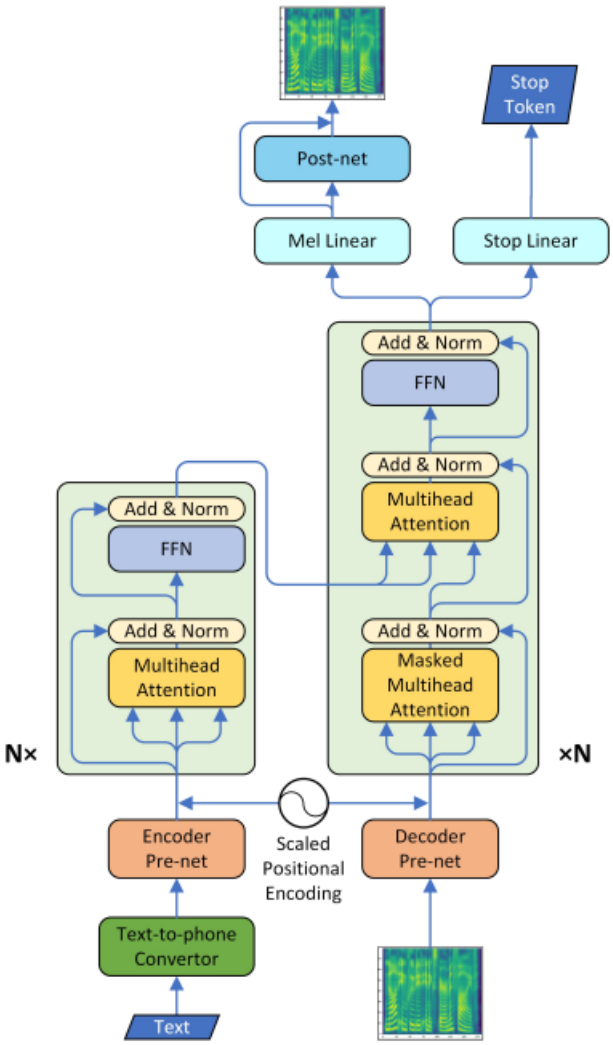
Η διαδικασία εκπαίδευσης του μοντέλου αποτελείται από δύο στάδια. Σε πρώτη φάση εκπαιδεύεται το δίκτυο πρόβλεψης του φασματογραφήματος και ύστερα γίνεται ξεχωριστή εκπαίδευση του τροποποιημένου μοντέλου WaveNet για την παραγωγή των δειγμάτων του ηχητικού σήματος. Για την εκπαίδευση του πρώτου δικτύου χρησιμοποιείται η τεχνική teacher forcing. Σύμφωνα με αυτή ο decoder δέχεται ως είσοδο ένα ground truth frame από το φασματογράφημα προκειμένου να προβλέψει το frame για την επόμενη χρονική στιγμή, αντί να δέχεται ως είσοδο το frame που προέβλεψε στο προηγούμενο βήμα. Η μετρική που χρησιμοποιείται κατά την εκπαίδευση είναι το μέσο τετραγωνικό σφάλμα (MSE) και υπολογίζεται πριν και μετά από την έξοδο του δικτύου post-net. Κατά τη διαδικασία του inference, σε κάθε βήμα ο decoder χρησιμοποιεί την πρόβλεψη που έκανε στο προηγούμενο βήμα, προκειμένου να παράγει ένα frame του φασματογραφήματος, αφού σε αυτή την περίπτωση δεν είναι διαθέσιμα τα ground truth frames. Για την εκπαίδευση του μοντέλου Tacotron2 χρησιμοποιήθηκε το σύνολο δεδομένων US English [Wan+17] που έχει 24.6 ηχογραφημένες ώρες μαζί με τα αντίστοιχα κείμενα. Για την αξιολόγηση των αποτελεσμάτων του μοντέλου χρησιμοποιήθηκαν 100 δείγματα από το ίδιο σύνολο δεδομένων όπου καθένα βαθμολογήθηκε από τουλάχιστον οκτώ άτομα (ανεξάρτητα) στην κλίμακα 1-5, με διαβαθμίσεις 0.5. Τέλος υπολογίστηκε ένας μέσος όρος των βαθμολογιών για όλα τα δείγματα στο σύνολο αξιολόγησης. Η μετρική αυτή ονομάζεται Mean Opinion Score (MOS) και έλαβε την τιμή 4.526, η οποία είναι καλύτερη από προηγούμενες προσεγγίσεις στο πρόβλημα, όπως το απλό Tacotron με MOS $\approx 4.001$ που χρησιμοποιεί τον αλγόριθμο Griffin-Lim [GL84] για τη σύνθεση του ηχητικού σήματος. Η τιμή του MOS για τα πραγματικά audio ήταν περίπου 4.582.

## 2.4 Σύνθεση φωνής με το μοντέλο Transformer

Αν και το μοντέλο Tacotron2 πετυχαίνει state of the art αποτελέσματα στο πρόβλημα της σύνθεσης φωνής από κείμενο, εντούτοις έχει και αυτό κάποια μειονεκτήματα. Το πρώτο αφορά την ταχύτητά του κατά τη διάρκεια της εκπαίδευσης και της συμπερασματολογίας (inference), μιας και στην αρχιτεκτονική του, ο encoder και ο decoder αποτελούνται από αναδρομικά νευρωνικά δίκτυα (LSTMs). Ως γνωστόν τα δίκτυα αυτά επεξεργάζονται μια ακολουθία εισόδου σειριακά απαιτώντας έτσι σημαντικό χρόνο κατά την εκπαίδευση. Επιπλέον λόγω της δομής τους, είναι δύσκολο να μοντελοποιήσουν μακροπρόθεσμες εξαρτήσεις μεταξύ των στοιχείων μιας ακολουθίας εισόδου, αφού παραδείγματος χάρη για την έξοδο ενός LSTM τη χρονική στιγμή $t$, λαμβάνεται υπόψιν κυρίως το hidden state του προηγούμενου βήματος $t-1$.[1] Το μοντέλο Transformer [Vas+17] δίνει λύση σε τέτοιου είδους περιορισμούς αφού η αρχιτεκτονική του δε βασίζεται πλέον σε αναδρομικά δίκτυα, αλλά αυτά αντικαθίστανται από μηχανισμούς προσοχής, όπως ο Multi-Head Attention. Σε αντίθεση με ένα δίκτυο LSTM, το μοντέλο Transformer δέχεται ολόκληρη την ακολουθία εισόδου εξαρχής και έτσι λύνεται το πρόβλημα της αποδοτικότητας, εφόσον μπορεί να γίνει ταχύτερη εκπαίδευση με την παραλληλοποίηση του δικτύου. Επιπλέον είναι δυνατόν να μοντελοποιηθούν και μακροπρόθεσμες χρονικές εξαρτήσεις στην ακολουθία εισόδου μέσω του

[1] Αν και το hidden state $\mathbf{h}_{t-1}$ διατηρεί πληροφορία και από πιο προηγούμενα βήματα, η πληροφορία αυτή εξασθενεί προχωρώντας στα επόμενα βήματα.

μηχανισμού προσοχής που χρησιμοποιείται. Για τους λόγους αυτούς ερευνητές από τη Microsoft πρότειναν μια αρχιτεκτονική για το πρόβλημα της σύνθεσης φωνής από κείμενο, η οποία βασίζεται αποκλειστικά στο μοντέλο Transformer [Li+19]. Παρακάτω παρουσιάζουμε τα βασικά στοιχεία που δομούν την αρχιτεκτονική του, η οποία φαίνεται και στο Σχήμα 2.7.




Σχήμα 2.7: Η αρχιτεκτονική του μοντέλου Transformer [Li+19] για το πρόβλημα text-to-speech. (Αριστερά) Ο encoder μετατρέπει το κείμενο εισόδου (text) σε μια αναπαράσταση (context), την οποία ο decoder (δεξιά) χρησιμοποιεί για να εξάγει το φασματογράφημα.

### 2.4.1 Αρχιτεκτονική του μοντέλου

Το μοντέλο έχει ως στόχο να μετατρέψει ένα κείμενο σε μία ακολουθία δειγμάτων που αποτελούν το ηχητικό σήμα. Για να το επιτύχει αυτό χρησιμοποιεί δύο δίκτυα. Το πρώτο δέχεται ένα κείμενο και το μετατρέπει στο αντίστοιχο φασματογράφημα μέσω μιας αρχιτεκτονικής βασισμένη στον Transformer. Το δεύτερο δίκτυο είναι ένας vocoder που μετατρέπει το παραγόμενο φασματογράφημα σε δείγματα της κυματομορφής του ηχητικού σήματος. Όπως και στο μοντέλο Tacotron2, ως neural vocoder χρησιμοποιείται μια τροποποιημένη εκδοχή του WaveNet.

Το πρώτο δίκτυο (Transformer) αποτελείται από έναν encoder και έναν decoder. Ο encoder μετατρέπει την ακολουθία εισόδου, έστω $\mathbf{x} = (x_1, \ldots, x_L)$ σε μια «κρυφή» (hidden) αναπαράσταση $\mathbf{h} = (h_1, \ldots, h_L)$. Εν συνεχεία ο decoder παράγει την ακολουθία χαρακτήρων εξόδου $\mathbf{y} = (y_1, \ldots, y_S)$, όπου σε κάθε βήμα εξάγει ένα frame από το φασματογράφημα, χρησιμοποιώντας μόνο την πληροφορία $\mathbf{h}$ που έχει προκύψει από τον encoder, καθώς και την έξοδο του decoder στο προηγούμενο βήμα. Σε πρώτη φάση όμως, θα πρέπει το κείμενο εισόδου να μετατραπεί σε μια ακολουθία από φωνήματα (phonemes) τα οποία θα χρησιμοποιηθούν ως είσοδος στο μοντέλο. Η επιλογή να χρησιμοποιηθούν φωνήματα αντί για χαρακτήρες (όπως στο Tacotron2) έγκειται στο γεγονός ότι ορισμένοι χαρακτήρες προφέρονται με διαφορετικό τρόπο σε άλλες λέξεις (π.χ. το γράμμα "o" στη λέξη "shoot" προφέρεται διαφορετικά απ' ότι στην λέξη "on"). Η διαφοροποίηση αυτή δε μπορεί να μοντελοποιηθεί ικανοποιητικά μόνο με τη χρήση χαρακτήρων, στην περίπτωση όπου στα δεδομένα υπάρχουν λίγα δείγματα που προφέρονται με ένα συγκεκριμένο τρόπο. Εφόσον το κείμενο μετατραπεί σε μια ακολουθία από φωνήματα θα πρέπει να εισαχθεί στον encoder για την εξαγωγή των hidden features.

Πριν τον encoder χρησιμοποιείται ένα δίκτυο που ονομάζεται encoder-prenet και επεξεργάζεται τα φωνήματα στην είσοδο για να παράγει embeddings με διάσταση 512. Το δίκτυο αυτό χρησιμοποιεί τρία stacked συνελικτικά στρώματα με 512 φίλτρα το καθένα καθώς και Batch Normalization layers, συνάρτηση ενεργοποίησης τη ReLU και Dropout (όπως και στο Tacotron2). Στο τέλος του encoder-prenet προστίθεται και ένα fully connected επίπεδο. Παρομοίως, πριν τον decoder εφαρμόζεται ένα δίκτυο που ονομάζεται decoder-prenet. Το decoder-prenet αποτελείται από δύο fully connected επίπεδα με 256 νευρώνες το καθένα με συνάρτηση ενεργοποίησης τη ReLU και προβάλλει τα frames του φασματογραφήματος σε διανύσματα ίδιας διάστασης 512 με τα embeddings των φωνημάτων του encoder. Αυτό βοηθά τη λειτουργία του μηχανισμού προσοχής στον decoder, ο οποίος δέχεται στοιχεία και από τον decoder αλλά και από τον encoder (συγκεκριμένα τα hidden states).

Όπως έχουμε αναφέρει, ο Transformer βασίζεται αποκλειστικά σε μηχανισμούς προσοχής χωρίς να κάνει χρήση αναδρομικών νευρωνικών δικτύων όπως RNN, LSTM και GRU. Τα δίκτυα αυτά έχουν τη δυνατότητα να λαμβάνουν πληροφορία για τη θέση ενός διανύσματος στην ακολουθία εισόδου, αφού όλα τα στοιχεία της επεξεργάζονται σειριακά. Αντιθέτως ο Transformer δέχεται ολόκληρη την ακολουθία εισόδου εξαρχής και έτσι το μοντέλο δεν έχει γνώση για τη θέση κάθε στοιχείου σε αυτήν. Προκειμένου λοιπόν να ληφθεί υπόψιν και η θέση κάθε φωνήματος στο μοντέλο, χρησιμοποιείται το Positional encoding, όπου κάθε φώνημα μετατρέπεται σε ένα διάνυσμα διαστάσεων $d_{model} = 512$ σύμφωνα με τη σχέση:

$$\mathrm{PE}_{(pos,2i)} = \sin\left(\frac{pos}{10000^{\frac{2i}{d_{model}}}}\right),$$

$$\mathrm{PE}_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{\frac{2i}{d_{model}}}}\right).$$

Όπου $pos$ είναι η θέση του φωνήματος στην ακολουθία εισόδου και το $i$ αντιστοιχεί στην $i-$οστή διάσταση του διανύσματος. Στο πρόβλημα TTS τα φωνήματα και τα frames του φασματογραφήματος βρίσκονται σε διαφορετική κλίμακα κάτι που μπορεί να περιορίσει τη λειτουργία του encoder και του decoder prenet. Για το λόγο αυτό χρησιμοποιείται μια παραλλαγή του στατικού positional encoding που ονομάζεται scaled positional encoding και έχει μια παράμετρο προς εκμάθηση, με στόχο τα embeddings του encoder και του decoder να έρθουν στην ίδια κλίμακα. Το scaled

positional encoding δίνεται από τη σχέση:

$$x_i = \text{prenet}(\text{phoneme}_i) + \alpha \text{PE}(i), \tag{2.4.1}$$

όπου α είναι η παράμετρος προς εκμάθηση. Τα embeddings $x_i$ παρέχουν την πληροφορία για τις θέσεις των φωνημάτων στην ακολουθία εισόδου και επεξεργάζονται από τον encoder για την εξαγωγή των κρυφών αναπαραστάσεων. Πιο συγκεκριμένα ο encoder αποτελείται από $N_e$ στοιβαγμένα encoder blocks, όπου κάθε ένα παίρνει ως είσοδο την έξοδο του προηγούμενου block και το τελευταίο από αυτά παράγει την κρυφή αναπαράσταση $\mathbf{h} = (h_1, \ldots, h_L)$ της ακολουθίας εισόδου $\mathbf{x} = (x_1, \ldots, x_L)$. Τα δύο βασικά μέρη ενός encoder block είναι το Multi-Head Attention και το Position-wise Feed-Forward Network. Πρωτού περιγράψουμε τη δομή τους θα δούμε τη λειτουργία του βασικού μηχανισμού προσοχής Scaled Dot-Product Attention.

### Scaled Dot-Product Attention

Ο μηχανισμός προσοχής Scaled Dot-Product Attention χρησιμοποιείται για να συσχετίσει τα στοιχεία της ακολουθίας εισόδου μεταξύ τους προκειμένου να εξάγει αναπαραστάσεις οι οποίες περιέχουν χρήσιμη και «ανάμεικτη» πληροφορία από ολόκληρη την ακολουθία. Ως είσοδο δέχεται τα queries ($\mathbf{Q}$), keys ($\mathbf{K}$) και values ($\mathbf{V}$) όπου έχουν διαστάσεις $\mathbf{Q} \in \mathbb{R}^{L_q \times d_q}$, $\mathbf{K} \in \mathbb{R}^{L_k \times d_k}$ και $\mathbf{V} \in \mathbb{R}^{L_v \times d_v}$. Συνήθως η πρώτη διάσταση των queries, keys και values αναφέρεται στο πλήθος των χρονικών βημάτων της ακολουθίας, επομένως $L_q = L_k = L_v = L$ και η δεύτερη διάσταση υποδηλώνει το πλήθος των χαρακτηριστικών σε κάθε βήμα, δηλαδή $d_q = d_k = d_v$. Τα queries, keys και values συνδέονται μέσω του μηχανισμού προσοχής που δίνεται από τον τύπο:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V}. \tag{2.4.2}$$

Το διάνυσμα εξόδου self-attention προκύπτει ως ένα σταθμισμένο άθροισμα των values με συγκεκριμένα βάρη που παίρνουν θετικές τιμές και αθροίζουν στη μονάδα. Για τον υπολογισμό των βαρών θεωρούμε το γινόμενο του πίνακα των queries $\mathbf{Q}$ με τον ανάστροφο πίνακα των keys $\mathbf{K}^T$ και διαιρούμε με την ποσότητα $\sqrt{d_k}$. Έπειτα εφαρμόζουμε τη συνάρτηση softmax προκειμένου να πάρουμε βάρη που αθροίζουν στη μονάδα. Ο παράγοντας $1/\sqrt{d_k}$ χρησιμοποιείται προκειμένου να έχουμε πιο σταθερές παραγώγους και κατά συνέπεια καλύτερη εκπαίδευση. Όσο μεγαλύτερη είναι η διάσταση των queries και των keys, τόσο μεγαλύτερο ενδέχεται να γίνει το εσωτερικό τους γινόμενο $\mathbf{Q}\mathbf{K}^T$. Τότε η παράγωγος της συνάρτησης softmax ελαττώνεται ή τείνει προς το 0 και αυτό έχει ως αποτέλεσμα τα βάρη του δικτύου να ανανεώνονται ελάχιστα ή και καθόλου, γεγονός που αποτρέπει την εκπαίδευση του μοντέλου.

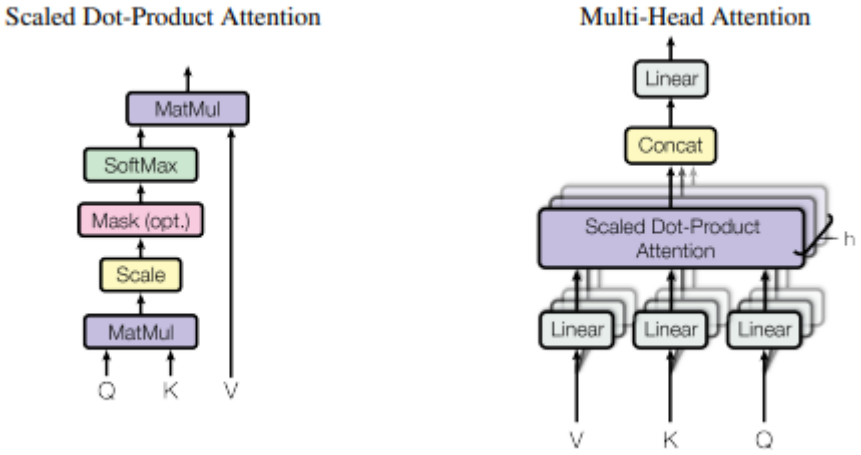
### Multi-Head Attention

Ο μηχανισμός scaled dot-product attention μπορεί να χρησιμοποιηθεί παραπάνω από μία φορές προκειμένου το μοντέλο να εξάγει περισσότερη πληροφορία. Μετατρέποντας τα queries, keys και values μέσω ενός γραμμικού μετασχηματισμού και εφαρμόζοντας h-φορές scaled dot-product attention, έχουμε το μηχανισμό Multi-Head Attention (το h υποδηλώνει τον αριθμό των heads). Συγκεκριμένα για κάθε head θεωρούμε τα γινόμενα $\mathbf{Q}\mathbf{W}_i^Q$, $\mathbf{K}\mathbf{W}_i^K$ και $\mathbf{V}\mathbf{W}_i^V$ με $\mathbf{W}_i^Q \in \mathbb{R}^{d_{model} \times d_q}$, $\mathbf{W}_i^K \in \mathbb{R}^{d_{model} \times d_k}$ και $\mathbf{W}_i^V \in \mathbb{R}^{d_{model} \times d_v}$ για $i = 1, \ldots, h$ και εφαρμόζουμε το scaled dot-product attention. Εν συνεχεία συνενώνουμε τα αποτελέσματα από όλα τα heads και κάνουμε έναν γραμμικό μετασχηματισμό ώστε τα διανύσματα εξόδου του Multi-Head Attention να έχουν διάσταση

$d_{model}$. Η έξοδος προκύπτει σύμφωνα με τη σχέση:

$$\text{MultiHead}(\mathbf{Q},\mathbf{K},\mathbf{V}) = \text{Concat}(\text{head}_1,\ldots,\text{head}_h)\mathbf{W}^O, \quad \text{με head}_i = \text{Attention}(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V) \tag{2.4.3}$$

και $\mathbf{W}^O \in \mathbb{R}^{hd_v \times d_{model}}$. Στο μοντέλο Transformer ισχύει $d_q = d_k = d_v = d_{model}/h$. Οι μηχανισμοί Scaled Dot-Product Attention και Multi-Head Attention φαίνονται στο Σχήμα 2.8.




Σχήμα 2.8: Μηχανισμοί προσοχής στο μοντέλο Transformer. (Αριστερά) Scaled Dot-Product Attention. (Δεξιά) Multi-Head Attention με h-heads. [Vas+17]

### Position-wise Feed-Forward Network

Αφού εφαρμόσουμε το μηχανισμό Multi-Head Attention συνεχίζουμε με ένα δίκτυο πρόσθιας τροφοδότησης που αποτελείται από δύο γραμμικούς μετασχηματισμούς με συνάρτηση ενεργοποίησης ReLU στο πρώτο επίπεδο. Το δίκτυο αυτό επεξεργάζεται κάθε διάνυσμα της εξόδου Multi-Head Attention με διάσταση $d_{model}$ ανεξάρτητα (position-wise), προβάλλοντάς το αρχικά σε διάσταση $d_{ff} = 4d_{model}$ και έπειτα ξανά στη διάσταση $d_{model}$. Η σχέση που περιγράφει το δίκτυο πρόσθιας τροφοδότησης είναι:

$$\text{FFN}(\mathbf{x}) = \text{ReLU}(\mathbf{x}\mathbf{W}_1 + \mathbf{b}_1)\mathbf{W}_2 + \mathbf{b}_2, \tag{2.4.4}$$

όπου $\mathbf{W}_1 \in \mathbb{R}^{d_{model}\times d_{ff}}$, $\mathbf{b}_1 \in \mathbb{R}^{d_{ff}}$, $\mathbf{W}_2 \in \mathbb{R}^{d_{ff}\times d_{model}}$ και $\mathbf{b}_2 \in \mathbb{R}^{d_{model}}$.

Κάθε encoder block δέχεται ως είσοδο τα queries, keys και values και αρχικά εφαρμόζει ένα επίπεδο κανονικοποίησης-layer normalization [BKH16]. Για κάθε διάνυσμα $\mathbf{u}$ το layer normalization υπολογίζεται σύμφωνα με τη σχέση:

$$\mathbf{u}_{norm} = \frac{\mathbf{u} - \mathbb{E}[\mathbf{u}]}{\sqrt{\text{Var}[\mathbf{u}] + \epsilon}} * \gamma + \beta, \tag{2.4.5}$$

όπου η μέση τιμή και η διασπορά είναι:

$$\mathbb{E}[\mathbf{u}] = \frac{1}{m}\sum_{i=1}^{m} u_i$$

$$\text{Var}[\mathbf{u}] = \frac{1}{m}\sum_{i=1}^{m}(u_i - \mathbb{E}[\mathbf{u}])^2.$$

και $\gamma, \beta$ είναι παράμετροι προς εκμάθηση. Η σταθερά $\epsilon$ στον παρονομαστή χρησιμοποιείται για λόγους αριθμητικής ευστάθειας. Η κανονικοποίηση κατά αυτό τον τρόπο μπορεί να βοηθήσει στην επιτάχυνση και τη σταθεροποίηση της διαδικασίας εκπαίδευσης. Στη συνέχεια εφαρμόζεται το Multi-Head Attention και η έξοδός του αθροίζεται με την είσοδο στο block (πριν το layer normalizaion). Η σύνδεση αυτή είναι γνωστή ως residual connection και μπορεί να βοηθήσει στο να διατηρηθούν κάποιες αρχικές πληροφορίες ακόμα και μετά τους μετασχηματισμούς που επιφέρει ο μηχανισμός Multi-Head Attention. Ύστερα εφαρμόζουμε ξανά layer normalization και περνάμε τις εξόδους στο δίκτυο πρόσθιας τροφοδότησης (FFN). Εφαρμόζουμε residual connection αθροίζοντας την είσοδο πριν το (δεύτερο) layer normalization με την έξοδο από το feed forward δίκτυο. Συνεπώς η σχέση που περιγράφει κάθε encoder block είναι:

$$\mathbf{x} + \text{SubBlock}(\text{LayerNorm}(\mathbf{x})), \tag{2.4.6}$$

όπου το SubBlock αναφέρεται πρώτα στο Multi-Head Attention και έπειτα στο position-wise FFN. Στην έξοδο του τελευταίου encoder block εφαρμόζουμε layer normalization και λαμβάνουμε τελικά την κρυφή αναπαράσταση $\mathbf{h} = (h_1, \ldots, h_L)$ της ακολουθίας εισόδου. Η δομή του encoder φαίνεται στο αριστερό μέρος του Σχήματος 2.7.

Στη συνέχεια ο decoder αναλαμβάνει την αποκωδικοποίηση των hidden features προκειμένου να εξάγει την ακολουθία εξόδου $\mathbf{y} = (y_1, \ldots, y_S)$, όπου στην περίπτωσή μας αποτελεί τα frames του φασματογραφήματος στην κλίμακα mel. Σε κάθε βήμα ο decoder δέχεται ως είσοδο ένα frame μαζί με τα τα encoder hidden states και παράγει το επομένο frame. Ο decoder αποτελείται από $N_d$ blocks όπου η έξοδος του κάθε block τροφοδοτείται στο επόμενο. Κάθε ένα από αυτά περιέχει δύο μηχανισμούς Multi-Head Attention και ένα δίκτυο πρόσθιας τροφοδότησης. Ο πρώτος μηχανισμός ονομάζεται Masked Multi-Head Attention. Όταν ο decoder επεξεργάζεται το frame $y_t$ στο βήμα t, τότε θα πρέπει να «εστιάζει την προσοχή του» μόνο στα frames που έχουν παραχθεί μέχρι και εκείνη τη στιγμή και όχι σε εκείνα που βρίσκονται μετά από αυτό. Αυτό συμβαίνει και κατά την εκπαίδευση αλλά κυρίως κατά τη διάρκεια του inference, αφού σε αυτή την περίπτωση δεν είναι διαθέσιμα τα frames του φασματογραφήματος και πρέπει να προβλεφθούν. Χρησιμοποιώντας μία μάσκα αποτρέπουμε τον decoder να εστιάσει σε μελλοντικά frames. Μετά το Masked Multi-Head Attention ακολουθεί ένα επίπεδο κανονικοποίησης μαζί με ένα residual connection. Έπειτα η δομή του decoder block είναι ίδια με ένα encoder block. Ακολουθεί ο μηχανισμός Multi-Head Attention που δέχεται ως query την έξοδο του Masked Multi-Head Attention, ενώ για keys και values χρησιμοποιούνται τα hidden states του encoder. Ως τελευταίο module χρησιμοποιείται το δίκτυο πρόσθιας τροφοδότησης. Κατ' αυτό τον τρόπο ο decoder εκμεταλλεύεται το περιεχόμενο (context) της ακολουθίας των φωνημάτων για να παράγει βήμα βήμα τα frames του φασματογραφήματος.

Ύστερα από το τελευταίο decoder block χρησιμοποιούνται δύο fully connected επίπεδα. Το πρώτο (Mel Linear) παράγει το frame και ακολουθεί ένα δίκτυο post-net, όπως και στο μοντέλο Tacotron2 με 5 συνελικτικά στρώματα, που στοχεύει να βελτιώσει το αποτέλεσμα του frame μέσω ενός residual connection. Το άλλο δίκτυο (Stop Linear) δίνει ως έξοδο έναν αριθμό έτσι ώστε το δίκτυο να μπορεί να τερματίζει δυναμικά κατά το inference και να μην παράγει συνεχώς frames.

### 2.4.2 Εκπαίδευση και αξιολόγηση του μοντέλου

Για την εκπαίδευση του μοντέλου στο πρόβλημα της σύνθεσης φωνής από κείμενο χρησιμοποιήθηκε ένα εσωτερικό σύνολο δεδομένων (US English female dataset) που περιέχει ζεύγη από κείμενα με τα αντίστοιχα ηχητικά δείγματα (waveforms). Το σύνολο αυτό περιέχει συνολικά 25 ώρες ηχογράφησης και για την εκπαίδευση του μοντέλου χρειάστηκαν 4 GPUs. Εφόσον η αρχιτεκτονική

του μοντέλου επιτρέπει την παραλληλοποίησή του, η εκπαίδευση επιτυγχάνεται σε λιγότερο χρόνο. Συγκεκριμένα για ένα βήμα εκπαίδευσης με ίδιο πλήθος δειγμάτων (16) ανά batch, ο χρόνος είναι περίπου 4 φορές λιγότερος σε σχέση με το μοντέλο Tacotron2. Παρ' όλ' αυτά ο συνολικός χρόνος εκπαίδευσης παραμένει υψηλός (περίπου 3 μέρες) λόγου του μεγάλου πλήθους παραμέτρων στο μοντέλο. Για την αξιολόγηση χρησιμοποιήθηκαν 38 τυχαία δείγματα από το dataset. Η μετρική που υπολογίστηκε είναι το MOS (Mean Opinion Score). Από τουλάχιστον 20 άτομα βαθμολογήθηκαν οι προβλέψεις του μοντέλου Transformer, του Tacotron2 καθώς και οι πραγματικές ηχογραφήσεις και προέκυψαν τα αντίστοιχα scores. Το MOS των δύο μοντέλων ήταν ίδιο (με τιμή 4.39 ενώ οι πραγματικές ηχογραφήσεις είχαν τιμή 4.44), γι' αυτό χρησιμοποιήθηκε και η μετρική του CMOS (comparison-MOS). Σύμφωνα με τους συγγραφείς οι βαθμολογητές αξιολόγησαν τα αποτελέσματα των δύο μοντέλων, βαθμολογώντας τα στην κλίμακα $[-3, 3]$ με διαβάθμιση μιας μονάδας. Το μοντέλο Transformer-TTS υπερέχει κατά 0.048 έναντι του Tacotron2. Παρ' όλ' αυτά προτείνονται και άλλες αρχιτεκτονικές προκειμένου να βελτιώσουν τα αρνητικά του Transformer, όπως το μεγάλο πλήθος παραμέτρων και ο αρκετός χρόνος που απαιτείται κατά την εκπαίδευση και συμπερασματολογία. Εν συνεχεία αναλύουμε βασικές αρχιτεκτονικές που αφορούν τα μοντέλα vocoders.

## 2.5 WaveNet

Το WaveNet [Oor+16] είναι ένα μοντέλο που ανήκει στην κατηγορία των vocoders και χρησιμοποιείται για την παραγωγή μιας κυματομορφής ήχου. Αναπτύχθηκε από ερευνητές της Google Deep Mind προκείμενου να βελτιώσει ήδη υπάρχοντα αποτελέσματα σε προβλήματα όπως η σύνθεση φωνής από κείμενο και η παραγωγή μουσικής. Πρόκειται για ένα πιθανοτικό και αυτοπαλινδρομικό (autoregressive) δίκτυο του οποίου η αρχιτεκτονική στηρίζεται στο μοντέλο PixelCNN [Van+16]. Συγκεκριμένα, το βασικό στοιχείο της αρχιτεκτονικής του είναι η χρήση διεσταλμένων (dilated) συνελίξεων, οι οποίες αυξάνουν το δεκτικό πεδίο (receptive field) του μοντέλου και μπορούν να μοντελοποιήσουν αποδοτικά μακροπρόθεσμες χρονικές εξαρτήσεις, όπως θα δούμε παρακάτω. Από την αξιολόγησή του προκύπτει ότι τα παραγόμενα ηχητικά δείγματα υπερέχουν ως προς το πόσο φυσικά ακούγονται σε σχέση με αντίστοιχα ηχητικά δείγματα που προκύπτουν από άλλες παραμετρικές και συναθροιστικές μεθόδους. Παρακάτω παρουσιάζουμε αναλυτικά τα βασικά στοιχεία που δομούν την αρχιτεκτονική του μοντέλου καθώς και τα αντίστοιχα αποτελέσματα.

### 2.5.1 Αρχιτεκτονική

Το μοντέλο WaveNet προσπαθεί να μοντελοποιήσει την από κοινού συνάρτηση πυκνότητας πιθανότητας μιας κυματομορφής $\mathbf{x} = (x_1, \ldots, x_T)$, η οποία μπορεί να γραφεί ως

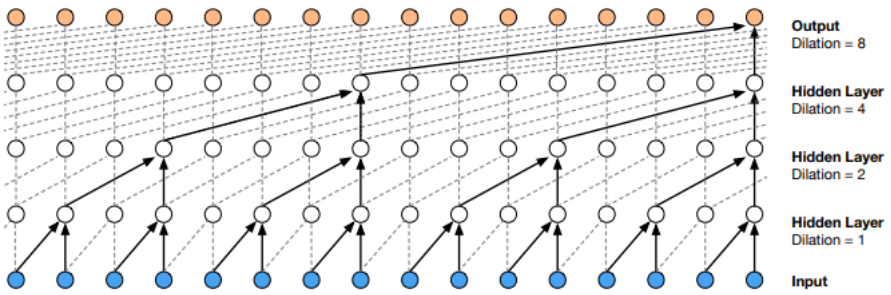
$$p(\mathbf{x}) = \prod_{t=1}^{T} p(x_t | x_1, \ldots, x_{t-1}). \tag{2.5.1}$$

Σε κάθε χρονική στιγμή το δείγμα $x_t$ εξαρτάται μόνο από τα δείγματα $x_1, \ldots, x_{t-1}$ που έχουν προηγηθεί αυτού. Κάθε τιμή $x_t$ της κυματομορφής αποθηκεύεται στη μνήμη χρησιμοποιώντας 16-bit που αναπαριστούν έναν ακέραιο, επομένως θα λαμβάνει μία από τις $2^{16} = 65536$ τιμές. Έτσι η έξοδος του δικτύου προκύπτει ύστερα από την εφαρμογή ενός επιπέδου softmax, που δίνει τις πιθανότητες για κάθε μία τιμή. Κατά την εκπαίδευση του δικτύου μεγιστοποιείται ο

λογάριθμος της πιθανοφάνειας των δεδομένων. Για να μειωθεί η διάσταση της εξόδου εφαρμόζεται ο μετασχηματισμός μ-law [Rec88] και ύστερα γίνεται quantization προκειμένου οι τιμές της εξόδου να μειωθούν στις 256. Ο μετασχηματισμός μ-law δίνεται από τη σχέση:

$$f(x_t) = \text{sign}(x_t)\frac{\ln(1+\mu|x_t|)}{\ln(1+\mu)}, \text{ όπου } \mu = 255 \text{ και } -1 < x_t < 1. \tag{2.5.2}$$

Όσον αφορά την αρχιτεκτονική, το βασικό στοιχείο που χρησιμοποιείται είναι τα dilated casual convolutions. Όπως φαίνεται και από το Σχήμα 2.9, για την έξοδο σε ένα χρονικό βήμα λαμβάνονται υπόψιν μόνο τα δείγματα που βρίσκονται πριν από αυτό, γεγονός που οφείλεται στη χρήση των casual convolutions. Κατά συνέπεια η πιθανότητα $p(x_t|x_1, \ldots, x_{t-1})$ δεν εξαρτάται από μελλοντικά χρονικά δείγματα παρά μόνο από όσα έχουν ήδη προηγηθεί. Επιπλέον για να αυξηθεί το δεκτικό πεδίο (receptive field) του μοντέλου, δηλαδή το πλήθος των δειγμάτων που λαμβάνει υπόψιν η έξοδος σε κάθε βήμα, χρησιμοποιούνται διεσταλμένες (dilated) συνελίξεις. Όπως βλέπουμε και στο Σχήμα 2.9 σε κάθε επίπεδο αυξάνεται εκθετικά ο παράγοντας διαστολής. Στο μοντέλο WaveNet, ο παράγοντας διαστολής διπλασιάζεται σε κάθε συνελικτικό επίπεδο λαμβάνοντας τις τιμές $1, 2, \ldots, 512$, ώστε στο τελευταίο επίπεδο το μοντέλο να «βλέπει» τα προηγούμενα 1024 δείγματα της εισόδου. Σημειώνεται ότι το πλήθος των σημείων της εξόδου παραμένει ίδιο με το πλήθος των σημείων της εισόδου ύστερα από την εφαρμογή κάθε συνελικτικού επιπέδου.

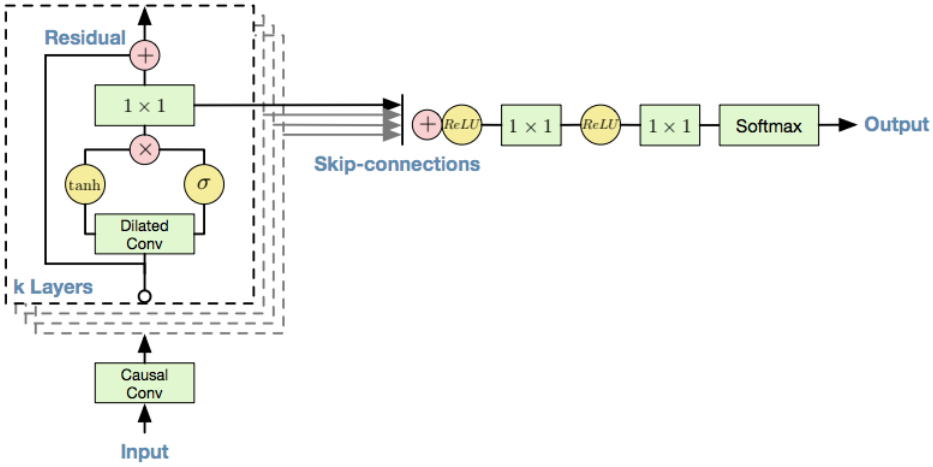



Σχήμα 2.9: Dilated casual convolutions στο μοντέλο WaveNet. [Oor+16]

Στο Σχήμα 2.10 παρουσιάζεται η αρχιτεκτονική του μοντέλου WaveNet. Παρατηρούμε ότι αποτελείται από ορισμένα residual blocks [Sze+17] των οποίων η έξοδος τροφοδοτείται στο επόμενο block. Κάθε ένα από αυτά χρησιμοποιεί αρχικά dilated casual convolutions, όπως περιγράψαμε προηγουμένως. Εν συνεχεία εφαρμόζεται μια συνάρτηση ενεργοποίησης που ονομάζεται gated activation [Van+16] και δίνεται από τον τύπο:

$$\mathbf{z} = \tanh(\mathbf{W}_{f,k} * \mathbf{x}) \odot \sigma(\mathbf{W}_{g,k} * \mathbf{x}). \tag{2.5.3}$$

Σύμφωνα με το τύπο αυτό εφαρμόζονται δύο συνελίξεις με την είσοδο $\mathbf{x}$ χρησιμοποιώντας ως φίλτρα τους πίνακες $\mathbf{W}_{f,k}$ και $\mathbf{W}_{g,k}$. Τα δύο αποτελέσματα περνούν από τις συναρτήσεις υπερβολική εφαπτομένη tanh και τη σιγμοειδή $\sigma$ και τέλος λαμβάνεται το γινόμενο τους. Ύστερα εφαρμόζεται άλλο ένα συνελικτικό επίπεδο με μέγεθος πυρήνα 1 ($1 \times 1$), η έξοδος του οποίου αθροίζεται με την αρχική είσοδο στο block και περνά στο επόμενο block (residual connection). Επιπλέον σε κάθε block η έξοδος του $1 \times 1$ συνελικτικού επιπέδου λειτουργεί ως skip connection ώστε στο τέλος όλες οι έξοδοι από τα block να αθροίζονται μεταξύ τους. Το άθροισμά τους περνά από μια σειρά τελικών μετασχηματισμών (συνάρτηση ενεργοποίησης ReLU και $1 \times 1$ convolution) και τέλος το επίπεδο softmax δίνει τις τελικές πιθανότητες.




Σχήμα 2.10: Αρχιτεκτονική του μοντέλου WaveNet. [Oor+16]

Για το πρόβλημα της σύνθεσης φωνής από κείμενο μπορούν να εξαχθούν ορισμένα χαρακτηριστικά τα οποία δίνονται και αυτά ως είσοδος στο μοντέλο. Συγκεκριμένα χρησιμοποιούνται κάποια γλωσσικά χαρακτηριστικά από το κείμενο (π.χ. χαρακτήρες, φωνήματα) καθώς και από το ηχητικό σήμα, όπως ο λογάριθμος της θεμελιώδους συχνότητας $\log F_0$. Τα χαρακτηριστικά αυτά μπορούν να αναπαρασταθούν από μια μεταβλητή $\mathbf{h}$, οπότε η συνάρτηση πυκνότητας πιθανότητας γράφεται:

$$p(\mathbf{x}|\mathbf{h}) = \prod_{t=1}^{T} p(x_t|x_1, \ldots, x_{t-1}, \mathbf{h}). \tag{2.5.4}$$

Τα χαρακτηριστικά $\mathbf{h}$ γίνονται upsampling με χρήση ενός transposed συνελικτικού επιπέδου προκειμένου να έρθουν στην ίδια διάσταση με την κυματομορφή $\mathbf{x}$. Το αποτέλεσμα αυτό εισάγεται στη συνάρτηση ενεργοποίησης, η οποία γράφεται ως

$$\mathbf{z} = \tanh(\mathbf{W}_{f,k} * \mathbf{x} + \mathbf{V}_{f,k} * \mathbf{y}) \odot \sigma(\mathbf{W}_{g,k} * \mathbf{x} + \mathbf{V}_{g,k} * \mathbf{y}), \tag{2.5.5}$$

όπου $\mathbf{y}$ είναι τα χαρακτηριστικά που προκύπτουν μετά το upsampling και $\mathbf{V}_{f,k}, \mathbf{V}_{g,k}$ δύο νέα φίλτρα συνέλιξης.

### 2.5.2 Αποτελέσματα

Για την εκπαίδευση του μοντέλου WaveNet χρησιμοποιήθηκαν δύο εσωτερικά σύνολα δεδομένων σε δύο γλώσσες αγγλικά και κινέζικα. Το πρώτο (North American English) περιείχε 24.6 ώρες ενώ το δεύτερο (Mandarin Chinese) 34.8 ώρες ηχογραφήσεων. Για τη σύγκριση με άλλες μεθόδους σύνθεσης φωνής χρησιμοποιήθηκε ένα συναθροιστικό μοντέλο τύπου HMM [Gon+16] και ένα παραμετρικό μοντέλο τύπου LSTM [Zen+16]. Η σύγκριση των παραγόμενων ηχητικών δειγμάτων έγινε με τη μετρική του MOS (Mean Opinion Score) καθώς επίσης και με συγκρίσεις ανά δύο μεταξύ των συνθετικών δειγμάτων. Συγκεκριμένα όσον αφορά τη μετρική του MOS, οι βαθμολογητές άκουγαν πρώτα κάποιο δείγμα που είτε είχε παραχθεί με κάποιο από τα μοντέλα, είτε αντιστοιχούσε σε κανονικό ηχητικό δείγμα και στη συνέχεια έδιναν ένα σκορ από το 1 έως το 5 για το πόσο φυσικά ακουγόταν. Οι υψηλότερες βαθμολογίες αντιστοιχούσαν σε καλύτερης ποιότητας δείγματα. Τα αποτελέσματα για το μοντέλο WaveNet είναι σαφώς υψηλότερα σε σχέση με τις δύο άλλες μεθόδους. Για τα κανονικά δείγματα η μετρική του MOS είχε τιμή περίπου 4.55

και 4.21 για τα Αγγλικά και τα Κινέζικα αντίστοιχα, ενώ για το WaveNet οι αντίστοιχες τιμές ήταν περίπου 4.21 και 4.08. Οι άλλες δύο μέθοδοι έδιναν τιμές κάτω από 4 και για τα δύο σύνολα δεδομένων. Η υπεροχή του μοντέλου WaveNet προκύπτει και από τα αποτελέσματα βάσει των συγκρίσεων που έγιναν μεταξύ ορισμένων δειγμάτων. Συγκεκριμένα, συγκρίνοντας ηχητικά δείγματα που προκύπτουν από το μοντέλο WaveNet και της βέλτιστης από τις άλλες δύο μεθόδους[2] προκύπτει ότι για τα αγγλικά το 49.3% επιλέγει ως καλύτερα τα δείγματα του WaveNet, το 20.1% τα δείγματα της συναθροιστικής μεθόδου και το 30.6% δεν έχει κάποια προτίμηση μεταξύ των δύο. Αντίστοιχα για τα κινέζικα το 29.3% επιλέγει ως καλύτερα τα δείγματα του WaveNet, το 12.5% τα δείγματα της παραμετρικής μεθόδου και το 58.2% δεν έχει κάποια προτίμηση μεταξύ των δύο. Λαμβάνοντας λοιπόν υπόψιν τις μετρικές αξιολόγησης στα δύο σύνολα δεδομένων, συμπεραίνουμε ότι το μοντέλο WaveNet μπορεί να παράγει ηχητικά δείγματα πολύ καλής ποιότητας σε σχέση με παραδοσιακές μεθόδους που χρησιμοποιούνταν πριν από αυτό και να χρησιμοποιηθεί εναλλακτικά ως μέρος ενός συστήματος σύνθεσης φωνής από κείμενο.

## 2.6 WaveGlow

Ως γνωστόν το πρόβλημα για τη σύνθεση φωνής από κείμενο (text-to-speech) μπορεί να διασπαστεί σε δύο επιμέρους προβλήματα. Το πρώτο αφορά τη μετατροπή του κειμένου σε ορισμένα χαρακτηριστικά, όπως είναι το φασματογράφημα στην κλίμακα mel και μπορεί να προκύψει χρησιμοποιώντας αρχιτεκτονικές όπως το Tacotron2. Στη συνέχεια αναλαμβάνει ένα μοντέλο vocoder προκειμένου να μετατρέψει τα χαρακτηριστικά αυτά στα αντίστοιχα δείγματα μιας κυματομορφής ήχου. Στην ενότητα αυτή εστιάζουμε σε ένα μοντέλο vocoder που ονομάζεται WaveGlow [PVC19]. Το WaveGlow είναι ένα γενετικό μοντέλο βασισμένο σε ροή (flow-based generative network), που αποτελείται από μία σειρά αντίστροφων μετασχηματισμών για την παραγωγή δειγμάτων μιας κυματομορφής. Χρησιμοποιώντας πληροφορία από ένα φασματογράφημα στην κλίμακα mel μπορεί να παράγει ηχητικά δείγματα πολύ καλής ποιότητας, όπως θα δούμε παρακάτω και από τα σχετικά αποτελέσματα. Η δομή του συνδυάζει στοιχεία από τα μοντέλα Glow [KD18] και WaveNet [Oor+16], αλλά σε αυτή την περίπτωση πρόκειται για ένα μη-αυτοπαλινδρομικό (non-autoregressive) μοντέλο, με την έννοια ότι για την παραγωγή ενός δείγματος σε μια δεδομένη χρονική στιγμή, δε λαμβάνονται υπόψιν μόνο τα προηγούμενα δείγματα[3]. Το βασικό θετικό στοιχείο του μοντέλου είναι ότι για την εκπαίδευσή του χρειάζεται μόνο η ελαχιστοποίηση μιας απλής συνάρτησης κόστους, δηλαδή του αρνητικού λογαρίθμου της πιθανοφάνειας των δεδομένων, που όπως θα δούμε υπολογίζεται εύκολα λόγω των αντίστροφων μετασχηματισμών που συνθέτουν το μοντέλο και με χρήση του θεωρήματος αλλαγής μεταβλητών. Αν και πρόκειται για ένα απλό δίκτυο ως προς την υλοποίηση με πολύ καλά αποτελέσματα, στα αρνητικά του συγκαταλέγεται ο μεγάλος χρόνος που απαιτείται κατά την εκπαίδευση και τη συμπερασματολογία (inference). Προτού περιγράψουμε την αρχιτεκτονική του, παραθέτουμε ορισμένες βασικές έννοιες στις οποίες στηρίζεται το μοντέλο WaveGlow.

[2] για τα αγγλικά καλύτερα αποτελέσματα δίνει η συναθροιστική μέθοδος HMM, ενώ για τα κινέζικα η παραμετρική μέθοδος LSTM.

[3] Η autoregressive φύση στο μοντέλο WaveNet οφείλεται στη χρήση των casual convolutions στην αρχιτεκτονική του, ενώ το WaveGlow δε χρησιμοποιεί casual convolutions, γεγονός που το καθιστά ένα non-autoregressive μοντέλο.

### 2.6.1 Normalizing Flow

Ένα μοντέλο ομαλοποιημένης ροής (normalizing flow) [RM15] μετασχηματίζει μία απλή κατανομή σε μία σύνθετη μέσω μιας σειράς αντίστροφων μετασχηματισμών, με στόχο τον ευκολότερο υπολογισμό της σύνθετης κατανομής των τελικών δεδομένων που προκύπτουν. Αυτό ακριβώς συμβαίνει και στο μοντέλο WaveGlow αφού για τη σύνθεση μιας κυματομορφής ήχου, χρειάζεται πρώτα να πάρουμε δείγματα από την κανονική κατανομή και στη συνέχεια μέσω ορισμένων μετασχηματισμών (steps of flow) να παράγουμε τα τελικά δείγματα ήχου. Με το μοντέλο normalizing flow και με τη χρήση του θεωρήματος αλλαγής μεταβλητών, ο υπολογισμός της πιθανοφάνειας των ηχητικών δειγμάτων ανάγεται εύκολα στον υπολογισμό της πιθανοφάνειας των δειγμάτων που προέρχονται από την κανονική κατανομή.

Αρχικά ας υποθέσουμε ότι έχουμε δύο συνεχείς τυχαίες μεταβλητές (τ.μ.) $X, Y$ με τιμές στο $\mathbb{R}$, οι οποίες συνδέονται μέσω ενός αντίστροφου μετασχηματισμού $f : \mathbb{R} \to \mathbb{R}$ σύμφωνα με τη σχέση

$$y = f(x) \iff x = f^{-1}(y), \text{ όπου } X \sim p_X(x) \text{ και } Y \sim p_Y(y). \tag{2.6.1}$$

Έστω ότι είναι γνωστή η συνάρτηση πυκνότητας πιθανότητας (σ.π.π.) $p_X(x)$ της τ.μ. $X$ και μπορούμε εύκολα να πάρουμε δείγματα από αυτήν (π.χ. μια κανονική κατανομή). Για να υπολογίσουμε τη σ.π.π. της τ.μ. $Y$ έχουμε:

$$F_Y(y) = \mathbb{P}[Y \leq y] = \mathbb{P}[f(X) \leq y] = \begin{cases} \mathbb{P}[X \leq f^{-1}(y)] = F_X(f^{-1}(y)), & \text{αν } f^{-1} \text{ αύξουσα} \\ \mathbb{P}[X \geq f^{-1}(y)] = 1 - F_X(f^{-1}(y)), & \text{αν } f^{-1} \text{ φθίνουσα,} \end{cases}$$

επομένως λαμβάνοντας την παράγωγο ως προς $y$ παίρνουμε

$$p_Y(y) = \frac{dF_Y(y)}{dy} = \begin{cases} p_X(f^{-1}(y))(f^{-1}(y))', & \text{αν } f^{-1} \text{ αύξουσα} \\ -p_X(f^{-1}(y))(f^{-1}(y))', & \text{αν } f^{-1} \text{ φθίνουσα.} \end{cases}$$

Τελικά η σ.π.π. της τ.μ. $Y$ θα είναι $p_Y(y) = p_X(f^{-1}(y))|(f^{-1}(y))'|$. Για την περίπτωση των πολυδιάστατων κατανομών προκύπτει η εξής ανάλογη σχέση

$$p_Y(\mathbf{y}) = p_X(f^{-1}(\mathbf{y}))\left|\det\left(\mathbf{J}(f^{-1}(\mathbf{y}))\right)\right|, \tag{2.6.2}$$

όπου $\det(\mathbf{J}(f^{-1}(\mathbf{y})))$ είναι η ορίζουσα του Ιακωβιανού (Jacobian) πίνακα των παραγώγων πρώτης τάξης του αντίστροφου μετασχηματισμού $f^{-1}$. Έστω τώρα ότι έχουμε μια σειρά από αντίστροφους μετασχηματισμούς, όπως φαίνεται στο Σχήμα 2.11, όπου η αρχική τ.μ. $Z_0$ συνδέεται με την τ.μ. $Z_k = X$ μέσω της σχέσης

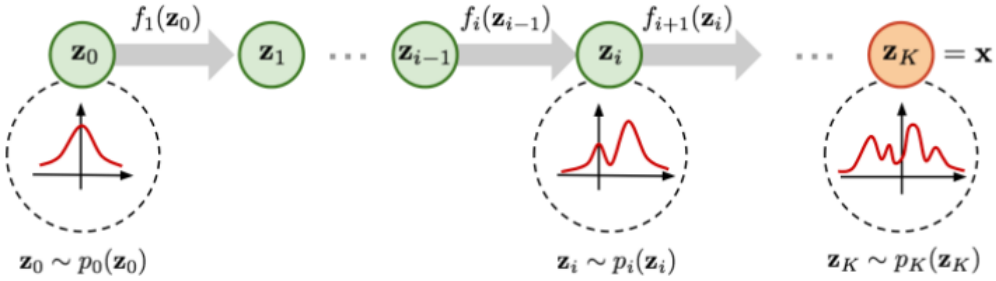
$$\mathbf{x} = \mathbf{z}_k = f_k \circ f_{k-1} \cdots \circ f_1(\mathbf{z}_0) \iff \mathbf{z}_0 = f_1^{-1} \circ f_2^{-1} \cdots \circ f_k^{-1}(\mathbf{x}). \tag{2.6.3}$$

Όμως από τη σχέση 2.6.2 έχουμε $p_i(\mathbf{z}_i) = p_{i-1}(f_i^{-1}(\mathbf{z}_i))\left|\det\left(\mathbf{J}(f_i^{-1}(\mathbf{z}_i))\right)\right|$, για κάθε $i = 1, \ldots, k$. Επιπλέον ισχύει:

$$\mathbf{J}(f_i^{-1}(\mathbf{z}_i)) = \frac{\partial f_i^{-1}(\mathbf{z}_i)}{\partial \mathbf{z}_i} = \frac{\partial \mathbf{z}_{i-1}}{\partial \mathbf{z}_i} = \left(\frac{\partial \mathbf{z}_i}{\partial \mathbf{z}_{i-1}}\right)^{-1} = \left(\frac{\partial f_i(\mathbf{z}_{i-1})}{\partial \mathbf{z}_{i-1}}\right)^{-1} = (\mathbf{J}(f_i(\mathbf{z}_{i-1})))^{-1} \tag{2.6.4}$$

και για έναν αντιστρέψιμο πίνακα $A$ είναι

$$\det\left(AA^{-1}\right) = \det(\mathbb{I}) = 1 \Rightarrow \det\left(A^{-1}\right) = (\det(A))^{-1}. \tag{2.6.5}$$




Σχήμα 2.11: Μοντέλο normalizing-flow. Τα τελικά δεδομένα $\mathbf{x}$ υπολογίζονται μέσω μιας σειράς αντίστροφων μετασχηματισμών $f_i$ από τα αρχικά δεδομένα $\mathbf{z}_0$. Έτσι ο υπολογισμός της κατανομής των δεδομένων $\mathbf{x}$ ανάγεται στην κατανομή των δεδομένων $\mathbf{z}_0$. [Wen18]

Από τις σχέσεις 2.6.4 και 2.6.5 συμπαιρένουμε ότι οι σ.π.π. δύο διαδοχικών τ.μ. σε ένα μοντέλο normalizing-flow δίνονται από τη σχέση

$$p_i(\mathbf{z}_i) = p_{i-1}(f_i^{-1}(\mathbf{z}_i))\left|\det\left((\mathbf{J}(f_i(\mathbf{z}_{i-1})))^{-1}\right)\right| = p_{i-1}(\mathbf{z}_{i-1})|\det(\mathbf{J}(f_i(\mathbf{z}_{i-1})))|^{-1},$$

με λογάριθμο

$$\log p_i(\mathbf{z}_i) = \log p_{i-1}(\mathbf{z}_{i-1}) - \log|\det(\mathbf{J}(f_i(\mathbf{z}_{i-1})))|.$$

Υπολογίζοντας αναδρομικά την κατανομή των τελικών δεδομένων $\mathbf{x}$ λαμβάνουμε

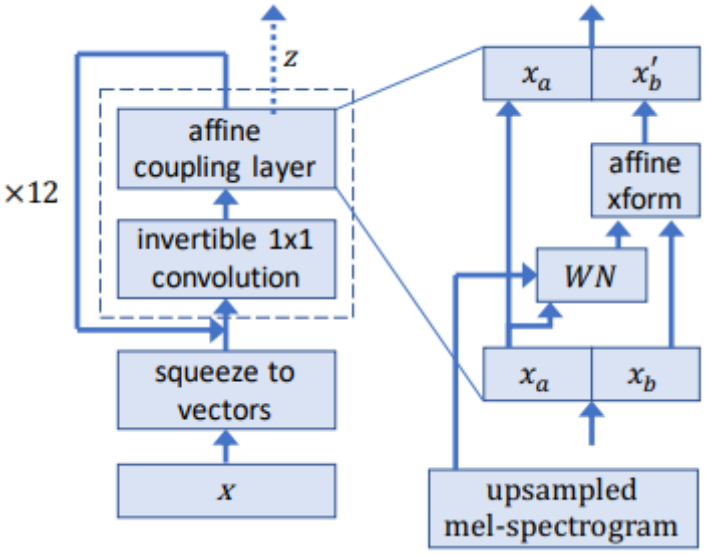
$$\log p_X(\mathbf{x}) = \log p_k(\mathbf{z}_k) = \log p_{k-1}(\mathbf{z}_{\kappa-1}) - \log|\det(\mathbf{J}(f_k(\mathbf{z}_{k-1})))|$$

$$\Rightarrow \log p_X(\mathbf{x}) = \log p_0(\mathbf{z}_0) - \sum_{i=1}^{k} \log|\det(\mathbf{J}(f_i(\mathbf{z}_{i-1})))|.$$

Για να είναι εύκολος ο υπολογισμός του λογαρίθμου της πιθανοφάνειας των τελικών δεδομένων, θα πρέπει κάθε μετασχηματισμός $f_i$ στην ομαλοποιημένη ροή να είναι εύκολα αντιστρέψιμος και επιπλέον να μπορεί να υπολογιστεί εύκολα η ορίζουσα όλων των Ιακωβιανών πινάκων. Στην περίπτωση του μοντέλου WaveGlow οι μετασχηματισμοί που χρησιμοποιούνται ικανοποιούν και τις δύο παραπάνω ιδιότητες. Στη συνέχεια περιγράφουμε την αρχιτεκτονική του μοντέλου.

### 2.6.2 Αρχιτεκτονική του μοντέλου

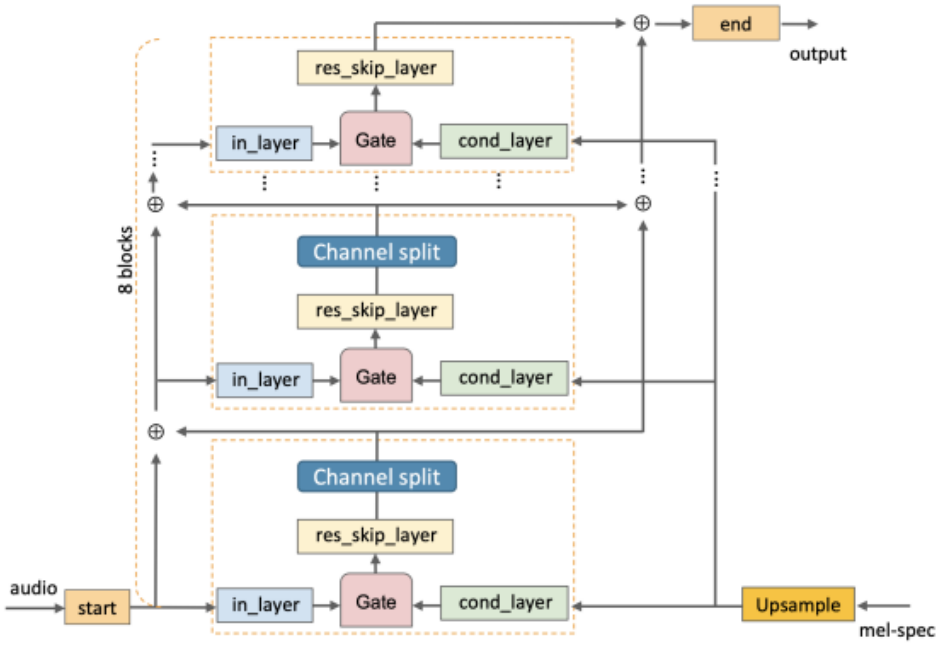
Σε πρώτη φάση για να παράγουμε μια κυματομορφή ήχου με το μοντέλο WaveGlow, λαμβάνουμε δείγματα $\mathbf{z} \sim N(\mathbf{0}, \mathbb{I})$ από την πολυδιάστατη κανονική κατανομή με μέση τιμή $\mathbf{0}$ και μοναδιαίο πίνακα συνδιασποράς. Τα δείγματα αυτά έχουν ίδια διάσταση με τη διάσταση της εξόδου, δηλαδή με το πλήθος των σημείων που αποτελούν την κυματομορφή. Εν συνεχεία ακολουθεί μια σειρά αντιστρέψιμων μετασχηματισμών που επεξεργάζονται τα δείγματα $\mathbf{z}$ καθώς και το φασματογράφημα στην κλίμακα mel και παράγουν τα τελικά δείγματα ήχου. Κατά τη φάση της εκπαίδευσης το δίκτυο δέχεται ως είσοδο μια κυματομορφή, η οποία ομαδοποιείται ανά 8 δείγματα. Για παράδειγμα αν η κυματομορφή αποτελείται από 16000 σημεία, τότε η είσοδος στο μοντέλο θα έχει διάσταση $(8, 2000)$. Η ομαδοποίηση σε groups των 8 δειγμάτων αναφέρεται ως «squeeze» από τους συγγραφείς. Στη συνέχεια ακολουθεί μια σειρά από 12 μετασχηματισμούς (steps of flow) μέχρι και την τελική έξοδο $\mathbf{z}$ του δικτύου. Η αρχιτεκτονική του μοντέλου παρουσιάζεται στο Σχήμα 2.12.




Σχήμα 2.12: Αρχιτεκτονική του μοντέλου WaveGlow. [PVC19]

Τα δύο βασικά modules που δομούν το μοντέλο WaveGlow είναι το αντιστρέψιμο $1 \times 1$ συνελικτικό επίπεδο (invertible $1 \times 1$ convolution) και το επίπεδο affine coupling [DSB16]. Το invertible $1 \times 1$ convolution είναι ένα απλό συνελικτικό επίπεδο με μέγεθος πυρήνα (kernel size) 1, όπου ο πίνακας βαρών $\mathbf{W}$ της συνέλιξης είναι αντιστρέψιμος. Ο πίνακας αυτός αρχικοποιείται κατά τέτοιο τρόπο ώστε να είναι ορθομοναδιαίος και κατά συνέπεια αντιστρέψιμος. Ένας πίνακας $\mathbf{W}$ ονομάζεται ορθομοναδιαίος όταν οι γραμμές και οι στήλες του είναι ορθομοναδιαία διανύσματα, δηλαδή έχουν μέτρο 1 και είναι ανά δύο κάθετα. Για έναν ορθομοναδιαίο πίνακα ισχύει ότι $\mathbf{W}\mathbf{W}^T = \mathbb{I} \Rightarrow \det \mathbf{W} \cdot \det \mathbf{W}^T = 1 \Rightarrow \det \mathbf{W} \neq 0$, επομένως ο πίνακας αντιστρέφεται. Το αποτέλεσμα του invertible $1 \times 1$ convolution για μια είσοδο $\mathbf{x}$ δίνεται από τη σχέση $\mathbf{y} = f_{conv}(\mathbf{x}) = \mathbf{W}\mathbf{x} \Leftrightarrow \mathbf{x} = \mathbf{W}^{-1}\mathbf{y}$. Ο Ιακωβιανός πίνακας αυτής της συνέλιξης είναι απλά ο $\mathbf{W}$, επομένως θα ισχύει $\log|\det(\mathbf{J}(f_{conv}(\mathbf{x})))| = \log|\det \mathbf{W}|$.

Στη συνέχεια εφαρμόζεται ένα επίπεδο affine coupling, όπως φαίνεται στο δεξί μέρος του Σχήματος 2.12. Η έξοδος $\mathbf{x}$ από το invertible $1 \times 1$ convolution διασπάται αρχικά σε δύο μέρη $\mathbf{x}_a$ και $\mathbf{x}_b$ με ίδιο πλήθος καναλιών το καθένα. Επιπλέον γίνεται ένα upsampling του φασματογραφήματος με χρήση ενός ανάστροφου (transpose) συνελικτικού επιπέδου, προκειμένου να έρθει στην ίδια κλίμακα με το διάνυσμα $\mathbf{x}$. Αν δηλαδή η διάσταση του $\mathbf{x}$ είναι $(8, 2000)$ τότε το upsampled φασματογράφημα θα έχει διάσταση (mel channels, 2000), όπου mel channels $= 80$. Ύστερα το μέρος $\mathbf{x}_a$ μαζί με το upsampled φασματογράφημα περνούν ως είσοδος σε ένα δίκτυο WN τύπου Wavenet, του οποίου η αρχιτεκτονική φαίνεται στο Σχήμα 2.13. Το δίκτυο WN αποτελείται από 8 blocks με συνελίξεις που χρησιμοποιούν διαστολή (dilation), η οποία διπλασιάζεται σε κάθε block. Αυξάνοντας το dilation, αυξάνεται το πεδίο λήψης (receptive field), επομένως για τον υπολογισμό ενός στοιχείου στην έξοδο, συνεισφέρουν πολλά περισσότερα δείγματα από την ακολουθία εισόδου, αντί να χρησιμοποιούσαμε απλές συνελίξεις. Πιο συγκεκριμένα, το διάνυσμα $\mathbf{x}_a$ (στο Σχήμα 2.13 ονομάζεται audio) περνά από ένα απλό convolution layer (start) ώστε να προκύψει ένα στοιχείο με 256 κανάλια. Ύστερα κάθε block περιλαμβάνει ένα συνελικτικό επίπεδο με όνομα in_layer και dilation $= 2^b$ (ανάλογα το block $b = 0, \ldots, 7$), του οποίου η έξοδος έχει 512 κανάλια. Επίσης το upsampled φασματογράφημα διέρχεται από το συνελικτικό επίπεδο με όνομα cond_layer, ώστε η έξοδος να έχει επίσης 512 κανάλια. Εν συνεχεία ο μετασχηματισμός




Σχήμα 2.13: Αρχιτεκτονική WN, τύπου Wavenet που χρησιμοποιείται στο μοντέλο WaveGlow σε ένα επίπεδο affine coupling.

Gate εφαρμόζεται στο άθροισμα των εξόδων του in_layer και του cond_layer. Στα πρώτα μισά (256) κανάλια εφαρμόζεται η υπερβολική εφαπτομένη (tanh) και στα υπόλοιπα 256 η σιγμοειδής συνάρτηση ενεργοποίησης (sigmoid) και τα αποτελέσματα πολλαπλασιάζονται μεταξύ τους. Στη συνέχεια ακολουθεί ένα συνελικτικό επίπεδο με όνομα res_skip_layer ώστε η έξοδός του να έχει 512 κανάλια (εκτός από το τελευταίο block όπου η έξοδος έχει 256 κανάλια). Έπειτα τα κανάλια διαχωρίζονται εκ νέου στα δύο (Channel split), όπου τα πρώτα 256 αθροίζονται στην είσοδο πριν το in_layer του επόμενου block και τα υπόλοιπα 256 αθροίζονται μέχρι και την έξοδο του τελευταίου res_skip_layer. Μετά το τελευταίο block ακολουθεί ένα απλό συνελικτικό επίπεδο (end) που «επαναφέρει» τα κανάλια στην αρχική διάσταση της εισόδου στο affine coupling layer. Το affine coupling layer περιγράφεται από τις παρακάτω σχέσεις:

$$
\begin{aligned}
\mathbf{x}_a, \mathbf{x}_b &= \text{split}(\mathbf{x}) \\
(\log \mathbf{s}, \mathbf{t}) &= \text{WN}(\mathbf{x}_a, \text{mel-spectrogram}) \\
\mathbf{x}'_b &= \mathbf{s} \odot \mathbf{x}_b + \mathbf{t} \\
\mathbf{x}'_a &= \mathbf{x}_a \\
f_{coupling}(\mathbf{x}) &= \text{concat}(\mathbf{x}'_a, \mathbf{x}'_b).
\end{aligned}
$$

Από την έξοδο του δικτύου WN προκύπτουν τα $(\log \mathbf{s}, \mathbf{t})$, όπου το $\log \mathbf{s}$ αντιστοιχεί στα πρώτα μισά κανάλια της εξόδου και το $\mathbf{t}$ στα υπόλοιπα. Η τελική έξοδος του affine coupling layer προκύπτει από τη συνένωση των $\mathbf{x}'_a$ και $\mathbf{x}'_b$. Το $\mathbf{x}'_a$ ισούται απλά με το $\mathbf{x}_a$, δηλαδή το πρώτο μισό ως προς τα κανάλια της εισόδου $\mathbf{x}$, ενώ το $\mathbf{x}'_b$ είναι ένας γραμμικός μετασχηματισμός του $\mathbf{x}_b$ με βάση τα $\mathbf{s}$ και $\mathbf{t}$. Εύκολα μπορούμε να δούμε ότι το επίπεδο affine coupling αποτελεί έναν αντίστροφο μετασχηματισμό. Συγκεκριμένα αν έχουμε τα $\mathbf{x}'_a$ και $\mathbf{x}'_b$ τότε η αρχική είσοδος $\mathbf{x} = \text{split}(\mathbf{x}_a, \mathbf{x}_b)$

μπορεί να ανακτηθεί από τις παρακάτω σχέσεις:

$$\begin{aligned} \mathbf{x}_a &= \mathbf{x}'_a \\ (\log \mathbf{s}, \mathbf{t}) &= \mathrm{WN}(\mathbf{x}'_a, \text{mel-spectrogram}) \\ \mathbf{x}_b &= \frac{\mathbf{x}'_b - \mathbf{t}}{\mathbf{s}}. \end{aligned}$$

Παρατηρούμε ότι για τον υπολογισμό των $(\log \mathbf{s}, \mathbf{t})$ αρκεί ένα forward pass από το δίκτυο WN αφού $\mathbf{x}_a = \mathbf{x}'_a$. Επομένως το δίκτυο WN δε χρειάζεται να είναι αντιστρέψιμο και μπορεί να έχει οποιαδήποτε σύνθετη μορφή. Επιπλέον το επίπεδο affine coupling συνεισφέρει στη συνάρτηση κόστους μέσω της σχέσης $\log|\det(\mathbf{J}(f_{coupling}(\mathbf{x})))| = \log|\mathbf{s}|$. Τελικά, ύστερα από την εφαρμογή όλων των επιπέδων invertible $1 \times 1$ convolution και affine coupling ο λογάριθμος της πιθανοφάνειας των δεδομένων θα είναι

$$\log p_\theta(\mathbf{x}) = -\frac{\mathbf{z}^T\mathbf{z}}{2\sigma^2} + \sum_{i=1}^{\#\text{conv}} \log\det|\mathbf{W}_i| + \sum_{i=1}^{\#\text{coupling}} \log|s_i|, \tag{2.6.6}$$

όπου $\theta$ είναι οι παράμετροι του μοντέλου και ο όρος $-\frac{\mathbf{z}^T\mathbf{z}}{2\sigma^2}$ είναι ο λογάριθμος της σ.π.π. της κανονικής κατανομής $N(\mathbf{0}, \sigma^2\mathbb{I})$.

Οι συγγραφείς αναφέρουν ότι κατά τη διάρκεια της εκπαίδευσης και του inference το μοντέλο έδινε ως έξοδο δύο κανάλια έπειτα από κάθε 4 affine coupling layers. Συγκεκριμένα, εφόσον το μοντέλο αποτελείται από 12 coupling layers και η είσοδος μετά το «squeeze» έχει 8 κανάλια, έπειτα από τα πρώτα 4 layers η έξοδος θα έχει 2 κανάλια, έπειτα από τα 8 layers θα έχει επίσης 2 κανάλια και έπειτα από το 12ο layer η έξοδος θα έχει 4 κανάλια (στο τελευταίο επίπεδο τα κανάλια παραμένουν τα μισά από τα αρχικά κανάλια που ήταν 8). Τέλος γίνεται μια συνένωση των δύο ενδιάμεσων εξόδων και της τελικής και προκύπτει το τελικό αποτέλεσμα $\mathbf{z}$ με 8 κανάλια. Η διαδικασία αυτή χρησιμεύει προκειμένου το μοντέλο να διατηρεί πληροφορία από την αρχική είσοδο καθώς αυτή μετασχηματίζεται από τα layers του δικτύου.

### 2.6.3 Εκπαίδευση, Αξιολόγηση και Αποτελέσματα

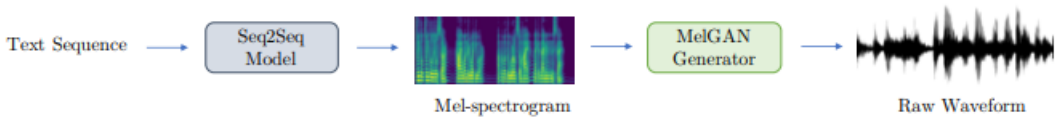
Το μοντέλο WaveGlow εκπαιδεύτηκε πάνω στο σύνολο δεδομένων LJ Speech [IJ17a], το οποίο περιλαμβάνει 13100 ηχητικά δείγματα συνολικής διάρκειας περίπου 24 ωρών μαζί με τα αντίστοιχα κείμενα (transcriptions). Από τα ηχητικά δείγματα έγινε εξαγωγή των φασματογραφημάτων στη κλίμακα mel χρησιμοποιώντας τις εξής παραμέτρους: FFT length = 1024, hop length = 256 και window length = 1024. Οι συγγραφείς αναφέρουν ότι για την εκπαίδευση του δικτύου χρειάστηκαν 8 GPUs. Το δίκτυο εκπαιδεύτηκε για 1000 εποχές με χρήση του βελτιστοποιητή Adam [KB14], μέγεθος batch ίσο με 24 και αρχικό βήμα εκμάθησης (learning rate) $10^{-4}$, το οποίο στη συνέχεια μειώθηκε στο $5 \cdot 10^{-5}$. Επίσης κατά την εκπαίδευση για τη διασπορά των μεταβλητών $\mathbf{z}$ της κανονικής κατανομής χρησιμοποιήθηκε η τιμή $\sigma^2 = 0.5$, ενώ κατά το inference η τιμή της διασποράς επιλέχθηκε ως $\sigma^2 = 0.36$.

Το μοντέλο αξιολογήθηκε βάσει της μετρικής του MOS (Mean Opinion Score). Κάθε βαθμολογητής έδινε ένα score στην κλίμακα 1 έως 5 για ορισμένες εκφωνήσεις οι οποίες δεν περιέχονταν στο σύνολο εκπαίδευσης. Το μοντέλο συγκρίθηκε και με άλλες μεθόδους όπως ο αλγόριθμος Griffin-Lim [GL84] και το μοντέλο WaveNet και παρουσίασε το υψηλότερο score με τιμή MOS $\approx 3.96$. Η τιμή του MOS για τις πραγματικές εκφωνήσεις ήταν περίπου 4.27, για

το Griffin-Lim 3.82 και για το WaveNet 3.88. Όπως φαίνεται από τα αποτελέσματα το μοντέλο WaveGlow παράγει ηχητικά δείγματα υψηλής ποιότητας. Παρ' όλο που απαιτείται μεγάλος χρόνος και υπολογιστική δύναμη για την εκπαίδευσή του, στα θετικά του στοιχεία συγκαταλέγεται η απλή συνάρτηση κόστους που ελαχιστοποιεί. Τέλος, αν και η ταχύτητά του κατά το inference είναι περίπου 25 φορές μεγαλύτερη από real-time με τιμή 520kHz σε μία GPU, εντούτοις παρουσιάζει μεγαλύτερη ταχύτητα και από τον αλγόριθμο Griffin-Lim (507kHz) και από το WaveNet (0.11kHz). Τα παραπάνω καθιστούν το WaveGlow ένα state of the art μοντέλο vocoder που μπορεί να χρησιμοποιηθεί σε ένα end-to-end σύστημα για την παραγωγή φωνής. Στη συνέχεια παρουσιάζουμε μια διαφορετική προσέγγιση που βασίζεται στη χρήση των GANs.

## 2.7 MelGAN

Το MelGAN [Kum+19] είναι ένα μοντέλο για τη σύνθεση κυματομορφής, του οποίου η δομή είναι βασισμένη στα GANs (Generative Adversarial Networks) [Goo+14]. Το μοντέλο αυτό μπορεί να χρησιμοποιηθεί εναλλακτικά στη θέση ενός vocoder σε ένα end-to-end σύστημα για την παραγωγή φωνής, όπως φαίνεται και στο Σχήμα 2.14. Σε αντίθεση με μοντέλα όπως το WaveNet [Oor+16], το MelGAN είναι ένα μη-αυτοπαλινδρομικό (non-autoregressive) και πλήρως συνελικτικό δίκτυο, του οποίου η αρχιτεκτονική αποτελείται από αρκετά λιγότερες παραμέτρους. Επίσης μπορεί να εκπαιδευτεί σε σημαντικά λιγότερο χρόνο και να παράγει ικανοποιητικά αποτελέσματα ως προς τη ποιότητα της συνθετικής φωνής. Το MelGAN, όπως όλα τα GANs αποτελείται από

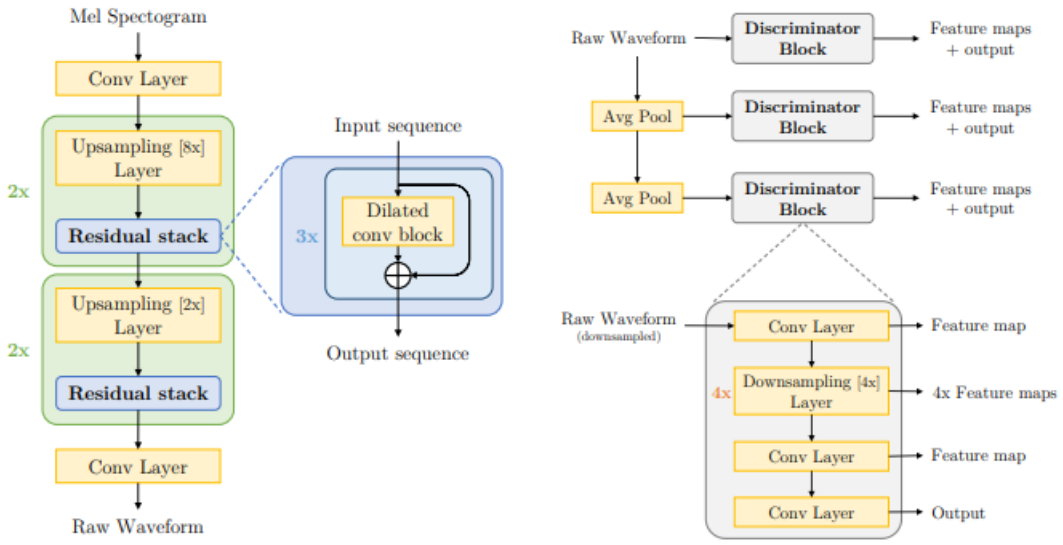



Σχήμα 2.14: End-to-end σύστημα για την παραγωγή κυματομορφής από κείμενο. Ο Generator του MelGAN μπορεί να χρησιμοποιηθεί ως ένας vocoder. [Kum+19]

δύο επιμέρους δίκτυα, τον Generator και τον Discriminator. Σκοπός του generator είναι να μετατρέψει ένα φασματογράφημα σε μία κυματομορφή που μοιάζει όσο το δυνατόν περισσότερο με πραγματική. Για παράδειγμα, στο πρόβλημα της σύνθεσης φωνής ο generator θα πρέπει να παράγει κυματομορφές που μοιάζουν σε μεγάλο βαθμό και είναι δύσκολο να ξεχωρίσουν από κυματομορφές που αντιστοιχούν σε ανθρώπινη ομιλία. Αντιθέτως ο discriminator προσπαθεί να διακρίνει αν μια κυματομορφή έχει παραχθεί από τον generator (άρα είναι ένα fake sample) ή αν αντιστοιχεί σε πραγματικό δείγμα. Κατά τη σύγκλιση θα πρέπει ο discriminator να μην μπορεί να διακρίνει αν τα δείγματα που παράγει ο generator είναι αληθινά ή όχι. Κατά συνέπεια ο generator μπορεί να χρησιμοποιηθεί ως ένας vocoder για την παραγωγή συνθετικών κυματομορφών. Στη συνέχεια παρουσιάζουμε την αρχιτεκτονική του μοντέλου MelGAN, η οποία φαίνεται στο Σχήμα 2.15.

### 2.7.1 Generator

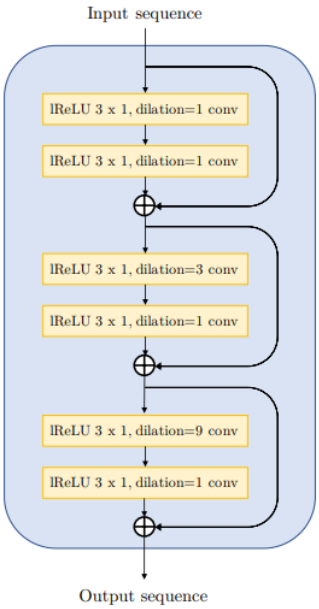
Όπως αναφέρθηκε ο generator δέχεται ως είσοδο ένα φασματογράφημα (mel-spectrogram) και παράγει μία συνθετική κυματομορφή. Σε αντίθεση με τα παραδοσιακά GANs η είσοδος τυχαίου θορύβου στον generator δε δίνει ουσιαστική βελτίωση στα αποτελέσματα και γι' αυτό χρησιμοποιείται ως είσοδος μόνο το φασματογράφημα. Ως γνωστόν η κλίμακα ενός φασματογραφήματος είναι αρκετά μικρότερη από την κλίμακα της αντίστοιχης κυματομορφής. Για παράδειγμα αν χρησιμοποιήσουμε hop length ίσο με 256 στο STFT [GL84] για την παραγωγή του φασματογραφήματος,




Σχήμα 2.15: Αρχιτεκτονική του MelGAN [Kum+19]. (a) Ο Generator παράγει μια συνθετική κυματομορφή από ένα φασματογράφημα, χρησιμοποιώντας upsampling layers και residual stacks αυξάνοντας σταδιακά την κλίμακα του φασματογραφήματος. (b) Ο Discriminator αποτελείται από 3 blocks όπου το καθένα λειτουργεί σε διαφορετική κλίμακα της κυματομορφής.

τότε το πλήθος των frames θα είναι κατά 256 φορές μικρότερο από το πλήθος των δειγμάτων-timesteps της κυματομορφής. Για το λόγο αυτό ο generator επιλέγεται ως ένα συνελικτικό δίκτυο που αποτελείται κυρίως από transposed convolutions [Zei+10] που φέρνουν σταδιακά την κλίμακα του φασματογραφήματος στην κλίμακα της κυματομορφής. Πιο συγκεκριμένα, όπως φαίνεται και στο Σχήμα 2.15 (αριστερά), το φασματογράφημα εισάγεται πρώτα σε ένα απλό συνελικτικό επίπεδο και έπειτα ακολουθούν δύο upsampling blocks που επαναλαμβάνονται δύο φορές το καθένα. Κάθε τέτοιο block χρησιμοποιεί ένα transposed convolutional επίπεδο που ακολουθείται από ένα Residual stack (βλ. Σχήμα 2.16). Η κλίμακα του mel-spectrogram αυξάνεται σταδιακά 4 φορές σύμφωνα με τους παράγοντες $[8, 8, 2, 2]$, δηλαδή το πρώτο transposed συνελικτικό επίπεδο αυξάνει την κλίμακα 8 φορές, το δεύτερο 8 φορές κ.ο.κ. Κάθε residual stack αποτελείται από τρία συνελικτικά blocks με διαστολή (dilation) ίση με 1, 3 και 9 αντίστοιχα. Όπως και στην αρχιτεκτονική του Wavenet, αυξάνοντας εκθετικά το dilation επιτυγχάνεται και αύξηση του receptive field, επομένως για την παραγωγή μιας τιμής στην ακολουθία εξόδου λαμβάνονται υπόψιν πολλά περισσότερα timesteps από την ακολουθία εισόδου, απ' ότι αν χρησιμοποιούσαμε απλές συνελίξεις. Κατά συνέπεια με τη χρήση του dilation μπορεί να μοντελοποιηθεί αποτελεσματικότερα η χρονική εξάρτηση μακρινών timesteps. Επιπλέον μεταξύ των συνελικτικών επιπέδων χρησιμοποιούνται και skip connections για να διατηρείται η πληροφορία από την ακολουθία εισόδου. Ύστερα από το δεύτερο upsampling block ακολουθεί ένα ακόμα συνελικτικό στρώμα που δίνει ως έξοδο τη συνθετική κυματομορφή. Σε όλα τα επίπεδα του generator χρησιμοποιείται ως συνάρτηση ενεργοποίησης η $\text{LeakyReLU}(x) = \max(0, x) + 0.2 \min(0, x)$, εκτός από το τελευταίο που χρησιμοποιείται η υπερβολική εφαπτομένη Tanh.

Οι συγγραφείς αναφέρουν ότι έγινε προσεκτική επιλογή των παραμέτρων στα transposed convolutional επίπεδα, διότι διαφορετικά τα αποτελέσματα δεν ήταν ικανοποιητικά. Συγκεκριμένα το



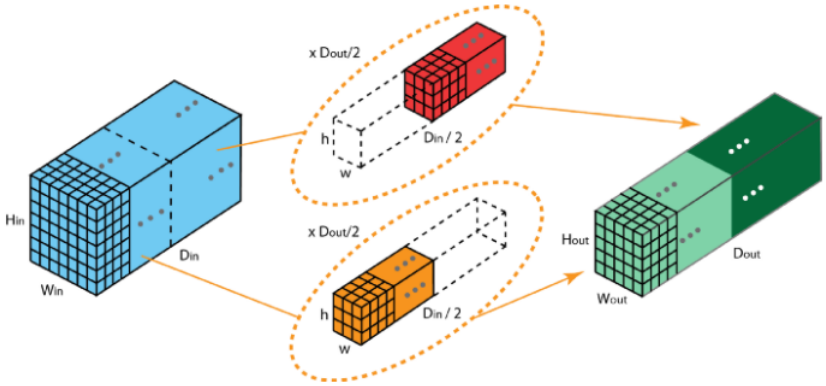

Σχήμα 2.16: Residual stack. Το dilation αυξάνεται σταδιακά από 1,3 σε 9 μοντελοποιώντας έτσι μακρινές χρονικές εξαρτήσεις στην ακολουθία εισόδου. [Kum+19]

μέγεθος του πυρήνα (kernel size) επιλέχθηκε ως πολλαπλάσιο του βήματος (stride) και επιπλέον το dilation να αυξάνεται ως δύναμη του kernel size ($3^0 = 1, 3^1 = 3, 3^2 = 9$). Ως τεχνική κανονικοποίησης χρησιμοποιήθηκε το weight normalization [SK16]. Η μέθοδος αυτή κάνει μια αναπαραμετροποίηση στα βάρη του δικτύου σύμφωνα με τη σχέση $\mathbf{w} = g\frac{\mathbf{v}}{\|\mathbf{v}\|}$, όπου $g = \|\mathbf{w}\|$ είναι το μέγεθος (κλίμακα) του διανύσματος βαρών και $\mathbf{v}$ η κατεύθυνση. Αναφέρεται ότι η τεχνική αυτή χρησιμοποιήθηκε σε όλα τα επίπεδα του generator και έδωσε τα καλύτερα αποτελέσματα σε σχέση με άλλες μεθόδους κανονικοποίησης.

### 2.7.2 Discriminator

Ο discriminator είναι μία multi-scale αρχιτεκτονική που αποτελείται από τρία discriminator blocks $D_1, D_2, D_3$ και όλα έχουν την ίδια δομή που φαίνεται στο Σχήμα 2.15 (κάτω δεξιά). Κάθε discriminator block δέχεται ως είσοδο μια κυματομορφή και παράγει ενδιάμεσες εξόδους από κάθε layer που ονομάζονται feature maps καθώς και την τελική έξοδο (output) του block. Το πρώτο block $D_1$ δέχεται ως είσοδο την κυματομορφή στην αρχική κλίμακα, ενώ τα blocks $D_2$ και $D_3$ λαμβάνουν ως είσοδο την κυματομορφή μειωμένης κλίμακας (downsampled) κατά 2 και 4 φορές αντίστοιχα. Το downsampling επιτυγχάνεται με χρήση του Average Pooling με μέγεθος πυρήνα ίσο με 4 και stride ίσο με 2. Η λογική για τη χρήση τριών blocks που επιδρούν σε διαφορετικές κλίμακες της κυματομορφής είναι ότι κάθε block στοχεύει στο να μάθει χαρακτηριστικά για διαφορετικό εύρος συχνοτήτων. Όσον αφορά την αρχιτεκτονική, κάθε discriminator block έχει στην αρχή ένα απλό συνελικτικό επίπεδο και ακολουθούν 4 συνελικτικά επίπεδα που κάνουν downsampling χρησιμοποιώντας grouped convolutions [KSH12] με μέγεθος πυρήνα ίσο με 41 και stride ίσο με 4. Ένα βασικό πλεονέκτημα για τη χρήση grouped convolutions είναι ότι μειώνεται ο αριθμός των παραμέτρων σε ένα συνελικτικό επίπεδο, αντί να χρησιμοποιούσαμε απλές συνελίξεις. Συγκεκριμένα, όπως φαίνεται και από το Σχήμα 2.17, για την περίπτωση ενός τανυστή διαστάσεων $H_{in} \times W_{in}$ με $D_{in}$ κανάλια εισόδου και $D_{out}$ κανάλια εξόδου, αν χρησιμοποιήσουμε δύο groups

φίλτρων, τότε τα πρώτα $D_{out}/2$ φίλτρα με κίτρινο χρώμα θα επιδράσουν στα $D_{in}/2$ πρώτα μισά κανάλια και τα υπόλοιπα $D_{out}/2$ φίλτρα με κόκκινο χρώμα θα επιδράσουν στα $D_{in}/2$ δεύτερα μισά κανάλια του τανυστή εισόδου. Έτσι το πλήθος των παραμέτρων για κάθε φίλτρο θα είναι $h \cdot w \cdot \frac{D_{in}}{2}$ άρα συνολικά θα έχουμε $D_{out} \cdot h \cdot w \cdot \frac{D_{in}}{2}$ παραμέτρους (χωρίς τα biases). Αντιθέτως με τη χρήση απλών συνελίξεων οι παράμετροι προς εκμάθηση θα ήταν $D_{out} \cdot h \cdot w \cdot D_{in}$, δηλαδή θα είχαμε διπλάσιο αριθμό παραμέτρων. Η τεχνική αυτή επιτρέπει τη χρήση φίλτρων με μεγάλα μεγέθη σε κάθε discriminator block χωρίς να αυξάνεται υπερβολικά το πλήθος των παραμέτρων. Για την




Σχήμα 2.17: Grouped Convolutions. Χρησιμοποιώντας δύο groups φίλτρων όπου τα πρώτα $D_{out}/2$ (κίτρινα) επιδρούν στο πρώτα μισά κανάλια $D_{in}/2$ και τα υπόλοιπα $D_{out}/2$ (κόκκινα) στα δεύτερα μισά κανάλια $D_{in}/2$ του τανυστή εισόδου, τότε το πλήθος των παραμέτρων μειώνεται στο μισό, αντί να χρησιμοποιούσαμε απλές συνελίξεις.

τελική έξοδο του block ακολουθούν δύο επιπλέον συνελικτικά επίπεδα. Σε όλα τα επίπεδα του block (εκτός από το τελευταίο) χρησιμοποιείται η συνάρτηση ενεργοποίησης LeakyReLU καθώς και weight normalization.

### 2.7.3 Εκπαίδευση

Κατά την εκπαίδευση του MelGAN ο generator και ο discriminator χρησιμοποιούν διαφορετικές συναρτήσεις κόστους. Συγκεκριμένα για τον discriminator χρησιμοποιείται μια εκδοχή του hinge loss βασισμένη σε GANs [LY17], που δίνεται από τη σχέση:

$$\min_{D_k} \left( \mathbb{E}_x \Big[ \min(0, 1 - D_k(x)) \Big] + \mathbb{E}_s \Big[ \min(0, 1 + D_k(G(s))) \Big] \right), \forall k = 1, 2, 3. \tag{2.7.1}$$

Όπως έχουμε αναφέρει κάθε discriminator block στοχεύει στη διάκριση μεταξύ των πραγματικών κυματομορφών x και των συνθετικών κυματομορφών $G(s)$ που παράγει ο generator. Αντιθέτως ο generator στοχεύει στην παραγωγή δειγμάτων που μοιάζουν με αληθινά προκειμένου ο discriminator να μην μπορεί να ξεχωρίσει μεταξύ των πραγματικών και των συνθετικών δειγμάτων. Σε αυτή την περίπτωση ο generator προσπαθεί να ελαχιστοποιήσει την ποσότητα $-D_k(G(s))$ για κάθε discriminator block. Επιπλέον χρησιμοποιείται και μια τεχνική ελαχιστοποίησης που ονομάζεται Feature Matching [Lar+16] και δίνεται από τη σχέση

$$L_{FM}(G, D_k) = \mathbb{E}_{x,s \sim p_{data}} \left[ \sum_{i=1}^{T} \frac{1}{N_i} \| D_k^{(i)}(x) - D_k^{(i)}(G(s)) \|_1 \right], \tag{2.7.2}$$

όπου $D_k^{(i)}$ είναι η έξοδος (feature map) του $i$-οστού επιπέδου στο $k$ discriminator block με $T$ επίπεδα, όπου καθένα έχει $N_i$ μονάδες. Σύμφωνα με τη μετρική αυτή, ο generator προσπαθεί να ελαχιστοποιήσει την απόσταση ($L_1-$νόρμα) των εξόδων του discriminator για μια πραγματική κυματομορφή x και μια συνθετική κυματομορφή $G(s)$. Συνδυάζοντας λοιπόν τις δύο συναρτήσεις κόστους, η τελική αντικειμενική συνάρτηση του generator δίνεται από τη σχέση

$$\min_G \left( \mathbb{E}_s \left[ \sum_{k=1}^{3} -D_k(G(s)) \right] + \lambda \sum_{k=1}^{3} L_{FM}(G, D_k) \right), \tag{2.7.3}$$

όπου το $\lambda$ λαμβάνει την τιμή 10.

Το μοντέλο εκπαιδεύτηκε στο σύνολο δεδομένων LJ Speech [IJ17b] με χρήση μόνο μίας GPU. Το σύνολο αυτό περιέχει 13100 ηχογραφήσεις μικρής διάρκειας από έναν ομιλητή στην αγγλική γλώσσα (συνολικά περίπου 24 ώρες), μαζί με τα αντίστοιχα κείμενα (transcriptions). Για την εκμάθηση των παραμέτρων του δικτύου χρησιμοποιήθηκε ο βελτιστοποιητής Adam [KB14] με ρυθμό εκμάθησης (learning rate) $10^{-4}$ και $\beta_1 = 0.5, \beta_2 = 0.9$, για τον generator και τον discriminator. Επίσης το μέγεθος των batches ήταν 16 κατά την εκπαίδευση. Τα σημαντικότερα αποτελέσματα που προέκυψαν αναφέρονται παρακάτω.

### 2.7.4 Αποτελέσματα-Αξιολόγηση

Όπως έχουμε αναφέρει το MelGAN μπορεί να χρησιμοποιηθεί εναλλακτικά ως ένας vocoder σε ένα end-to-end σύστημα για την παραγωγή φωνής. Ο λόγος γι' αυτό είναι ότι έχει ορισμένα πλεονεκτήματα έναντι άλλων μοντέλων. Συγκεκριμένα, οι συγγραφείς αναφέρουν ότι το πλήθος των παραμέτρων του MelGAN είναι 4.26 εκ. σε αντίθεση με άλλες state of the art αρχιτεκτονικές, όπως το Wavenet με 24.7 εκ. και το WaveGlow με 87.9 εκ. παραμέτρους. Το μικρό πλήθος παραμέτρων σε συνδυασμό με την non-autoregressive φύση και τη συνελικτική δομή του MelGAN, το καθιστούν αυτόματα ταχύτερο κατά τη διάρκεια του inference (περίπου 10 φορές ταχύτερο σε GPU και 25 φορές ταχύτερο σε CPU από το ταχύτερο διαθέσιμο μοντέλο).

Η μετρική για την αξιολόγηση των συνθετικών κυματομορφών είναι το MOS (Mean Opinion Score) και επιπλέον υπολογίζεται ένα 95% διάστημα εμπιστοσύνης γι' αυτή την ποσότητα. Συνολικά 200 άτομα βαθμολόγησαν στην κλίμακα 1 έως 5 ορισμένα ηχητικά κλιπ που προέκυψαν από τον generator αλλά και από άλλα μοντέλα καθώς και τις αντίστοιχες πραγματικές ηχογραφήσεις. Συγκεκριμένα για τη σύγκριση με state of the art vocoders, προέκυψαν υποδεέστερα αποτελέσματα για το MelGAN με MOS $\approx 3.61$ έναντι του WaveGlow ($\approx 4.11$) και του Wavenet ($\approx 4.05$). Επιπλέον σε ένα end-to-end σύστημα για τη σύνθεση φωνής από κείμενο, το MelGAN σε συνδυασμό με το μοντέλο Text2Mel [TUA18] έδωσε MOS $\approx 3.72$, το οποίο είναι συγκρίσιμο με συνδυασμούς όπως Tacotron2 με WaveGlow (MOS $\approx 3.52$) και Text2mel με WaveGlow (MOS $\approx 4.10$). Λαμβάνοντας υπόψιν τα παραπάνω αποτελέσματα, «θυσιάζοντας» λίγο από την ποιότητα των παραγόμενων ηχογραφήσεων αλλά κερδίζοντας σημαντικά ως προς την ταχύτητα εκπαίδευσης και συμπερασματολογίας (inference) αλλά και από τον σημαντικά μικρότερο αριθμό παραμέτρων, το MelGAN μπορεί να αποτελέσει μια εναλλακτική προσέγγιση στο πρόβλημα της σύνθεσης φωνής σε ένα end-to-end σύστημα.

## 2.8 WaveGrad 2: Επαναληπτική βελτίωση για σύνθεση φωνής από κείμενο

Τα περισσότερα συστήματα παραγωγής ομιλίας από κείμενο που έχουμε εξετάσει μέχρι στιγμής αποτελούνται από δύο επιμέρους τμήματα. Αρχικά ένα δίκτυο που δέχεται ως είσοδο το κείμενο (ακολουθία χαρακτήρων ή φωνημάτων) εξάγει ορισμένα ενδιάμεσα χαρακτηριστικά, όπως το φασματογράφημα στην κλίμακα mel και στη συνέχεια ένας vocoder παράγει την κυματομορφή ήχου χρησιμοποιώντας τα ενδιάμεσα αυτά χαρακτηριστικά. Τις περισσότερες φορές επιλέγουμε το φασματογράφημα στην κλίμακα mel ως ενδιάμεσο χαρακτηριστικό αλλά ενδεχομένως υπάρχουν και άλλες επιλογές που δίνουν καλά αποτελέσματα όσον αφορά τον τελικό παραγόμενο ήχο. Το WaveGrad 2 [Che+21] είναι ένα μη-αυτοπαλινδρομικό (non-autoregressive) μοντέλο το οποίο παράγει απευθείας την κυματομορφή ήχου από ένα κείμενο, χωρίς να χρειάζεται η εξαγωγή ενδιάμεσων χαρακτηριστικών. Η αρχιτεκτονική του αποτελείται από έναν encoder όπως στο μοντέλο Tacotron2 και σε συνδυασμό με τον decoder από το μοντέλο WaveGrad [Che+20] καθώς και ένα δίκτυο που προβλέπει τη διάρκεια κάθε στοιχείου της ακολουθίας φωνημάτων (duration predictor), παράγουν ηχητικό σήμα υψηλής ποιότητας. Το μοντέλο WaveGrad 2 στηρίζεται σε ιδέες όπως το score matching [HD05] και τα diffusion probabilistic models [HJA20]. Κατά την εκπαίδευσή του εκτιμάται το gradient $\nabla_y \log p(y|x)$ του λογαρίθμου της δεσμευμένης συνάρτησης πυκνότητας πιθανότητας της κυματομορφής $y$, όπου το $x$ αντιπροσωπεύει τα χαρακτηριστικά στα οποία δεσμεύεται η σ.π.π.[4] Το μοντέλο παράγει δείγματα από την κυματομορφή ήχου ξεκινώντας από Γκαουσιανό «θόρυβο» εκτελώντας ορισμένα βήματα (refinement steps) μέσω μιας διαδικασίας που ονομάζεται denoising. Η ποιότητα των παραγόμενων δειγμάτων και η ταχύτητα παραγωγής τους δημιουργούν ένα trade-off που εξαρτάται από το πλήθος των βημάτων refinement. Ο μηχανισμός σύνθεσης των δειγμάτων ήχου στηρίζεται σε ιδέες από τα Diffusion probabilistic μοντέλα, τα οποία παρουσιάζουμε στη συνέχεια.

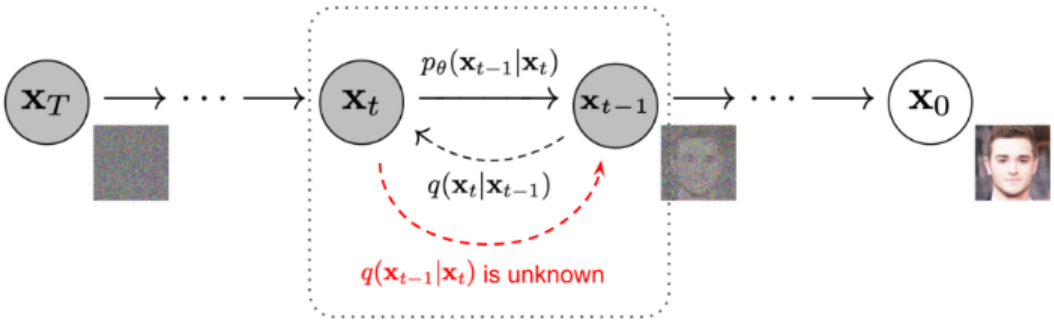
### 2.8.1 Diffusion Probabilistic Models

Τα diffusion μοντέλα τα οποία είναι εμπνευσμένα από τη θερμοδυναμική χρησιμοποιούνται για να παράγουμε δείγματα από μια επιθυμητή κατανομή, όπως για παράδειγμα την κατανομή των δεδομένων μιας κυματομορφής ήχου ή μιας εικόνας. Ξεκινώντας από την κατανομή των δεδομένων μας εκτελούμε ένα πεπερασμένο αριθμό βημάτων που ορίζονται μέσω μιας Μαρκοβιανής αλυσίδας, όπου σε κάθε βήμα προσθέτουμε Γκαουσιανό θόρυβο μέχρις ότου καταλήξουμε σε δείγματα τα οποία ακολουθούν την κανονική κατανομή. Έπειτα εκτελώντας την αντίστροφη διαδικασία, ξεκινώντας δηλαδή από δείγματα που προέρχονται από την κανονική κατανομή το μοντέλο «μαθαίνει» την κατανομή των δεδομένων. Κατ' αυτό τον τρόπο είμαστε σε θέση να παράγουμε δείγματα από μια κυματομορφή ήχου ή μια εικόνα ξεκινώντας από «θόρυβο». Η διαδικασία αυτή φαίνεται στο Σχήμα 2.18 για την περίπτωση μιας εικόνας.

#### Forward diffusion

Έστω ένα δείγμα $\mathbf{x}_0 \sim q(\mathbf{x})$, όπου $q$ είναι η συνάρτηση πυκνότητας πιθανότητας των πραγματικών δεδομένων. Κατά τη forward διαδικασία προσθέτουμε σταδιακά θόρυβο που ορίζεται βάσει μιας ακολουθίας παραμέτρων διασποράς $\{\beta_t \in (0,1)\}_{t=1}^{T}$ σύμφωνα με τη σχέση:

[4] Για παράδειγμα ως $x$ μπορεί να χρησιμοποιηθεί το mel-spectrogram που έχει προκύψει με STFT από την κυματομορφή $y$.




Σχήμα 2.18: Diffusion process. Κατά την forward διαδικασία ($\mathbf{x}_0 \to \mathbf{x}_T$), ξεκινάμε από την πραγματική εικόνα $\mathbf{x}_0$ και ύστερα από T βήματα καταλήγουμε στο «θόρυβο» $\mathbf{x}_T$. Εκτελώντας την αντίστροφη διαδικασία ($\mathbf{x}_T \to \mathbf{x}_0$), ξεκινώντας δηλαδή από θόρυβο το μοντέλο σε κάθε βήμα «μαθαίνει» την κατανομή των πραγματικών δεδομένων της εικόνας. [HJA20]

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = N(\sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbb{I}), \qquad q(\mathbf{x}_{1:T}|\mathbf{x}_0) = \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}), \tag{2.8.1}$$

όπου η δεύτερη ισότητα οφείλεται στη Μαρκοβιανή ιδιότητα:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}, \dots, \mathbf{x}_0) = q(\mathbf{x}_t|\mathbf{x}_{t-1}), \quad t = 1, \dots, T.$$

Έτσι δημιουργείται η ακολουθία δειγμάτων $\mathbf{x}_1, \dots, \mathbf{x}_T$ καθένα από τα οποία έχει ίδια διάσταση με το πραγματικό δείγμα $\mathbf{x}_0$. Αν θεωρήσουμε $\alpha_t = 1 - \beta_t$ και $\bar{\alpha}_t = \prod_{i=1}^{t}(1-\beta_i)$, τότε από τη Σχέση 2.8.1 μπορούμε να υπολογίσουμε την κατανομή ενός δείγματος $\mathbf{x}_t$ δοθέντος του αρχικού δείγματος $\mathbf{x}_0$ ως εξής:

$$\begin{aligned}
\mathbf{x}_t &= \sqrt{\alpha_t}\mathbf{x}_{t-1} + \sqrt{1-\alpha_t}\mathbf{z}_{t-1} \\
&= \sqrt{\alpha_t}(\underbrace{\sqrt{\alpha_{t-1}}\mathbf{x}_{t-2} + \sqrt{1-\alpha_{t-1}}\mathbf{z}_{t-2}}_{\mathbf{x}_{t-1}}) + \sqrt{1-\alpha_t}\mathbf{z}_{t-1} \\
&= \sqrt{\alpha_t\alpha_{t-1}}\mathbf{x}_{t-2} + \sqrt{\alpha_t(1-\alpha_{t-1})}\mathbf{z}_{t-2} + \sqrt{1-\alpha_t}\mathbf{z}_{t-1} \\
&= \sqrt{\alpha_t\alpha_{t-1}}\mathbf{x}_{t-2} + \sqrt{1-\alpha_t\alpha_{t-1}}\tilde{\mathbf{z}}_{t-2} \\
&= \dots \\
&= \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\mathbf{z}.
\end{aligned} \tag{2.8.2}$$

Στις παραπάνω ισότητες χρησιμοποιούμε ότι μια τυχαία μεταβλητή που ακολουθεί κανονική κατανομή $\mathbf{X} \sim N(\boldsymbol{\mu}, \sigma^2\mathbb{I})$ μπορεί να γραφεί ως $\mathbf{X} = \boldsymbol{\mu} + \sigma\mathbf{Z}$, όπου $\mathbf{Z} \sim N(\mathbf{0}, \mathbb{I})$. Επίσης η τυχαία μεταβλητή $\tilde{\mathbf{z}}_{t-2}$ ακολουθεί κανονική κατανομή ως άθροισμα των κανονικών τ.μ. $\mathbf{z}_{t-2}$ και $\mathbf{z}_{t-1}$. Καταλήγουμε λοιπόν ότι η σχέση που συνδέει τα $\mathbf{x}_t$ και $\mathbf{x}_0$ είναι η $\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\mathbf{z}$ και συνεπώς θα ισχύει:

$$q(\mathbf{x}_t|\mathbf{x}_0) = N(\sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbb{I}). \tag{2.8.3}$$

## Reverse diffusion

Στην πραγματικότητα προκειμένου να παράγουμε ένα δείγμα από την κατανομή των δεδομένων θα πρέπει να εκτελέσουμε την αντίστροφη διαδικασία. Ξεκινώντας δηλαδή από την τυχαία μεταβλητή $\mathbf{x}_T \sim N(\mathbf{0}, \mathbb{I})$ που αποτελεί «θόρυβο», σε κάθε βήμα λαμβάνουμε ένα δείγμα από τη δεσμευμένη κατανομή $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ ώστε τελικά να ανακτήσουμε το δείγμα $\mathbf{x}_0$ της πραγματικής κατανομής. Όμως όπως φαίνεται και στο Σχήμα 2.18 η κατανομή $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ είναι άγνωστη, επομένως θα πρέπει να την προσεγγίσουμε με μια κατανομή $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$. Η κατανομή αυτή επιλέγεται να είναι κανονική με μέση τιμή και διασπορά που εξαρτώνται από τις παραμέτρους $\theta$, δηλαδή:

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = N(\boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t)), \qquad p_\theta(\mathbf{x}_{0:T}) = p(\mathbf{x}_T)\prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t). \tag{2.8.4}$$

Επιπλέον με χρήση του κανόνα Bayes και ύστερα από πράξεις προκύπτει ότι η τ.μ. $\mathbf{x}_{t-1}$ όταν γνωρίζουμε την τ.μ. $\mathbf{x}_t$ και το αρχικό δείγμα $\mathbf{x}_0$, ακολουθεί και αυτή την κανονική κατανομή:

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = N(\tilde{\boldsymbol{\mu}}(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t\mathbb{I}), \tag{2.8.5}$$

$$\text{όπου} \quad \tilde{\boldsymbol{\mu}}(\mathbf{x}_t, \mathbf{x}_0) = \frac{\sqrt{\alpha}_t(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathbf{x}_0,$$

$$\tilde{\boldsymbol{\mu}}(\mathbf{x}_t, \mathbf{x}_0) \overset{2.8.2}{=} \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\mathbf{z}_t\right) \tag{2.8.6}$$

$$\text{και} \quad \tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t.$$

Για να βρούμε λοιπόν την κατανομή $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$ θα πρέπει να εκτιμήσουμε τις παραμέτρους $\theta$ που προσδιορίζουν τη μέση τιμή και τη διασπορά της. Αυτό επιτυγχάνεται μεγιστοποιώντας το λογάριθμο $\log p_\theta(\mathbf{x}_0)$ της πιθανοφάνειας των δεδομένων. Εναλλακτικά μπορούμε να ελαχιστοποιήσουμε το κάτω φράγμα διασποράς (variational lower bound) [KW13]:

$$L = \mathbb{E}_q\left[\log\frac{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}{p_\theta(\mathbf{x}_{0:T})}\right], \tag{2.8.7}$$

αφού αν ξεκινήσουμε από την απόκλιση (divergence) Kullback–Leibler [HO07] που εκφράζει το πόσο καλά προσεγγίζουμε την πραγματική κατανομή $q(\mathbf{x}_{1:T}|\mathbf{x}_0)$ από την κατανομή $p_\theta(\mathbf{x}_{1:T}|\mathbf{x}_0)$, τότε θα έχουμε:

$$\begin{aligned} D_{KL}(q(\mathbf{x}_{1:T}|\mathbf{x}_0) \parallel p_\theta(\mathbf{x}_{1:T}|\mathbf{x}_0)) &= \mathbb{E}_q\left[\log\frac{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}{p_\theta(\mathbf{x}_{1:T}|\mathbf{x}_0)}\right] \\ &= \mathbb{E}_q\left[\log\frac{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}{p_\theta(\mathbf{x}_{0:T})}\right] + \mathbb{E}_q\left[\log p_\theta(\mathbf{x}_0)\right] \\ &= L + \log p_\theta(\mathbf{x}_0). \end{aligned}$$

Η απόκλιση KL είναι πάντα μη αρνητικός αριθμός επομένως θα ισχύει $\log p_\theta(\mathbf{x}_0) \geq -L$. Συνεπώς η μεγιστοποίηση του λογαρίθμου της πιθανοφάνειας των δεδομένων ισοδυναμεί με ελαχιστοποίηση του κάτω φράγματος διασποράς L. Το L μπορεί να διασπαστεί σε επιμέρους τμήματα μετά από πράξεις ως εξής:

$$L = \mathbb{E}_q \Bigg[ \underbrace{D_{KL}(q(\mathbf{x}_T|\mathbf{x}_0) \parallel p_\theta(\mathbf{x}_T))}_{L_T} + \sum_{t=2}^{T} \underbrace{D_{KL}(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \parallel p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t))}_{L_{t-1}} \underbrace{- \log p_\theta(\mathbf{x}_0|\mathbf{x}_1)}_{L_0} \Bigg].$$

Συνεπώς για να εκτιμήσουμε την κατανομή $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$ αρκεί να ελαχιστοποιήσουμε κάθε όρο της παραπάνω ισότητας. Ο όρος $L_T$ δεν εξαρτάται από τις παραμέτρους $\theta$ μιας και η κατανομή $p_\theta(\mathbf{x}_T)$ είναι κανονική με μέση τιμή 0 και μοναδιαίο πίνακα διασποράς, επομένως μπορεί να παραλειφθεί κατά την ελαχιστοποίηση. Επίσης για τον πίνακα διασποράς της $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$ μπορεί να επιλεγεί σύμφωνα με τους [HJA20] μια σταθερή τιμή $\mathbf{\Sigma}_\theta(\mathbf{x}_t, t) = \beta_t \mathbb{I}$. Άρα κάθε όρος $L_{t-1}$ υπολογίζεται αναλυτικά ως εξής[5]

$$\begin{aligned}
L_{t-1} &= D_{KL}(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \parallel p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)) \\
&= \mathbb{E}_q \left[ \log \frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)} \right] \\
&= \mathbb{E}_q \left[ \frac{1}{2\beta_t} \|\tilde{\boldsymbol{\mu}}(\mathbf{x}_t, \mathbf{x}_0) - \boldsymbol{\mu}_\theta(\mathbf{x}_t, t)\|_2^2 \right] \qquad (2.8.8) \\
&\overset{2.8.6}{=} \mathbb{E}_q \left[ \frac{1}{2\beta_t} \| \frac{1}{\sqrt{\alpha_t}} \left( \mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \mathbf{z}_t \right) - \boldsymbol{\mu}_\theta(\mathbf{x}_t, t) \|_2^2 \right]. \qquad (2.8.9)
\end{aligned}$$

Κατά την εκπαίδευση οι τιμές $\mathbf{x}_t$ είναι γνωστές οπότε μπορούμε να μοντελοποιήσουμε τη μέση τιμή ως

$$\boldsymbol{\mu}_\theta(\mathbf{x}_t, t) = \frac{1}{\sqrt{\alpha_t}} \left( \mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \mathbf{z}_\theta(\mathbf{x}_t, t) \right),$$

ώστε κάθε όρος να γράφεται ως:

$$L_{t-1} = \mathbb{E}_{\mathbf{x}_0, \mathbf{z}} \left[ \frac{\beta_t}{2\alpha_t(1-\bar{\alpha}_{t-1})} \|\mathbf{z}_t - \mathbf{z}_\theta(\mathbf{x}_t, t)\|_2^2 \right].$$

Για απλότητα τελικά ελαχιστοποιείται ο όρος:

$$L_{t-1} = \mathbb{E}_{\mathbf{x}_0, \mathbf{z}} \left[ \|\mathbf{z}_t - \mathbf{z}_\theta(\mathbf{x}_t, t)\|_2^2 \right]. \qquad (2.8.10)$$

Εφόσον το μοντέλο εκπαιδευτεί κάθε επόμενο δείγμα στο reverse diffusion θα προκύπτει ως:

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left( \mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \mathbf{z}_\theta(\mathbf{x}_t, t) \right) + \sqrt{\beta_t} \mathbf{z}, \qquad \mathbf{z} \sim N(\mathbf{0}, \mathbb{I}). \qquad (2.8.11)$$

[5] Στη Σχέση 2.8.8 λαμβάνουμε υπόψιν ότι η απόκλιση KL δύο κανονικών κατανομών $P_1, P_2$ διάστασης $n$ με μέσες $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2$ και πίνακες διασπορών $\mathbf{\Sigma}_1, \mathbf{\Sigma}_2$, δίνεται από τον τύπο $\frac{1}{2}\left( \log \frac{\det \mathbf{\Sigma}_1}{\det \mathbf{\Sigma}_2} - n + \mathrm{tr}(\mathbf{\Sigma}_2^{-1}\mathbf{\Sigma}_1) + (\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)^T \mathbf{\Sigma}_2^{-1} (\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1) \right)$.

**Εκπαίδευση και δειγματοληψία στα diffusion probabilistic μοντέλα**

Σύμφωνα με όσα είδαμε παραπάνω μπορούμε να κατασκευάσουμε ένα μοντέλο που θα προβλέπει την ποσότητα $\mathbf{z}_\theta(\mathbf{x}_t, t)$. Εφόσον αυτό εκπαιδευτεί μπορούμε να εκτελέσουμε τη διαδικασία reverse diffusion προκειμένου να καταλήξουμε σε ένα δείγμα από την κατανομή των δεδομένων. Συγκεκριμένα για να εκπαιδεύσουμε το μοντέλο ξεκινάμε με δείγμα $\mathbf{x}_0 \sim q(\mathbf{x})$ από την πραγματική κατανομή και ορίζουμε το πλήθος βημάτων T του forward diffusion μαζί με τις παραμέτρους $\{\beta_t \in (0,1)\}_{t=1}^T$. Για κάθε $t = 1, \ldots, T$ υπολογίζουμε τα δείγματα $\mathbf{x}_t$ και εν συνεχεία εκτελούμε π.χ. τη μέθοδο Gradient Descent [Rud16] προκειμένου να βρούμε τις παραμέτρους $\theta$ του δικτύου που ελαχιστοποιούν την ποσότητα 2.8.10. Το $\mathbf{z}_t$ είναι μια τ.μ. από την τυποποιημένη κανονική κατανομή και ως είσοδο στο δίκτυο δίνουμε το δείγμα $\mathbf{x}_t$, που όπως είδαμε εξαρτάται από το αρχικό δείγμα $\mathbf{x}_0$ και την τ.μ. $\mathbf{z}_t$ σύμφωνα με τη σχέση $\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\mathbf{z}_t$. Ύστερα από την εκπαίδευση του δικτύου, προκειμένου να παράγουμε δείγμα από την πραγματική κατανομή ξεκινάμε πρώτα από την τ.μ. $\mathbf{x}_T \sim N(\mathbf{0}, \mathbb{I})$ και υπολογίζουμε προς τα πίσω τις τιμές $\mathbf{x}_{T-1}, \ldots, \mathbf{x}_0$ σύμφωνα με τη Σχέση 2.8.11. Η διαδικασία εκπαίδευσης και δειγματοληψίας παρουσιάζεται και στην Εικόνα 2.19.

**Algorithm 1** Training

1: **repeat**
2: $\mathbf{x}_0 \sim q(\mathbf{x}_0)$
3: $t \sim \text{Uniform}(\{1, \ldots, T\})$
4: $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
5: Take gradient descent step on
$$\nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t) \right\|^2$$
6: **until** converged

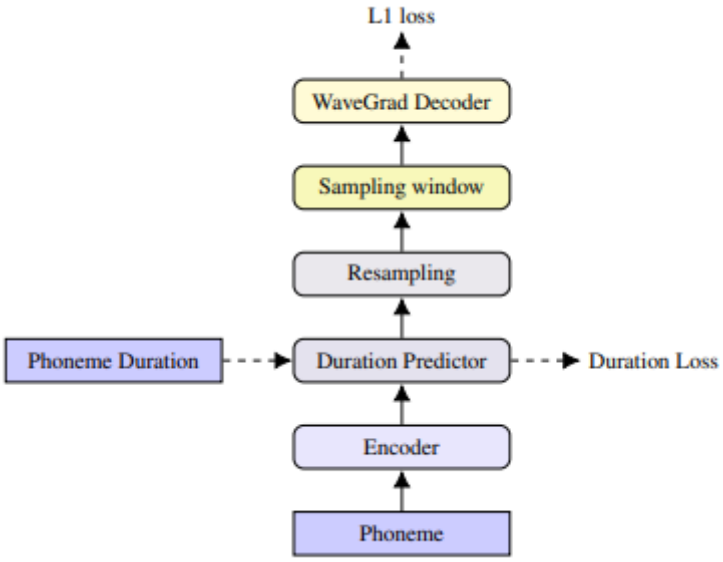
**Algorithm 2** Sampling

1: $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
2: **for** $t = T, \ldots, 1$ **do**
3: $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
4: $\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left( \mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) \right) + \sigma_t \mathbf{z}$
5: **end for**
6: **return** $\mathbf{x}_0$

Εικόνα 2.19: Η διαδικασία εκπαίδευσης και δειγματοληψίας στα diffusion probabilistic μοντέλα. [HJA20]

### 2.8.2 Αρχιτεκτονική του μοντέλου WaveGrad2

Η βασική ιδέα για την εκπαίδευση και τη συμπερασματολογία του μοντέλου Wavegrad2 στηρίζεται στα diffusion probabilistic μοντέλα που εξετάσαμε προηγουμένως. Σε αυτή την υποενότητα περιγράφουμε την αρχιτεκτονική του και ορισμένες μικρές παραλλαγές που χρησιμοποιεί αναφορικά με ότι είδαμε μέχρι τώρα. Όπως έχουμε αναφέρει το μοντέλο WaveGrad2 παράγει απευθείας μια κυματομορφή ήχου από το κείμενο εισόδου χωρίς να χρησιμοποιεί ενδιάμεσα χαρακτηριστικά όπως το φασματογράφημα στην κλίμακα mel. Αυτό φαίνεται και από την αρχιτεκτονική του στο Σχήμα 2.20, όπου σαν είσοδος δίνεται η ακολουθία φωνημάτων από το κείμενο για να παραχθεί η αντίστοιχη κυματομορφή ήχου. Τα βασικά στοιχεία της αρχιτεκτονικής του είναι ο encoder, ένα resampling layer και ο WaveGrad decoder. Αρχικά ο encoder που έχει δομή όπως και στο μοντέλο Tacotron2 (Embedding, 3 Conv, Bi-LSTM) δέχεται την ακολουθία φωνημάτων που αντιστοιχεί στο κείμενο εισόδου και έπειτα παράγει μια «κρυφή» αναπαράσταση από αυτή. Εκτός από τα tokens των φωνημάτων χρησιμοποιούνται και ορισμένα silence tokens ενδιάμεσα των λέξεων καθώς και ένα "eos" (end of sequence) token που υποδηλώνει το τέλος της ακολουθίας φωνημάτων. Έπειτα ακολουθεί το resampling layer προκειμένου να αντιστοιχίσει τη διάσταση της κρυφής αναπαράστασης του encoder στη διάσταση της κυματομορφής. Αντί για το μηχανισμό location sensitive attetion χρησιμοποιείται η μέθοδος Gaussian upsampling όπως στο μοντέλο non



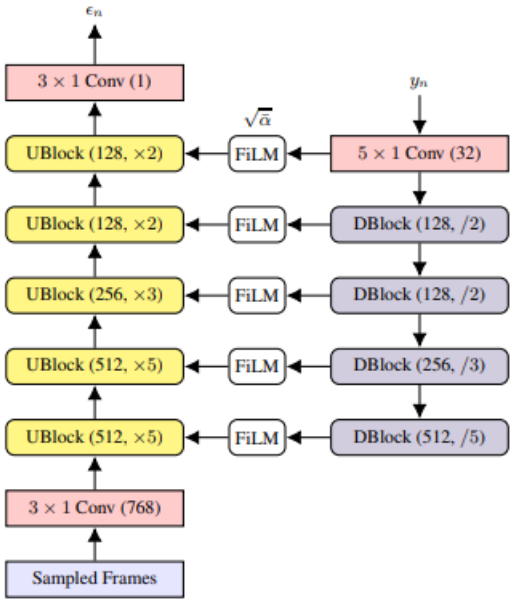

Σχήμα 2.20: Αρχιτεκτονική του μοντέλου του WaveGrad2. Το μοντέλο δέχεται ως είσοδο την ακολουθία φωνημάτων και παράγει την αντίστοιχη κυματομορφή. [Che+21]

attentive Tacotron [She+20]. Συγκεκριμένα για να φέρουμε τη διάσταση $L$ της αναπαράστασης $\mathbf{h} = (\mathbf{h}_1, \ldots, \mathbf{h}_L)$ του encoder σε μια μεγαλύτερη διάσταση $T$ λαμβάνουμε υπόψιν τη διάρκεια $\mathbf{d} = (d_1, \ldots, d_L)$ από κάθε φώνημα μαζί με ορισμένες τιμές $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_L)$ που υποδηλώνουν το εύρος επιρροής κάθε φωνήματος. Έπειτα παράγουμε την upsampled ακολουθία διανυσμάτων $\mathbf{U} = (\mathbf{u}_1, \ldots, \mathbf{u}_T)$, όπου κάθε στοιχείο της είναι το σταθμισμένο άθροισμα $\mathbf{u}_t = \sum_{i=1}^{L} w_{t,i}\mathbf{h}_i$ των στοιχείων της αναπαράστασης $\mathbf{h}$. Τα βάρη προκύπτουν σύμφωνα με τη σχέση

$$w_{t,i} = \frac{N(c_i, \sigma_i^2)}{\sum_{j=1}^{L} N(c_j, \sigma_j^2)}, \qquad \text{όπου } c_i = \frac{d_i}{2} + \sum_{j=1}^{i-1} d_j. \tag{2.8.12}$$

Στο Σχήμα 2.20 παρατηρούμε ότι υπάρχει και ένα δίκτυο πρόβλεψης (Duration Predictor) για τη διάρκεια κάθε φωνήματος στην ακολουθία εισόδου. Το δίκτυο αυτό αποτελείται από δύο υπο-δίκτυα, ένα για την πρόβλεψη της διάρκειας $d_i$ και ένα για την πρόβλεψη του θετικού εύρους $\sigma_i$. Κάθε υπο-δίκτυο έχει ένα bidirectional LSTM και ένα γραμμικό επίπεδο που δέχεται την ακολουθία του encoder και δίνει ως έξοδο είτε τη διάρκεια $d_i$ είτε το εύρος $\sigma_i$[6] για κάθε $i = 1, \ldots, L$. Για την εκπαίδευσή του Duration Predictor χρησιμοποιείται το MSE loss μεταξύ της διάρκειας που προβλέπει και της πραγματικής διάρκειας για κάθε φώνημα. Αφού γίνει το upsampling της αναπαράστασης του encoder ακολουθεί το Sampling window, όπου κατά την εκπαίδευση λαμβάνουμε σε κάθε batch ένα τμήμα (segment) από το upsampled διάνυσμα του encoder προκειμένου να υπολογίσουμε το σφάλμα. Αυτό γίνεται διότι λόγω μνήμης δεν μπορούμε να υπολογίσουμε το σφάλμα για όλες τις τιμές της κυματομορφής, οπότε επιλέγουμε τυχαία ένα μικρότερο τμήμα από την έξοδο του resampling layer. Τέλος έχουμε τον decoder που χρησιμοποιείται και στο μοντέλο WaveGrad, ο οποίος φαίνεται στο Σχήμα 2.21.

[6] για το εύρος στο γραμμικό επίπεδο εφαρμόζεται η συνάρτηση ενεργοποίησης $\text{Softplus}(x) = \frac{1}{\beta}\log(1 + e^{\beta x})$.




Σχήμα 2.21: Η δομή του WaveGrad decoder. [Che+20]

### WaveGrad Decoder

Η λειτουργία του decoder στηρίζεται στα diffusion probabilistic μοντέλα που αναλύσαμε προηγουμένως. Κατά την εκπαίδευση υλοποιείται το forward diffusion ενώ για τη συμπερασματολογία το reverse diffusion. Πιο συγκεκριμένα κατά την εκπαίδευση, ξεκινώντας με μια κυματομορφή $\mathbf{y}_0$ από την κατανομή των δεδομένων, σε κάθε βήμα του forward diffusion δημιουργείται μια «θορυβώδης» κυματομορφή $\mathbf{y}_n$. Ο decoder δέχεται την κυματομορφή $\mathbf{y}_n$, το επίπεδο θορύβου $\sqrt{\bar{\alpha}}$ καθώς και το τμήμα της upsampled αναπαράστασης του encoder (Sampled Frames) και εξάγει την ποσότητα $\epsilon_n$ που χρησιμοποιείται για να υπολογίσουμε την κυματομορφή $y_{n+1}$ του επόμενου βήματος. Τα βήματα της διαδικασίας φαίνονται στην Εικόνα 2.22. Ορισμένες μικρές διαφορές σε

**Algorithm 1** Training. WaveGrad directly conditions on the continuous noise level $\sqrt{\bar{\alpha}}$. $l$ is from a predefined noise schedule.

**repeat**
  $y_0 \sim q(y_0)$
  $s \sim \text{Uniform}(\{1, \dots, S\})$
  $\sqrt{\bar{\alpha}} \sim \text{Uniform}(l_{s-1}, l_s)$
  $\epsilon \sim \mathcal{N}(0, I)$
  Take gradient descent step on
    $\nabla_\theta \left\| \epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}}\, y_0 + \sqrt{1-\bar{\alpha}}\, \epsilon, x, \sqrt{\bar{\alpha}}) \right\|_1$
**until** converged

**Algorithm 2** Sampling. WaveGrad generates samples following a gradient-based sampler similar to Langevin dynamics.

$y_N \sim \mathcal{N}(0, I)$
**for** $n = N, \dots, 1$ **do**
  $z \sim \mathcal{N}(0, I)$
  $y_{n-1} = \frac{\left(y_n - \frac{1-\alpha_n}{\sqrt{1-\bar{\alpha}_n}} \epsilon_\theta(y_n, x, \sqrt{\bar{\alpha}_n})\right)}{\sqrt{\alpha_n}}$
  if $n > 1$, $y_{n-1} = y_{n-1} + \sigma_n z$
**end for**
**return** $y_0$

Εικόνα 2.22: Η διαδικασία εκπαίδευσης και δειγματοληψίας στο μοντέλο WaveGrad. [Che+20]

σχέση με το πλαίσιο που είδαμε στα diffusion μοντέλα είναι ότι στο WaveGrad το δίκτυο $\epsilon_\theta$ εκτός από την κυματομορφή $\mathbf{y}_n$ δεσμεύεται και από το επίπεδο θορύβου $\sqrt{\bar{\alpha}}$ αλλά και από το upsampled τμήμα $x$ της αναπαράστασης του encoder. Επίσης για την εκπαίδευση του ελαχιστοποιούμε την

$L_1$ νόρμα:

$$\mathbb{E}_{\bar{\alpha},\epsilon}\left[\|\epsilon_\theta(\sqrt{\bar{\alpha}}\mathbf{y}_0+\sqrt{1-\bar{\alpha}}\epsilon,x,\sqrt{\bar{\alpha}})-\epsilon\|_1\right]. \tag{2.8.13}$$

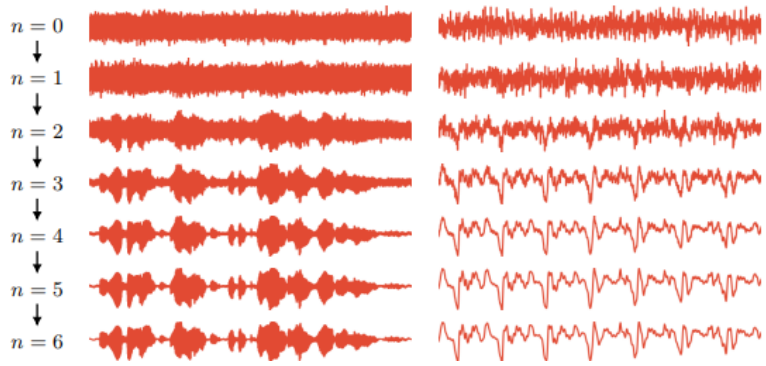
Το επίπεδο θορύβου υπολογίζεται από τη σχέση:

$$\sqrt{\bar{\alpha}}=\text{Uniform}(l_{s-1},l_s),\qquad \text{με } l_s=\sqrt{\prod_{i=1}^{s}(1-\beta_i)},\quad l_0=1, \tag{2.8.14}$$

όπου τα $\{\beta_i\in(0,1)\}_{i=1}^{S}$ είναι οι παράμετροι διασποράς που ορίζονται εξαρχής μαζί με πλήθος των βημάτων της διαδικασίας diffusion. Αφού εκπαιδεύσουμε το δίκτυο $\epsilon_\theta$ ο decoder εκτελεί την αντίστροφη διαδικασία για να παράγει τελικά μια κυματομορφή από την κατανομή των δεδομένων. Ξεκινώντας από δείγμα $\mathbf{y}_N\sim N(\mathbf{0},\mathbb{I})$ σε κάθε βήμα χρησιμοποιούμε τη σχέση:

$$\mathbf{y}_{n-1}=\frac{1}{\sqrt{\alpha_n}}\left(\mathbf{y}_n-\frac{\beta_n}{\sqrt{1-\bar{\alpha}_n}}\epsilon_\theta(\mathbf{y}_n,x,\sqrt{\alpha_n})\right)+\sqrt{\beta_n}\epsilon,\qquad \epsilon\sim N(\mathbf{0},\mathbb{I}), \tag{2.8.15}$$

μέχρι να καταλήξουμε στην κυματομορφή $\mathbf{y}_0$. Στο Σχήμα 2.23 φαίνεται η διαδικασία denoising στο μοντέλο WaveGrad.

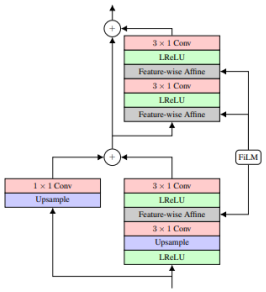



Σχήμα 2.23: Denoising στο μοντέλο WaveGrad. Δεξιά φαίνεται ένα τμήμα 50 ms από την αντίστοιχη κυματομορφή στα αριστερά. [Che+20]

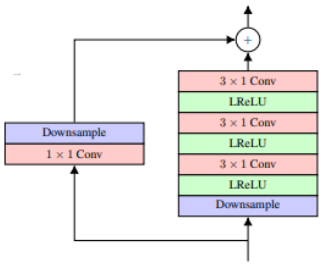
Όσον αφορά την αρχιτεκτονική του decoder παρατηρούμε από το Σχήμα 2.21 ότι αποτελείται από 5 upsampling blocks (UBlock), 4 downsampling blocks (DBlock) και 5 FiLM (Feature-wise Linear Modulation) modules [Dum+18], η αρχιτεκτονική των οποίων φαίνεται στο Σχήμα 2.24. Τα UBlock ουσιαστικά κάνουν upsampling του τμήματος της αναπαράστασης του encoder (Sampled Frames) ώστε η έξοδος να έχει ίδια διάσταση με τη κυματομορφή. Αποτελούνται κυρίως από συνελικτικά επίπεδα με dilation και αυξάνουν τη διάσταση σύμφωνα με τους παράγοντες 5,5,3,2,2. Επιπλέον χρησιμοποιούν πληροφορία και από την έξοδο του module FiLM στο επίπεδο Feature-Wise Affine σύμφωνα με τη σχέση:

$$\gamma(D,\sqrt{\bar{\alpha}})\odot U+\xi(D,\sqrt{\bar{\alpha}}), \tag{2.8.16}$$

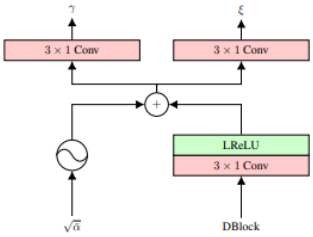
όπου $\gamma,\xi$ είναι οι έξοδοι από το module FiLM και $U$ είναι μια ενδιάμεση έξοδος εσωτερικά του UBlock. Το FiLM δέχεται ως είσοδο το επίπεδο θορύβου $\sqrt{\bar{\alpha}}$ και την έξοδο $D$ από το εκάστοτε DBlock. Τέλος κάθε Dblock κάνει downsampling στην κυματομορφή $\mathbf{y}_n$ με χρήση συνελικτικών επιπέδων με strides.




(a) UBlock




(b) DBlock




(c) FiLM

Σχήμα 2.24: Αρχιτεκτονική των submodules UBlock, DBlock και FiLM που χρησιμοποιούνται στον WaveGrad decoder. [Che+20]

### 2.8.3 Αποτελέσματα και αξιολόγηση

Το μοντέλο WaveGrad2 εκπαιδεύτηκε σε ένα εσωτερικό σύνολο δεδομένων στην Αγγλική γλώσσα με περίπου 39 ώρες ηχογραφήσεων. Επιπλέον έγιναν συγκρίσεις με άλλα end-to-end μοντέλα που χρησιμοποιούν το Tacotron2 για την παραγωγή φασματογραφήματος στην κλίμακα mel και έναν vocoder για την εξαγωγή της κυματομορφής. Για την αξιολόγησή τους χρησιμοποιήθηκε η μετρική του MOS στην κλίμακα 1-5 με διαβαθμίσεις της τάξης του 0.5. Τα αποτελέσματα που προέκυψαν φαίνονται στον Πίνακα 2.25, όπου για τα μοντέλα WaveGrad και WaveGrad2 χρησιμοποιήθηκαν 1000 βήματα κατά την diffusion διαδικασία (refinement steps).

| **Model** | **Model size** | **MOS (↑)** |
|---|---|---|
| Tacotron 2 + WaveRNN | 38M + 18M | 4.49 ± 0.04 |
| Tacotron 2 + WaveGrad(Base, 1000) | 38M + 15M | 4.47 ± 0.04 |
| Tacotron 2 + WaveGrad(Large, 1000) | 38M + 23M | 4.51 ± 0.04 |
| Tacotron 2 + MelGAN | 38M + 3M | 3.95 ± 0.06 |
| Tacotron 2 + GAN-TTS | 38M + 21M | 4.34 ± 0.04 |
| Wave-Tacotron [33] | 38M | 4.08 ± 0.06 |
| WaveGrad 2 | | |
| Encoder(2048) + WaveGrad(Large, 1000) | 193M | 4.37 ± 0.05 |
| Encoder(2048) + WaveGrad(Large, 1000) + MT | 193M | 4.39 ± 0.05 |
| Encoder(1024) + WaveGrad(Large, 1000) + MT + SpecAug | 73M | 4.43 ± 0.05 |
| Ground Truth | – | 4.58 ± 0.05 |

Πίνακας 2.25: Αποτελέσματα σύγκρισης του μοντέλου WaveGrad2 με άλλα end-to-end μοντέλα σύνθεσης φωνής από κείμενο. [Che+21]

Παρατηρούμε ότι το μοντέλο Tacotron2 μαζί με το WaveGrad (Large) δίνει καλύτερο MOS με τιμή 4.51, ενώ το μοντέλο WaveGrad2 και για τις τρεις υποπεριπτώσεις παράγει ικανοποιητικά αποτελέσματα σε σχέση με τις υπόλοιπες μεθόδους με βέλτιστο MOS $\approx 4.43$. Η τιμή του MOS

για τις πραγματικές ηχογραφήσεις ήταν περίπου στο 4.58. Επιπλέον από συγκρίσεις όσον αφορά το μέγεθος του μοντέλου WaveGrad2 προκύπτει ότι αύξηση των παραμέτρων του decoder είναι σημαντικότερη από την αύξηση των παραμέτρων στον encoder. Επίσης έγινε και δοκιμή της μεθόδου Spec-Augment [Par+19] στην upsampled αναπαράσταση του encoder που όμως επέφερε μικρή βελτίωση στο MOS. Λαμβάνοντας υπόψιν όλα τα παραπάνω συμπεραίνουμε ότι το μοντέλο WaveGrad2 μπορεί να παράγει ηχητικά δείγματα υψηλής ποιότητας χρησιμοποιώντας τεχνικές από τα diffusion probabilistic μοντέλα. Το βασικό θετικό στοιχείο του είναι ότι δεν χρειάζεται να παράγει ενδιάμεσα χαρακτηριστικά, όπως άλλα state-of-the-art end-to-end μοντέλα στη σύνθεση φωνής.

## 2.9 Συμπέρασμα

Στο κεφάλαιο αυτό μελετήσαμε ορισμένες από τις state of the art αρχιτεκτονικές για το πρόβλημα της σύνθεσης φωνής από κείμενο. Όπως είδαμε ο παραγόμενος ήχος αγγίζει τα ανθρώπινα επίπεδα για την πλειοψηφία των μοντέλων που εξετάσαμε. Παρ' όλο όμως που τα αποτελέσματα αυτά είναι αρκετά ικανοποιητικά, σχεδόν όλα τα μοντέλα υστερούν σημαντικά ως προς τον μεγάλο χρόνο αλλά και τους υπολογιστικούς πόρους που απαιτούνται κατά την εκπαίδευσή τους. Στη συνέχεια της εργασίας μελετάμε κάποια ειδικά θέματα που αφορούν το πρόβλημα της σύνθεσης φωνής, όπως είναι η προσθήκη προσωδίας και συναισθήματος στο παραγόμενο ηχητικό δείγμα. Επίσης μελετάμε το πρόβλημα της σύνθεσης για γλώσσες στις οποίες είναι διαθέσιμα λίγα δεδομένα (low resource languages) και τέλος γίνεται αναφορά και σε έναν επιπλέον τρόπο αξιολόγησης των αποτελεσμάτων της σύνθεσης φωνής.

# Κεφάλαιο 3

# Ειδικά θέματα σύνθεσης φωνής από κείμενο

## 3.1 Εισαγωγή

Στο παρόν κεφάλαιο εξετάζουμε ορισμένα ειδικά θέματα πάνω στη σύνθεση φωνής από κείμενο. Στην πρώτη ενότητα παρουσιάζεται η μέθοδος της επαύξησης δεδομένων για την παραγωγή εκφραστικών ηχητικών δειγμάτων από έναν ομιλητή με λίγα διαθέσιμα δεδομένα. Η προσέγγιση αυτή αξιοποιεί τεχνικές όπως η μετατροπή φωνής για να δημιουργήσει τεχνητά δεδομένα από έναν target ομιλητή, τα οποία στη συνέχεια χρησιμοποιούνται για να εκπαιδευτεί ένα μοντέλο σύνθεσης φωνής πάνω σε πραγματικά και σε συνθετικά δεδομένα. Στη συνέχεια παρουσιάζουμε τη μέθοδο LRSpeech [Xu+20], η οποία είναι χρήσιμη στην περίπτωση όπου θέλουμε να παράγουμε φωνή σε μια γλώσσα όπου υπάρχουν πολύ λίγα διαθέσιμα δεδομένα ηχογραφήσεων. Με τη συγκεκριμένη μέθοδο αξιοποιούνται γλώσσες, για τις οποίες υπάρχουν αρκετές ώρες ηχογραφήσεων ώστε να εκπαιδευτούν μοντέλα σύνθεσης αλλά και αναγνώρισης φωνής με στόχο τη δημιουργία νέων συνθετικών δεδομένων στη low resource γλώσσα. Τα δεδομένα αυτά μπορούν να χρησιμοποιηθούν εκ νέου για εκπαίδευση και να βελτιώσουν περαιτέρω τα αποτελέσματα για τη γλώσσα που επιθυμούμε. Έπειτα εξετάζουμε το μοντέλο Global Style Tokens (GST) [Wan+18] μέσω του οποίου μπορούμε να ρυθμίσουμε την εκφραστικότητα, την προσωδία και γενικότερα το ύφος και τον τόνο σε ένα σύστημα παραγωγής ομιλίας. Το μοντέλο αυτό υπολογίζει με έναν μη-επιβλεπόμενο τρόπο ορισμένα style tokens, καθένα από τα οποία υποδηλώνει ένα χαρακτηριστικό της ομιλίας. Τέλος κάνουμε μια αναφορά στην αξιολόγηση των συστημάτων σύνθεσης φωνής και τονίζεται η ανάγκη για εύρεση μεθόδων, στις οποίες η αξιολόγηση δε θα βασίζεται μόνο σε μεμονωμένες προτάσεις αλλά και σε παραγράφους ή συνδυασμό αυτών. Κατ' αυτό τον τρόπο θα μπορούν να λαμβάνονται υπόψιν το περιεχόμενο και το συναίσθημα μιας πρότασης μέσα σε ένα κείμενο και έτσι η αξιολόγηση της θα μπορεί να είναι πιο αντιπροσωπευτική.

## 3.2 Σύνθεση φωνής από κείμενο με εκφραστικότητα χρησιμοποιώντας επαύξηση δεδομένων

Τα περισσότερα end-to-end μοντέλα σύνθεσης φωνής από κείμενο χρειάζονται κατά την εκπαίδευσή τους ένα μεγάλο πλήθος δεδομένων. Τα δεδομένα αυτά αποτελούνται από ζεύγη κειμένου και ήχου. Για παράδειγμα το σύνολο δεδομένων LJ Speech, στο οποίο έχουν εκπαιδευτεί μοντέλα

όπως το Tacotron2, περιέχει συνολικά 13100 ηχογραφήσεις συνολικής διάρκειας περίπου 24 ωρών. Σε συγκεκριμένες εφαρμογές χρειάζεται η εκπαίδευση ενός μοντέλου ώστε να παράγει ομιλία από έναν συγκεκριμένο ομιλητή (target speaker) και σε ένα συγκεκριμένο στυλ. Τις περισσότερες φορές το πλήθος των ηχογραφήσεων που διαθέτουμε από αυτόν τον ομιλητή δεν επαρκεί για την εκπαίδευση ενός end to end μοντέλου και κατά συνέπεια τα αποτελέσματα που λαμβάνουμε δεν είναι ικανοποιητικά. Η συνηθέστερη τεχνική που αξιοποιείται για την επίλυση αυτού του προβλήματος είναι η μεταφορά μάθησης (transfer learning) [Jia+18]. Σύμφωνα με αυτή τη μέθοδο, πραγματοποιείται μεταφορά της γνώσης που έχει αποκτηθεί από την εκπαίδευση ενός μοντέλου σε δεδομένα ηχογραφήσεων μεγάλης διάρκειας και από διάφορους ομιλητές. Μία άλλη τεχνική με πολλά πλεονεκτήματα και σε άλλες εφαρμογές πέραν της σύνθεσης φωνής είναι η επαύξηση των δεδομένων (data augmentation) [SK19]. Με τον όρο αυτό εννοούμε την κατασκευή τεχνητών δεδομένων τα οποία μπορούν να χρησιμοποιηθούν μαζί με τα πραγματικά δεδομένα κατά την εκπαίδευση ενός μοντέλου για τη βελτίωση των αποτελεσμάτων του. Στην περίπτωσή μας το data augmentation αφορά τη δημιουργία συνθετικών ηχογραφήσεων για έναν συγκεκριμένο ομιλητή. Οι συνθετικές αυτές ηχογραφήσεις μπορούν να αξιοποιηθούν για την εκπαίδευση ενός μοντέλου σύνθεσης φωνής που θα παράγει τελικά εκφωνήσεις με τη φωνή του ομιλητή που επιθυμούμε (target speaker). Μία άλλη προσέγγιση που εκμεταλλεύεται την τεχνική της μετατροπής φωνής (Voice Conversion) παρουσιάζεται στην εργασία με τίτλο "*Low-Resource Expressive Text-To-Speech Using Data Augmentation*" [Huy+21]. Η τεχνική αυτή στοχεύει στη μετατροπή των εκφωνήσεων ενός ή περισσότερων ομιλητών, ώστε αυτές να είναι σα να έχουν παραχθεί από έναν συγκεκριμένο ομιλητή, διατηρώντας βέβαια το στυλ και το περιεχόμενο της αρχικής εκφώνησης. Η μεθοδολογία που χρησιμοποιείται αποτελείται από τρία βήματα τα οποία παρουσιάζουμε στη συνέχεια.

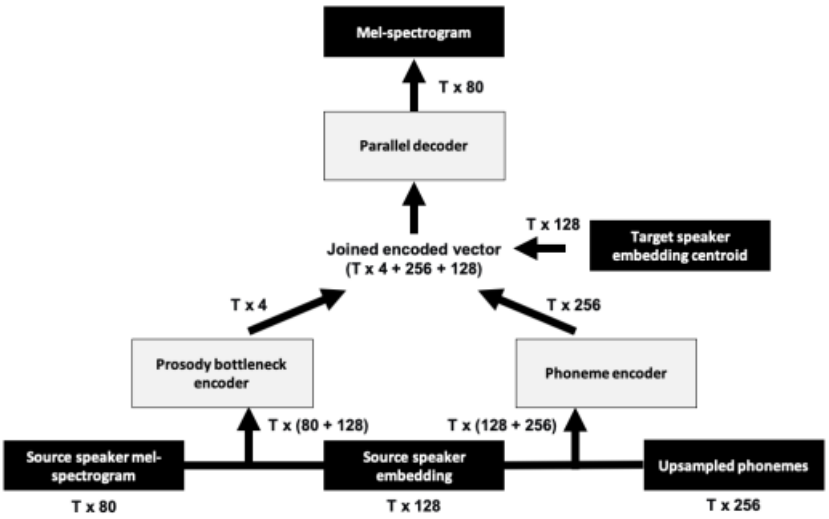
### 3.2.1 Μεθοδολογία

Στο σημείο αυτό περιγράφουμε τα βήματα που ακολουθούνται βάσει της μεθόδου επαύξησης δεδομένων και της μετατροπής φωνής. Έστω ότι θέλουμε να παράγουμε φωνή από έναν συγκεκριμένο ομιλητή και σε ένα συγκεκριμένο στυλ, αλλά διαθέτουμε λίγες ηχογραφήσεις από αυτόν. Μπορούμε να χρησιμοποιήσουμε ηχογραφήσεις από διαφορετικούς ομιλητές, οι οποίες όμως είναι στο στυλ που επιθυμούμε. Η μέθοδος που ακολουθείται είναι η εξής. Σε πρώτη φάση εκπαιδεύεται ένα μοντέλο μετατροπής φωνής το οποίο κατασκευάζει συνθετικές εκφωνήσεις για τον συγκεκριμένο ομιλητή, χρησιμοποιώντας δεδομένα που υπάρχουν από τους υπόλοιπους ομιλητές. Στη συνέχεια οι συνθετικές αυτές εκφωνήσεις σε συνδυασμό με τις πραγματικές εκφωνήσεις του ομιλητή χρησιμοποιούνται για την εκπαίδευση ενός μοντέλου σύνθεσης φωνής πάνω σε ηχογραφήσεις μεγάλης συνολικής διάρκειας (συνθετικές και πραγματικές) από τον target ομιλητή. Τέλος, το μοντέλο αυτό εκπαιδεύεται περαιτέρω χρησιμοποιώντας όμως μόνο τις πραγματικές εκφωνήσεις του ομιλητή. Το τελευταίο βήμα είναι γνωστό ως fine tuning.

#### Επαύξηση δεδομένων

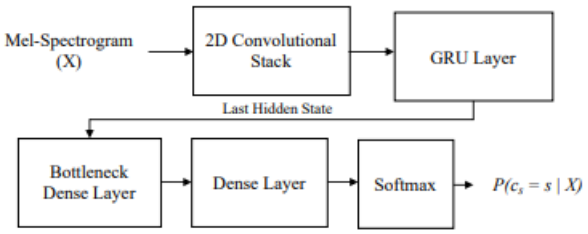
Το πρώτο βήμα της μεθόδου είναι η δημιουργία συνθετικών δεδομένων για έναν συγκεκριμένο ομιλητή με τη τεχνική της μετατροπής φωνής. Το μοντέλο που χρησιμοποιείται για τη μετατροπή φωνής είναι το CopyCat [Kar+20], του οποίου η αρχιτεκτονική φαίνεται στο Σχήμα 3.1.

Τα βασικά modules που δομούν το συγκεκριμένο μοντέλο είναι ο phoneme encoder, ο prosody bottleneck encoder και ο parallel decoder. Ο phoneme encoder δέχεται ως είσοδο τα upsampled φωνήματα μαζί με το embedding από έναν ομιλητή (source speaker), του οποίου θέλουμε να μετατρέψουμε τη φωνή. Η διάσταση των upsampled φωνημάτων επιλέγεται να είναι ίση με το




Σχήμα 3.1: Αρχιτεκτονική του μοντέλου CopyCat για τη μετατροπή φωνής. [Huy+21]

πλήθος των frames (T) του αντίστοιχου φασματογραφήματος. Για να επιτευχθεί αυτό, σε πρώτη φάση γίνεται ένα alignment μεταξύ των φωνημάτων και της κυματομορφής, ώστε να εκτιμηθεί η διάρκεια κάθε φωνήματος και έπειτα με βάση αυτή τη διάρκεια γίνεται η αντιστοίχιση κάθε φωνήματος στο ανάλογο πλήθος των frames στο φασματογράφημα. Στη συνέχεια ο phoneme encoder εξάγει μια αναπαράσταση δεχόμενος ως είσοδο το άθροισμα των φωνημάτων και το embedding του αρχικού ομιλητή. Το embedding ενός ομιλητή (source ή target) προκύπτει από το δίκτυο που φαίνεται στο Σχήμα 3.2. Συγκεκριμένα το δίκτυο δέχεται ως είσοδο ένα φασματογράφημα στην



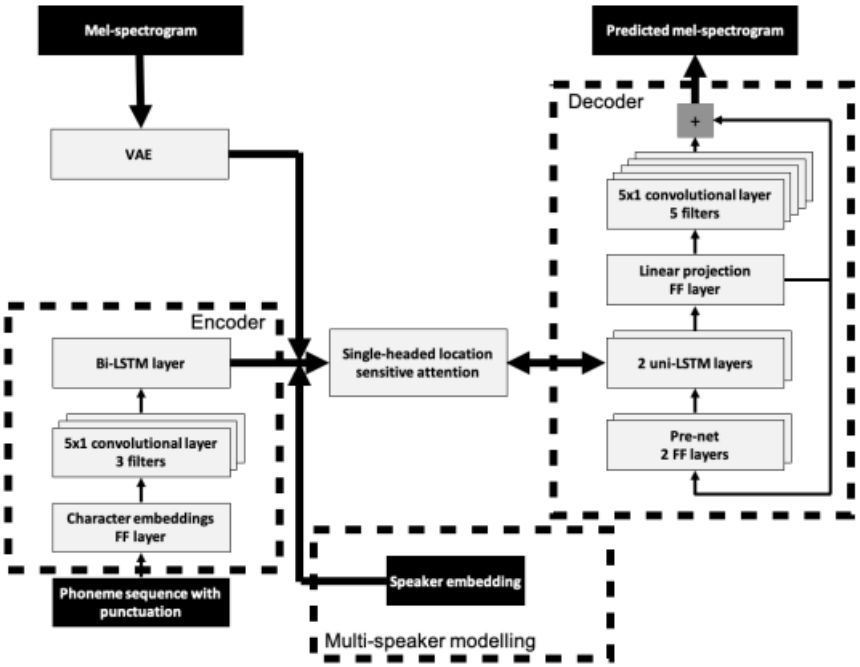

Σχήμα 3.2: Δίκτυο ταξινόμησης των ομιλητών (speaker classifier). Το embedding ενός ομιλητή εξάγεται από το bottleneck layer. [Kar+20]

κλίμακα mel και προβλέπει τον ομιλητή από τον οποίο προέρχεται. Αποτελείται από δύο συνελικτικά επίπεδα δύο διαστάσεων, ένα GRU layer από το οποίο κάθε hidden state εισέρχεται σε ένα bottleneck layer και τέλος ένα fully connected επίπεδο (dense layer), όπου με εφαρμογή της συνάρτησης softmax δίνει τις τελικές πιθανότητες για κάθε ομιλητή. Ύστερα από την εκπαίδευση του δικτύου, το embedding ενός ομιλητή (για το φασματογράφημα της εισόδου) προκύπτει από την έξοδο του bottleneck layer. Στο μοντέλο μας τώρα, ο prosody bottleneck encoder δέχεται επίσης ως είσοδο το embedding του αρχικού ομιλητή μαζί με το αντίστοιχο φασματογράφημα στην κλίμακα mel και εξάγει την αναπαράσταση ορισμένων χαρακτηριστικών που σχετίζονται με το στυλ της αρχικής εκφώνησης. Οι αναπαραστάσεις που προκύπτουν από τους encoders συνενώνονται με το embedding του ομιλητή, του οποίου θέλουμε να παράγουμε τη φωνή (target speaker). Για την ακρίβεια πρόκειται για το κεντροειδές των embeddings του target speaker, το οποίο υπ-

ολογίζεται ως ο μέσος όρος των embeddings του, που αντιστοιχούν στις εκφωνήσεις του συνόλου εκπαίδευσης. Δηλαδή για κάθε φασματογράφημα στο σύνολο εκπαίδευσης που προέρχεται από τον target ομιλητή, υπολογίζουμε τα αντίστοιχα embeddings και λαμβάνουμε το μέσο όρο αυτών. Τέλος, ο parallel decoder που αποτελείται από τρία συνελικτικά δίκτυα και ένα bidirectional-GRU, αναλαμβάνει την αποκωδικοποίηση των αναπαραστάσεων που έχουν προκύψει και εξάγει το φασματογράφημα στην κλίμακα mel που αντιστοιχεί στον target speaker. Κατά την εκπαίδευση το φασματογράφημα της εξόδου είναι ίδιο με αυτό της εισόδου. Στη συνέχεια περιγράφουμε τη δομή του μοντέλου σύνθεσης φωνής.

### Μοντέλο σύνθεσης φωνής

Όπως είδαμε το μοντέλο CopyCat παράγει φασματογραφήματα που αντιστοιχούν σε συνθετικές εκφωνήσεις από έναν συγκεκριμένο ομιλητή. Οι εκφωνήσεις αυτές μπορούν να χρησιμοποιηθούν σε συνδυασμό με τις πραγματικές εκφωνήσεις του ομιλητή, ώστε να εκπαιδεύσουμε ένα μοντέλο σύνθεσης φωνής με αρκετά δεδομένα. Στο Σχήμα 3.3 φαίνεται η αρχιτεκτονική ενός τέτοιου μοντέλου. Το μοντέλο αυτό εκπαιδεύεται είτε χρησιμοποιώντας εκφωνήσεις από έναν (source) ομιλητή σε διάφορα στυλ (multi style-single speaker), είτε χρησιμοποιώντας εκφωνήσεις από πολλούς ομιλητές σε ένα στυλ (single style-multi speaker).




Σχήμα 3.3: Αρχιτεκτονική του μοντέλου σύνθεσης φωνής. [Huy+21]

Η δομή του βασίζεται στο μοντέλο Tacotron [Wan+17] αλλά χρησιμοποιείται επιπλέον και ένας VAE [KW13] που επεξεργάζεται το φασματογράφημα που εξάγεται από το μοντέλο CopyCat. Ο encoder παράγει την αναπαράσταση της ακολουθίας εισόδου των φωνημάτων μαζί τα σημεία στίξης. Στην περίπτωση όπου το μοντέλο εκπαιδεύεται με εκφωνήσεις από διάφορους ομιλητές (multi-speaker modelling), τότε η έξοδος του encoder συνενώνεται με το embedding του ομιλητή καθώς και με την έξοδο από τον VAE. Σε διαφορετική περίπτωση δε χρησιμοποιείται το embedding του ομιλητή. Ύστερα ο decoder μαζί με έναν μηχανισμό προσοχής επεξεργάζεται

τις αναπαραστάσεις που έχουν προκύψει για να προβλέψει το τελικό φασματογράφημα στην κλίμακα mel. Τέλος το μοντέλο parallel WaveNet [Oor+18] χρησιμοποιείται ως ένας vocoder για να μετατρέψει το φασματογράφημα στην αντίστοιχη κυματομορφή. Το μοντέλο εκπαιδεύτηκε για περίπου 400 χιλιάδες βήματα με μέγεθος batch ίσο με 32. Ως μετρική για την εκπαίδευση του VAE χρησιμοποιήθηκε η απόκλιση Kullback-Leibler, ενώ για το φασματογράφημα στην έξοδο του decoder η L1 απόσταση.

**Fine tuning**

Ύστερα από την εκπαίδευση του μοντέλου με χρήση όλων των δεδομένων (συνθετικών και πραγματικών), πραγματοποιείται εκ νέου η εκπαίδευση του ίδιου μοντέλου για 4 χιλιάδες βήματα. Στη συγκεκριμένη φάση χρησιμοποιούνται μόνο τα πραγματικά δεδομένα από τον target ομιλητή, έτσι ώστε το μοντέλο να εστιάσει αποκλειστικά στον τρόπο εκφώνησης αλλά και στο στυλ του.

### 3.2.2 Αποτελέσματα-Αξιολόγηση

Στην ενότητα αυτή περιγράφουμε τα αποτελέσματα από την εκπαίδευση των μοντέλων. Τα πειράματα έγιναν πάνω σε ορισμένα σύνολα δεδομένων που περιείχαν εκφωνήσεις στην αγγλική γλώσσα (American english). Για το στυλ των παραγόμενων εκφωνήσεων εξετάστηκαν δύο περιπτώσεις. Η πρώτη αφορούσε ηχογραφήσεις σε στυλ εκφώνησης ειδήσεων (newscaster) και η δεύτερη σε στυλ συζήτησης (conversational). Για την κάθε περίπτωση χρησιμοποιήθηκαν μόλις 30 λεπτά ηχογραφήσεων από τον target ομιλητή. Επίσης για τα πειράματα χρησιμοποιήθηκε ο εξής συμβολισμός: τα $S_{i,j,k}$ και $S^*_{i,j,k}$ υποδηλώνουν τα σύνολα δεδομένων με πραγματικές και συνθετικές (voice converted) εκφωνήσεις αντίστοιχα, που προέρχονται από τον ομιλητή $i$ στο στυλ $j$ και έχουν συνολική διάρκεια $k$ ώρες. Επιπλέον το $\text{VC}(\{S_{i,j,k}\})$ υποδηλώνει το μοντέλο μετατροπής φωνής που εκπαιδεύεται στο σύνολο πραγματικών δεδομένων $\{S_{i,j,k}\}$, ενώ το $\text{TTS}(\{S^*_{i,j,k}\})$ υποδηλώνει το μοντέλο σύνθεσης φωνής που εκπαιδεύεται στο σύνολο συνθετικών δεδομένων $\{S^*_{i,j,k}\}$. Τέλος, ο συμβολισμός $\text{FT}(\text{TTS}, S_{i,j,k})$ υποδηλώνει το fine tuning του μοντέλου σύνθεσης φωνής TTS πάνω στις πραγματικές εκφωνήσεις $S_{i,j,k}$.

Το πρώτο πείραμα αφορά τη δημιουργία συνθετικών εκφωνήσεων για έναν ομιλητή σε στυλ εκφώνησης ειδήσεων (single speaker TTS-newscaster style). Συγκεκριμένα υπάρχουν δύο ομιλήτριες Female 1 και Female 2 για τις οποίες υπάρχουν αρκετές ηχογραφήσεις σε ουδέτερο στυλ (neutral). Για την κάθε ομιλήτρια εκπαιδεύεται ένα μοντέλο μετατροπής φωνής, ένα μοντέλο σύνθεσης φωνής και ακολουθεί fine tuning. Συγκεκριμένα για την πρώτη ομιλήτρια έχουμε τα εξής:

1. $\text{VC}_1(S_{1,neutral,20h}, S_{1,news,0.5h}, S_{2,neutral,20h}, S_{2,news,7h})$
2. $\text{TTS}_1(S_{1,neutral,20h}, S_{1,news,0.5h}, S^*_{1,news,7h})$
3. $\text{FT}(\text{TTS}_1, S_{1,news,0.5h})$

ενώ για τη δεύτερη ομιλήτρια:

1. $\text{VC}_2(S_{1,neutral,20h}, S_{1,news,4h}, S_{2,neutral,20h}, S_{2,news,0.5h})$
2. $\text{TTS}_2(S_{2,neutral,20h}, S_{2,news,0.5h}, S^*_{2,news,4h})$
3. $\text{FT}(\text{TTS}_2, S_{2,news,0.5h})$

Από τα παραπάνω βλέπουμε ότι σε πρώτη φάση το μοντέλο μετατροπής φωνής VC δημιουργεί συνθετικά δεδομένα τα οποία χρησιμοποιούνται από το μοντέλο σύνθεσης φωνής TTS για εκπαίδευση. Παραδείγματος χάριν, για την πρώτη ομιλήτρια έχουμε ηχογραφήσεις συνολικής διάρκειας μισής ώρας σε στυλ εκφώνησης ειδήσεων $S_{1,news,0.5h}$, από τις οποίες δημιουργούνται συνθετικές εκφωνήσεις $S^*_{1,news,7h}$ στο ίδιο στυλ αλλά μεγαλύτερης διάρκειας 7 ωρών. Ύστερα το μοντέλο $\text{TTS}_1$ εκπαιδεύεται χρησιμοποιώντας όλες τις εκφωνήσεις, πραγματικές και συνθετικές στα δύο στυλ (neutral και news) που ανήκουν όμως μόνο στην πρώτη ομιλήτρια. Τέλος γίνεται το fine tuning με τα πραγματικά δεδομένα $S_{1,news,0.5h}$ στο στυλ (news) που επιθυμούμε. Αντίστοιχα είναι και τα βήματα για τη δεύτερη ομιλήτρια, όπου οι πραγματικές εκφωνήσεις της στο στυλ εκφώνησης ειδήσεων έχουν επίσης συνολική διάρκεια μισή ώρα. Με το μοντέλο $\text{VC}_2$ δημιουργούνται τα συνθετικά δεδομένα $S^*_{2,news,4h}$ σε στυλ ειδήσεων διάρκειας 4 ωρών, τα οποία με τη σειρά τους χρησιμοποιούνται στην εκπαίδευση του μοντέλου $\text{TTS}_2$ μαζί με τις πραγματικές εκφωνήσεις από τη δεύτερη ομιλήτρια.

Το δεύτερο πείραμα αφορά τη σύνθεση φωνής σε στυλ συζήτησης από πολλούς ομιλητές (multi speaker TTS-conversational style). Συνολικά υπάρχουν 18 ομιλητές από τους οποίους οι πρώτοι 8 είναι εκείνοι για τους οποίους παράγουμε εκφωνήσεις στο συγκεκριμένο στυλ. Τα βήματα σε αυτή την περίπτωση είναι τα εξής:

1. $\text{VC}(S_{1:8,conv,0.5h}, S_{9:15,news,5h}, S_{16:18,conv,1.5h})$
2. $\forall i \in [1,8]: \quad \text{TTS}_i(S_{i,conv,0.5h}, S_{9:18,conv,\sum}, S^*_{i,conv,5h})$
3. $\forall i \in [1,8]: \quad \text{FT}(\text{TTS}_i, S_{i,conv,0.5h})$

Σε πρώτη φάση εκπαιδεύεται ένα μοντέλο μετατροπής φωνής χρησιμοποιώντας δεδομένα από όλους τους ομιλητές και στα δύο στυλ (conv και news). Ουσιαστικά για κάθε target i-ομιλητή, $i = 1, \ldots, 8$ κατασκευάζουμε συνθετικά δεδομένα $S^*_{i,conv,5h}$ εκμεταλλευόμενοι τις πραγματικές ηχογραφήσεις $S_{x,news,5h}$ από τους ομιλητές $x = 9, \ldots, 15$. Εν συνεχεία εκπαιδεύονται οκτώ ξεχωριστά μοντέλα σύνθεσης φωνής $\text{TTS}_i$ χρησιμοποιώντας τις εκφωνήσεις που δημιουργήθηκαν στο προηγούμενο βήμα καθώς και τις πραγματικές εκφωνήσεις από όλους τους ομιλητές στο στυλ που επιθυμούμε (conversational). Τέλος γίνεται το fine tuning κάθε μοντέλου σύνθεσης φωνής $\text{TTS}_i$ μόνο στα πραγματικά δεδομένα $S_{i,conv,0.5h}$ από κάθε ομιλητή σε στυλ συζήτησης.

Για την αξιολόγηση των μοντέλων λήφθηκαν υπόψιν τα εξής: α) η ποιότητα του σήματος της παραγόμενης εκφώνησης (signal quality) β) αν το στυλ της εκφώνησης είναι το επιθυμητό (style adequacy) γ) η φυσικότητα των παραγόμενων εκφωνήσεων (naturalness) και δ) η ομοιότητα της συνθετικής εκφώνησης με αυτή του target ομιλητή (speaker similarity). Η μετρική που χρησιμοποιήθηκε για τις συγκρίσεις των μοντέλων είναι τα MUSHRA tests [ITU03]. Για το πρώτο σενάριο (single speaker TTS-newscaster style) οι συγγραφείς εξετάζουν αρχικά ένα (B) baseline μοντέλο σύνθεσης φωνής και στη συνέχεια προσθέτουν σταδιακά είτε fine tuning (B+FT), είτε ένα μοντέλο μετατροπής φωνής (B+VC), είτε και τα δύο μαζί (B+VC+FT). Τα βέλτιστα αποτελέσματα παρουσιάζονται για την τελευταία περίπτωση και για τις δύο ομιλήτριες. Συγκεκριμένα τα scores για την ποιότητα των παραγόμενων δειγμάτων και για την ομοιότητα στο στυλ εκφώνησης είναι περίπου 72.9 και 67.9 αντίστοιχα για την πρώτη ομιλήτρια, ενώ για τη δεύτερη 71.2 και 70. Συνεπώς η προσθήκη συνθετικών δεδομένων στην εκπαίδευση, τα οποία έχουν προκύψει από ένα μοντέλο μετατροπής φωνής καθώς και το fine tuning του μοντέλου σύνθεσης φωνής στα πραγματικά δεδομένα παράγουν δείγματα υψηλότερης ποιότητας στο επιθυμητό στυλ. Στο δεύτερο σενάριο (multi speaker TTS-conversational style) εξετάζεται πως επηρεάζει η συνολική διάρκεια των εκφωνήσεων από έναν ομιλητή, τη φυσικότητα και την ομοιότητα μεταξύ

των πραγματικών και των συνθετικών εκφωνήσεων που παράγονται για αυτόν. Συγκεκριμένα η ελάχιστη συνολική διάρκεια εκφωνήσεων που χρησιμοποιείται για την εκπαίδευση των μοντέλων είναι μισή ώρα για κάθε ομιλητή. Επίσης ακολουθούνται δύο προσεγγίσεις. Η πρώτη αφορά την εκπαίδευση ενός μοντέλου σε όλα τα διαθέσιμα δεδομένα (non DR-non data reduced) από όλους τους ομιλητές, ενώ η δεύτερη αφορά την εκπαίδευση με χρήση των εκφωνήσεων συνολικής διάρκειας μισής ώρας αλλά με τη τεχνική της μετατροπής φωνής και του fine tuning (DR-data reduction+VC+FT). Τα αποτελέσματα δείχνουν ότι με τη δεύτερη προσέγγιση παράγονται πιο φυσικές εκφωνήσεις για τους target ομιλητές, οι οποίες είναι και αρκετά όμοιες με τις πραγματικές τους εκφωνήσεις. Λαμβάνοντας λοιπόν υπόψιν τα αποτελέσματα από τα πειράματα, είναι εμφανές ότι η δημιουργία συνθετικών δεδομένων από ένα μοντέλο μετατροπής φωνής και η χρήση τους για την εκπαίδευση ενός μοντέλου σύνθεσης φωνής μαζί με fine tuning στα πραγματικά δεδομένα, μπορεί να βοηθήσει αρκετά στη δημιουργία εκφωνήσεων από έναν συγκεκριμένο ομιλητή και στο στυλ που επιθυμούμε, ακόμα και στην περίπτωση που διαθέτουμε λίγες εκφωνήσεις από αυτόν. Στη συνέχεια εξετάζουμε την περίπτωση όπου θέλουμε να συνθέσουμε φωνή σε μια γλώσσα για την οποία υπάρχουν λίγα διαθέσιμα δεδομένα.

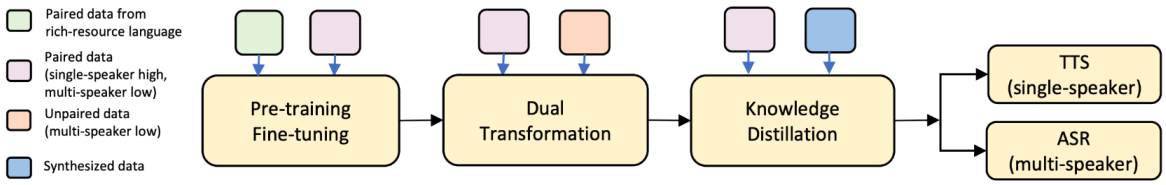
## 3.3 LRSpeech - Low Resource TTS

Όπως έχουμε δει μέχρι τώρα, τα συστήματα σύνθεσης φωνής από κείμενο απαιτούν μεγάλο πλήθος δεδομένων (text, audio) κατά την εκπαίδευση προκειμένου η παραγόμενη φωνή να έχει υψηλή ποιότητα. Αντίστοιχα, ένα σύστημα αναγνώρισης φωνής (ASR) απαιτεί αρκετές ώρες δεδομένων ομιλίας μαζί με τα transcriptions ώστε να μπορεί να αναγνωρίζει με μεγάλη ακρίβεια το κείμενο που εκφωνείται. Συνήθως τα συστήματα text-to-speech εκπαιδεύονται πάνω σε εκφωνήσεις υψηλής ποιότητας από έναν ομιλητή, ενώ για την αναγνώριση φωνής οι εκφωνήσεις μπορούν να προέρχονται από περισσότερους ομιλητές. Προφανώς η συλλογή των απαιτούμενων δεδομένων ήχου-κειμένου έχει κόστος και μπορεί να είναι αρκετά χρονοβόρα (π.χ. ηχογράφηση). Το πρόβλημα γίνεται ακόμη πιο σύνθετο όταν θέλουμε να κατασκευάσουμε ένα τέτοιο σύστημα σε γλώσσες όπως τα ελληνικά, για τις οποίες δεν διαθέτουμε τόσα πολλά δεδομένα, τα οποία να είναι και υψηλής ποιότητας. Γενικά υπάρχουν πάνω από 6000 γλώσσες στον κόσμο και μόνο για μερικές δεκάδες από αυτές υποστηρίζονται συστήματα σύνθεσης ή αναγνώρισης φωνής. Για το λόγο αυτό ερευνητές της Microsoft προτείνουν τη μέθοδο LRSpeech (low resource speech) [Xu+20] για την κατασκευή συστημάτων σύνθεσης και αναγνώρισης φωνής σε γλώσσες όπου τα διαθέσιμα δεδομένα είναι ελάχιστα και δεν επαρκούν για την εκπαίδευση. Η μέθοδος αυτή στηρίζεται σε τρία βασικά βήματα. Σε πρώτο στάδιο γίνεται εκπαίδευση των μοντέλων TTS και ASR πάνω σε γλώσσες όπου υπάρχουν πολλά δεδομένα εκπαίδευσης (rich-resource languages), όπως είναι τα αγγλικά. Στη συνέχεια δημιουργούνται συνθετικά δείγματα από τα εκπαιδευμένα μοντέλα, τα οποία αξιοποιούνται για την περαιτέρω εκπαίδευση και βελτίωσή τους. Τέλος για ακόμη μεγαλύτερη ακρίβεια και ποιότητα των δύο συστημάτων, πέρα από τα συνθετικά δείγματα χρησιμοποιούνται και μη ζευγαρωμένα (unpaired) δείγματα είτε κειμένου για την εκπαίδευση του συστήματος σύνθεσης φωνής, είτε ομιλίας για την εκπαίδευση του συστήματος αναγνώριση φωνής. Όπως θα δούμε στη συνέχεια, η συγκεκριμένη μέθοδος παράγει ικανοποιητικά αποτελέσματα τόσο στη σύνθεση όσο και στην αναγνώριση φωνής για γλώσσες με λίγα διαθέσιμα δεδομένα.

### 3.3.1 Μεθοδολογία

Για τα μοντέλα σύνθεσης φωνής από κείμενο συνήθως απαιτούνται αρκετές ώρες ηχογραφήσεων από έναν συγκεκριμένο ομιλητή (π.χ. το LJSpeech με 24 ώρες ηχογραφήσεων από μια ομιλήτρια). Επίσης για την αναγνώριση φωνής δεν είναι απαραίτητο οι ηχογραφήσεις να προέρχονται από έναν ομιλητή αλλά χρειάζονται εκατοντάδες ώρες δεδομένων ομιλίας κατά την εκπαίδευση. Το LibriSpeech [Pan+15] είναι ένα σύνολο δεδομένων για εκπαίδευση συστημάτων αναγνώρισης φωνής που περιέχει περίπου 1000 ώρες ηχογραφήσεων στην αγγλική γλώσσα από πολλούς ομιλητές. Τα low-resource συστήματα στοχεύουν στην ανάπτυξη τεχνικών και μοντέλων που μειώνουν το απαιτούμενο πλήθος δεδομένων αλλά διατηρούν την ποιότητα των αποτελεσμάτων. Στα low resource μοντέλα TTS οι ώρες ηχογραφήσεων που απαιτούνται για εκπαίδευση μπορούν να μειωθούν σε μερικές δεκάδες λεπτά, ενώ στα μοντέλα ASR μπορούν να μειωθούν σε μερικές δεκάδες ώρες. Το LRSpeech στοχεύει σε περαιτέρω μείωση των απαιτούμενων δεδομένων (extremely low-resource) αλλά και την αξιοποίηση unpaired δεδομένων (είτε μεμονωμένων audio είτε μεμονωμένων text). Συγκεκριμένα, όπως θα δούμε και στα πειράματα χρησιμοποιεί μόλις μερικά λεπτά υψηλής ποιότητας ηχογραφήσεων μαζί με τα αντίστοιχα κείμενα από έναν ομιλητή, μερικές ώρες με ελαφρώς χαμηλότερη ποιότητα καθώς και τα κείμενα από πολλούς ομιλητές και τέλος μερικές δεκάδες ώρες μεμονωμένων ηχογραφήσεων (χωρίς τα κείμενα) από διάφορους ομιλητές. Προφανώς με αυτή τη μέθοδο μειώνεται σημαντικά το κόστος συλλογής των δεδομένων αν σκεφτούμε ότι χρειάζονται μόλις μερικά λεπτά από ηχογραφήσεις υψηλής ποιότητας από έναν ομιλητή ενώ επίσης μπορούν εύκολα να βρεθούν ηχογραφήσεις με λίγο χαμηλότερη ποιότητα από πολλούς ομιλητές (π.χ. από το διαδίκτυο).

Στο Σχήμα 3.4 φαίνονται τα τρία βασικά βήματα που ακολουθεί η μέθοδος LRSpeech. Αρχικά εκπαιδεύονται δύο ξεχωριστά μοντέλα TTS και ASR σε κάποια γλώσσα με επαρκή δεδομένα ήχου-κειμένου. Η ιδέα είναι ότι αφού υπάρχουν κοινά σημεία μεταξύ των γλωσσών μπορεί να αξιοποιηθεί η μεταφορά μάθησης (transfer learning) προκειμένου να λάβουμε καλά αποτελέσματα σε μια γλώσσα με πολύ λιγότερα δεδομένα. Έπειτα ακολουθεί το fine-tuning των δύο μοντέλων. Συγκεκριμένα χρησιμοποιούνται ηχογραφήσεις υψηλής ποιότητας από έναν ομιλητή αλλά και ηχογραφήσεις με ελαφρώς χαμηλότερη ποιότητα από πολλούς ομιλητές καθώς και τα αντίστοιχα κείμενα από τη low-resource γλώσσα. Κατά το fine-tuning λαμβάνεται υπόψιν η διαφορά στους χαρακτήρες μεταξύ των δύο γλωσσών καθώς και η ταυτότητα των ομιλητών, οπότε οι παράμετροι που έχουν προκύψει από το pre-training αφορούν όλο το υπόλοιπο μοντέλο πλην των embeddings που αντιστοιχούν στους χαρακτήρες και στους ομιλητές. Επομένως σε πρώτη φάση εκπαιδεύονται ξεχωριστά τα συγκεκριμένα embeddings και έπειτα γίνεται fine-tuning σε ολόκληρο το μοντέλο.




Σχήμα 3.4: Τα τρία βήματα της μεθόδου LRSpeech. [Xu+20]

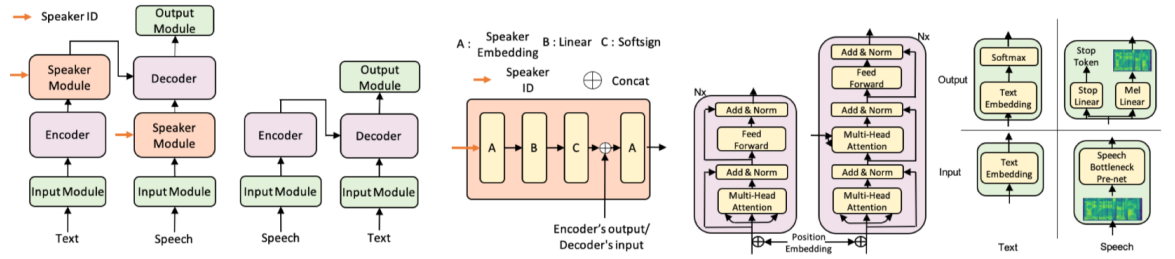
Στο δεύτερο βήμα (dual transformation) αξιοποιείται η συμπληρωματικότητα των δύο task TTS και ASR, ώστε καθένα να επωφεληθεί από τα αποτελέσματα του άλλου. Συγκεκριμένα ένα οποιοδήποτε κείμενο μπορεί να μετατραπεί μέσω του μοντέλου TTS σε φωνή. Κατ' αυτόν τον

τρόπο δημιουργείται ένα τεχνητό dataset που περιέχει ζεύγη κειμένου και συνθετικής φωνής. Τα ζεύγη αυτά χρησιμοποιούνται για την εκπαίδευση του μοντέλου ASR. Στο μοντέλο TTS μπορεί να επιλεγεί τυχαία η ταυτότητα ενός ομιλητή ώστε οι εκφωνήσεις που θα προκύψουν να προέρχονται από πολλούς ομιλητές. Αντίστοιχα μια ηχογράφηση μετατρέπεται σε ένα transcription μέσω του μοντέλου αναγνώρισης φωνής, οπότε προκύπτουν νέα ζεύγη κειμένου-φωνής ώστε να εκπαιδευτεί το μοντέλο σύνθεσης φωνής. Έτσι η μέθοδος LRSpeech εκμεταλλεύεται την ύπαρξη μεμονωμένων κειμένων ή ηχογραφήσεων για να παράγει στην πραγματικότητα νέα τεχνητά δεδομένα.

Το τρίτο βήμα (knowledge distillation) της μεθόδου πραγματοποιεί περαιτέρω βελτίωση των μοντέλων TTS και ASR όσον αφορά την ακρίβεια και την ποιότητα των εκφωνήσεων. Το TTS μοντέλο παράγει συνθετικές εκφωνήσεις από τον target ομιλητή χρησιμοποιώντας unpaired κείμενα. Οι εκφωνήσεις αυτές φιλτράρονται ώστε να γίνει διόρθωση όταν παραλείπεται ή επαναλαμβάνεται κάποια λέξη σε αυτές. Το μοντέλο TTS εκπαιδεύεται εκ νέου στο καινούριο φιλτραρισμένο dataset που προκύπτει για να παράγει εκφωνήσεις από τον target ομιλητή.

### 3.3.2 Αρχιτεκτονική των μοντέλων TTS και ASR

Η αρχιτεκτονική των συστημάτων TTS και ASR βασίζεται στο μοντέλο Transformer. Όπως φαίνεται στο Σχήμα 3.5 το βασικό στοιχείο της αρχιτεκτονικής τους είναι ο encoder και ο decoder. Επίσης και στα δύο μοντέλα χρησιμοποιούνται δύο επιπλέον modules, το input και το output module. Το μοντέλο TTS περιλαμβάνει επίσης και ένα speaker module που δέχεται ως είσοδο το speaker id για την περίπτωση όπου έχουμε εκφωνήσεις από πολλούς ομιλητές. Για το TTS μοντέλο




(a) TTS model (b) ASR model (c) Speaker module (d) Encoder (left) and Decoder (right) (e) Input/Output module for speech/text

Σχήμα 3.5: Αρχιτεκτονική των μοντέλων TTS και ASR της μεθόδου LRSpeech. [Xu+20]

το input module που βρίσκεται πριν τον encoder, υπολογίζει ένα embedding από την ακολουθία των χαρακτήρων του κειμένου, ενώ το input module πριν τον decoder δέχεται τα frames από το φασματογράφημα σε κλίμακα mel και τα μετασχηματίζει μέσω ορισμένων fully connected επιπέδων. Το output module περιλαμβάνει ένα fully connected επίπεδο (mel linear), το οποίο μετατρέπει την έξοδο του decoder σε frame του φασματογραφήματος στην κλίμακα mel. Επίσης ένα απλό δίκτυο (stop linear) με σιγμοειδή συνάρτηση ενεργοποίησης δίνει την πιθανότητα τερματισμού προκειμένου ο decoder να μην εξάγει άλλα frames από το φασματογράφημα κατά το inference. Το speaker module δέχεται όπως είπαμε την ταυτότητα ενός ομιλητή και παράγει το speaker embedding. Το πρώτο speaker module λαμβάνει υπόψιν την έξοδο του encoder ενώ το δεύτερο την είσοδο πριν τον decoder, όπως φαίνεται στο Σχήμα 3.5 (c) (concatenation). Το μοντέλο ASR έχει παρόμοια δομή με το μοντέλο TTS πέραν των speaker modules. Το input module πριν τον encoder δέχεται τα frames του φασματογραφήματος και μειώνει τη διάστασή τους μέσω ορισμένων συνελικτικών επιπέδων. Πριν τον decoder το input module παράγει embedding από το κείμενο.

Τέλος το output module είναι ένα fully connected επίπεδο με συνάρτηση ενεργοποίησης softmax για την πρόβλεψη του σωστού χαρακτήρα για το εκάστοτε frame.

### 3.3.3 Πειράματα - Αποτελέσματα

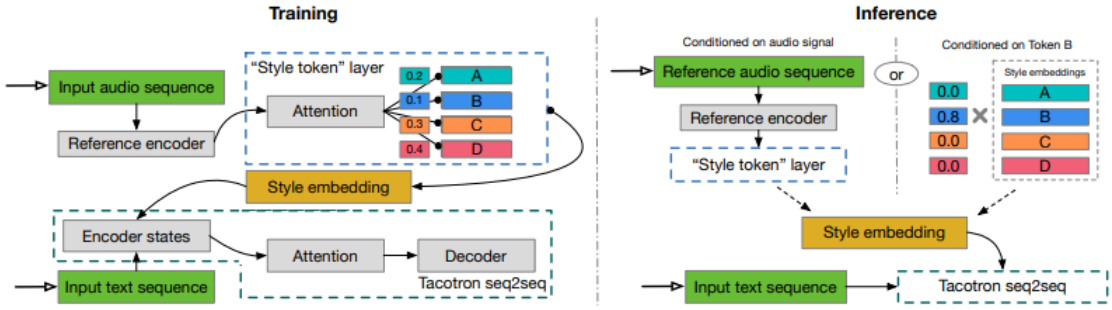
Η εκπαίδευση των μοντέλων TTS και ASR έγινε με χρήση δεδομένων από την κινέζικη (Mandarin Chinese) ως rich-resource γλώσσα και την αγγλική γλώσσα ως low-resource γλώσσα. Συγκεκριμένα για τα κινέζικα χρησιμοποιήθηκαν συνολικά 12 ώρες από έναν ομιλητή για το μοντέλο TTS ενώ περίπου 178 ώρες από 400 ομιλητές για το μοντέλο ASR. Για τα αγγλικά χρησιμοποιήθηκαν μόλις 5 λεπτά από έναν ομιλητή για το μοντέλο TTS, ενώ περίπου 3.5 ώρες από πολλούς ομιλητές με ελαφρώς χαμηλότερη ποιότητα για το μοντέλο ASR. Για την αξιολόγηση των μοντέλων σύνθεσης φωνής εφαρμόστηκαν οι μετρικές του MOS και IR (inteligibility rate). Η μετρική IR εκφράζει το πόσο κατανοητό είναι το αποτέλεσμα της συνθετικής φωνής και υπολογίζεται ως εξής: οι βαθμολογητές ακούν ορισμένες προτάσεις και σε κάθε μια από αυτές σημειώνουν τις λέξεις που δεν ήταν κατανοητές. Το IR προκύπτει ως το ποσοστό των λέξεων που ήταν κατανοητές σε σχέση με το συνολικό αριθμό λέξεων στις προτάσεις που κλήθηκαν να ακούσουν. Ως vocoder του συστήματος TTS χρησιμοποιήθηκε το Parallel WaveGAN [YSK20]. Στο συγκεκριμένο πείραμα η μέθοδος LRSpeech λαμβάνει την τιμή 98.08% για την μετρική IR, ενώ το MOS την τιμή 3.57 (για τις πραγματικές εκφωνήσεις 4.05), η οποία είναι ικανοποιητική αν λάβουμε υπόψιν το ελάχιστο πλήθος δεδομένων που χρησιμοποιήθηκαν από τη low resource γλώσσα. Επιπλέον η τεχνική αυτή εφαρμόζεται και στα Λιθουανικά με πολύ λίγα δεδομένα και πετυχαίνει MOS με τιμή 3.65, η οποία είναι επίσης ικανοποιητική αν λάβουμε υπόψιν ότι το MOS για τις πραγματικές εκφωνήσεις είναι 4.01. Συμπεραίνουμε λοιπόν ότι η μέθοδος LRSpeech μπορεί να αξιοποιήσει γλώσσες με μεγάλο αριθμό δεδομένων έτσι ώστε συνδυάζοντας τα συμπληρωματικά tasks TTS και ASR, να παράγει πολύ καλά αποτελέσματα σε γλώσσες όπου υπάρχουν ελάχιστα δεδομένα.

## 3.4 Global Style Tokens

Μέχρι στιγμής έχουμε εξετάσει ορισμένα state of the art μοντέλα για τη σύνθεση φωνής από κείμενο τα οποία όπως είδαμε παράγουν ομιλία με μεγάλη φυσικότητα. Πέραν της φυσικότητας, η προσωδία είναι ένα επιπλέον χαρακτηριστικό που θέλουμε να έχει η παραγόμενη φωνή. Η προσωδία αντικατοπτρίζει σημαντικά στοιχεία της ομιλίας όπως ο ρυθμός, ο τόνος και ο επιτονισμός της. Με άλλα λόγια πέρα από τη φυσικότητα μιας εκφώνησης μας ενδιαφέρει και ο τρόπος με τον οποίο εκφωνείται, δηλαδή οι παύσεις και ο τονισμός της, η συναισθηματική κατάσταση κ.ά. Ένα από τα μοντέλα που στοχεύει στη μοντελοποίηση του στυλ και της προσωδίας της παραγόμενης φωνής είναι το Global Style Tokens (GST) [Wan+18]. Για να το επιτύχει χρησιμοποιεί ένα σύνολο από embeddings (style tokens), τα οποία είναι υπεύθυνα για τον τρόπο σύνθεσης ομιλίας αφού χρησιμοποιούνται για τον έλεγχο χαρακτηριστικών στοιχείων όπως η ταχύτητα, το στυλ και το συναίσθημα. Στη συνέχεια παρουσιάζουμε τον τρόπο λειτουργίας του μοντέλου καθώς και ορισμένα αποτελέσματα σε πειράματα όπως ο έλεγχος (style control) και η μεταφορά στυλ (style transfer) στη σύνθεση φωνής από κείμενο.

### 3.4.1 Αρχιτεκτονική

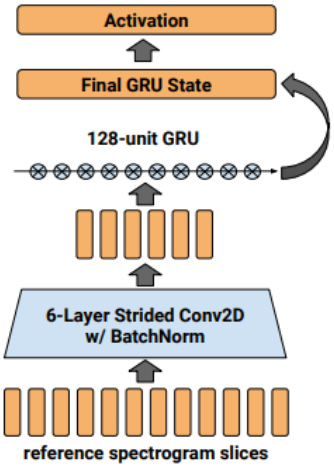
Το μοντέλο Global Style Tokens χρησιμοποιεί ως βάση το μοντέλο Tacotron [Wan+17] προκειμένου να μετατρέψει ένα κείμενο στο αντίστοιχο φασματογράφημα στην κλίμακα mel. Το Tacotron είναι ένα μοντέλο της μορφής sequence-to-sequence με μηχανισμό προσοχής. Για την περίπτωση του μοντέλου GST χρησιμοποιούνται ορισμένες μικρές παραλλαγές όσον αφορά τη δομή του Tacotron που έχουμε δει σε προηγούμενο κεφάλαιο. Αρχικά ως είσοδος δίνεται η ακολουθία των φωνημάτων αντί της ακολουθίας των χαρακτήρων του κειμένου. Επίσης ο decoder σε κάθε βήμα εξάγει δύο frames από το φασματογράφημα χρησιμοποιώντας δύο LSTM αντί για GRU cells. Για την παραγωγή φωνής χρησιμοποιείται ο αλγόριθμος Griffin-Lim που δέχεται ως είσοδο το φασματογράφημα σε γραμμική κλίμακα. Το γραμμικό φασματογράφημα προκύπτει από το φασματογράφημα στην κλίμακα mel μέσω ενός συνελικτικού δικτύου με dilations. Η τιμή του MOS για το νέο Tacotron είναι περίπου 4.0 το οποίο είναι καλύτερο σε σχέση με την τιμή 3.82 από την αρχική εκδοχή του. Στο Σχήμα 3.6 φαίνεται η λειτουργία του μοντέλου GST σε συνδυασμό με το Tacotron τόσο κατά την εκπαίδευση όσο και τη συμπερασματολογία. Όσον αφορά τη δομή




Σχήμα 3.6: Εκπαίδευση και συμπερασματολογία στο μοντέλο Global Style Tokens. [Wan+18]

του μοντέλου, παρατηρούμε ότι αποτελείται από τον reference encoder, το style token layer και το Tacotron. Ο reference encoder [Ske+18] είναι υπεύθυνος για την παραγωγή ενός διανύσματος σταθερού μήκους (reference embedding) που αντικατοπτρίζει την προσωδία σε ένα ηχητικό σήμα. Η αρχιτεκτονική του φαίνεται στο Σχήμα 3.7. Ως είσοδος δίνεται το φασματογράφημα στην κλίμακα mel το οποίο εξάγεται από μια κυματομορφή (input-reference audio sequence). Κατά την εκπαίδευση χρησιμοποιείται η πραγματική κυματομορφή που αντιστοιχεί στο κείμενο εισόδου (άρα και το αντίστοιχο ground truth mel-spectrogram), ενώ κατά τη συμπερασματολογία μπορεί να χρησιμοποιηθεί ένα οποιοδήποτε ηχητικό σήμα του οποίου θέλουμε να μοντελοποιήσουμε τον τρόπο εκφώνησης, όπως θα δούμε και στη συνέχεια (style transfer). Από το Σχήμα 3.7 βλέπουμε ότι για την εξαγωγή του reference embedding χρησιμοποιούνται αρχικά 6-(2D) συνελικτικά επίπεδα με κανάλια εξόδου 32, 32, 64, 64, 128, 128 αντίστοιχα, μέγεθος kernel $3 \times 3$, stride $2 \times 2$, same padding, συνάρτηση ενεργοποίησης ReLU καθώς και Batch normalization. Επομένως αν ένα φασματογράφημα έχει αρχικά διάσταση (Frames, n_mels, 1) ύστερα από το τελευταίο συνελικτικό επίπεδο η διάσταση θα είναι (Frames/64, n_mels/64, 128), αφού σε κάθε layer η διάσταση μειώνεται στο μισό. Στη συνέχεια ακολουθεί ένα δίκτυο GRU με διάσταση εξόδου 128. Σε κάθε timestep δίνεται ως είσοδος ένα διάνυσμα διάστασης $(128 \cdot \text{n\_mels}/64)$. Η έξοδος από το τελευταίο GRU cell, διέρχεται από ένα fully connected επίπεδο με συνάρτηση ενεργοποίησης tanh ώστε η τελική έξοδος να έχει την επιθυμητή διάσταση. Το διάνυσμα που προκύπτει είναι το reference embedding.

Έπειτα ακολουθεί το style token layer που δέχεται το reference embedding και παράγει ένα




Σχήμα 3.7: Αρχιτεκτονική του reference encoder στο μοντέλο GST. [Ske+18]

διάνυσμα style embedding, το οποίο στη συνέχεια προστίθεται στα encoder states του Tacotron. Όπως φαίνεται και στο Σχήμα 3.6 αριστερά, το style token layer χρησιμοποιεί ένα μηχανισμό προσοχής που δέχεται ως query το reference embedding και υπολογίζει ένα διάνυσμα από scores για κάθε token embedding (στο σχήμα είναι τα A, B, C, D). Σαν μηχανισμός προσοχής χρησιμοποιείται το content-based attention [GWD14] (εναλλακτικά μπορεί να χρησιμοποιηθεί το multi-head attention [Vas+17]), μέσω του οποίου υπολογίζεται ένα μέτρο ομοιότητας μεταξύ του reference embedding και κάθε style token. Επομένως αν $\mathbf{r}$ είναι το reference embedding και $\mathbf{s}_1, \ldots, \mathbf{s}_n$ τα style tokens, τότε τα scores για κάθε token υπολογίζονται ως εξής:

$$w_i = \text{softmax}[\cos(\mathbf{r}, \mathbf{s}_i)], \quad \text{όπου } \cos(\mathbf{r}, \mathbf{s}_i) = \frac{\mathbf{r} \cdot \mathbf{s}_i}{\|\mathbf{r}\| \cdot \|\mathbf{s}_i\|}, \quad i = 1, \ldots, n. \tag{3.4.1}$$

Τα token embeddings αρχικοποιούνται με τυχαίο τρόπο και έχουν διάσταση 256 για να συμπίπτει με τη διάσταση των states του Tacotron encoder. Στο μοντέλο GST επιλέγονται $n = 10$ tokens για να μοντελοποιήσουν το στυλ της παραγόμενης φωνής. Λαμβάνοντας το σταθμισμένο άθροισμα από όλα τα style tokens με βάρη τα scores από το attention module, προκύπτει τελικά το διάνυσμα style embedding. Έπειτα το style embedding επαναλαμβάνεται τόσες φορές όσες και το πλήθος των encoder states του Tacotron και προστίθεται σε κάθε ένα από αυτά. Τα νέα αυτά διανύσματα χρησιμοποιούνται από τον decoder ώστε σε κάθε βήμα να εξάγονται δύο frames από το φασματογράφημα. Κατά το inference (Σχήμα 3.6 δεξιά) υπάρχουν δύο επιλογές. Αρχικά ως είσοδος στον reference encoder μπορεί να δοθεί ένα reference ηχητικό σήμα (δηλαδή το αντίστοιχο φασματογράφημα) προκειμένου να κάνουμε μεταφορά του στυλ και της προσωδίας από το συγκεκριμένο δείγμα. Έτσι η φωνή που παράγεται από το μοντέλο Tacotron θα έχει χαρακτηριστικά που προκύπτουν από το reference audio. Εναλλακτικά μπορεί να γίνει κατευθείαν επιλογή ενός μόνο style token ή ακόμα και των scores για κάθε token. Με αυτόν τον τρόπο ελέγχεται άμεσα το στυλ της παραγόμενης εκφώνησης από τα tokens χωρίς να χρειάζεται ένα reference ηχητικό δείγμα.

### 3.4.2 Πειράματα

Στην ενότητα αυτή περιγράφουμε ορισμένα πειράματα από το μοντέλο GST για τον έλεγχο και τη μεταφορά του στυλ (style control/transfer). Για την εκπαίδευση των μοντέλων χρησιμοποιήθηκε

ένα σύνολο δεδομένων στην αγγλική γλώσσα με 147 ώρες ηχογραφήσεων από ένα audio-book. Όσον αφορά τον έλεγχο του στυλ μιας παραγόμενης φωνής έγιναν τα εξής πειράματα: style selection, style scaling, style sampling και text-side style control-morhping. Για το style selection κατά το inference αντί για το διάνυσμα style embedding που προκύπτει από το style token layer, μπορεί εναλλακτικά να γίνει χρήση ενός συγκεκριμένου token. Με αυτόν τον τρόπο βλέπουμε τι χαρακτηριστικό της φωνής αντιπροσωπεύει κάθε style token (π.χ. τόνος, ταχύτητα, συναίσθημα), ώστε να μπορούμε να ελέγξουμε την προσωδία της επιλέγοντας το επιθυμητό token. Για το style scaling επιλέγεται πάλι ένα style token αλλά αυτή τη φορά πολλαπλασιάζεται με μια τιμή. Για παράδειγμα αν ένα token εκφράζει την ταχύτητα στην ομιλία τότε αν αυξήσουμε την τιμή που κάνουμε scale στο συγκεκριμένο token, η παραγόμενη φωνή θα εκφωνείται σε μικρότερο χρόνο άρα με μεγαλύτερη ταχύτητα. Αντιθέτως αν χρησιμοποιήσουμε κάποια μικρότερη ή αρνητική τιμή τότε η ταχύτητα εκφώνησης μειώνεται. Για το style sampling αντί για ένα συγκεκριμένο token επιλέγονται τα scores σύμφωνα με τα οποία πολλαπλασιάζονται τα token embeddings. Έτσι μπορεί να γίνει μια σύνθεση των επιθυμητών χαρακτηριστικών της φωνής, όπου ο βαθμός επιρροής κάθε χαρακτηριστικού εξαρτάται από την τιμή του εκάστοτε score που επιλέγουμε. Τέλος για τον έλεγχο του στυλ από την πλευρά του κειμένου (text-side style morphing) ακολουθείται η εξής μέθοδος. Αντί σε κάθε state του Tacotron encoder να προσθέτουμε ένα μόνο διάνυσμα (είτε το style embedding είτε ένα επιθυμητό token), μπορούμε να αλλάζουμε τα tokens που προσθέτουμε σύμφωνα με τη σειρά των states. Για παράδειγμα αν έχουμε 50 encoder states μπορούμε στα πρώτα 25 να προσθέσουμε ένα token που αντιπροσωπεύει υψηλή ταχύτητα ομιλίας και στα υπόλοιπα 25 ένα token που αντιπροσωπεύει κάποιο τόνο ή ύφος. Σε αυτή την περίπτωση η παραγόμενη φωνή αρχικά θα ακούγεται γρήγορα και στη συνέχεια ο τόνος της θα αλλάξει ώστε να συμπίπτει με τον τόνο που προσδίδει η επιλογή του δεύτερου token.

Όσον αφορά το style transfer κατά το inference γίνεται μεταφορά του στυλ και της προσωδίας από ένα reference ηχητικό δείγμα. Επομένως θέλουμε να παράγουμε φωνή από ένα κείμενο, της οποίας τα χαρακτηριστικά θα προέρχονται από ένα επιθυμητό ηχητικό δείγμα. Κατ' αυτόν τον τρόπο τα scores των token embeddings υπολογίζονται από το reference embedding και δεν ορίζονται εξαρχής όπως στην περίπτωση του style control. Για τα πειράματα του style transfer ακολουθήθηκαν δύο προσεγγίσεις. Σύμφωνα με την πρώτη (parallel style transfer), το κείμενο του reference audio αντιστοιχεί στο ίδιο κείμενο από το οποίο θέλουμε να συνθέσουμε φωνή. Κατά το non-parallel style transfer μοντελοποιούμε το στυλ από ένα οποιοδήποτε reference audio με κείμενο διαφορετικό αυτού που καλούμαστε να μετατρέψουμε σε φωνή. Τα αποτελέσματα από τα πειράματα για το non-parallel style transfer δείχνουν ότι οι εκφωνήσεις που προκύπτουν από το μοντέλο GST δίνουν υψηλότερο MOS σε σχέση με εκφωνήσεις από ένα απλό Tacotron. Τα ηχητικά δείγματα από τα πειράματα που αναφέρουμε βρίσκονται στη σελίδα GST audio samples.

Λαμβάνοντας υπόψιν τα παραπάνω συμπεραίνουμε ότι το μοντέλο GST μπορεί να μοντελοποιήσει το στυλ και την προσωδία σε ένα σύστημα παραγωγής ομιλίας μέσω ενός συνόλου από token embeddings. Τα embeddings αυτά αντιπροσωπεύουν σημαντικά στοιχεία της ομιλίας, όπως ο τόνος, το συναίσθημα, η ταχύτητα κά. Επιπλέον μπορούν να χρησιμοποιηθούν κατά τη συμπερασματολογία προκειμένου η συνθετική φωνή να έχει τα επιθυμητά χαρακτηριστικά, που ρυθμίζονται είτε από ένα reference audio είτε απευθείας από τα style tokens.

## 3.5 Αξιολόγηση συστημάτων σύνθεσης φωνής από κείμενο

Η αξιολόγηση ενός συστήματος σύνθεσης φωνής από κείμενο (text-to-speech) γίνεται κατά βάσει με τη βαθμολόγηση μεμονωμένων προτάσεων. Στην πλειοψηφία των περιπτώσεων τέτοια συστήματα εκπαιδεύονται πάνω σε ζεύγη κειμένου-ήχου. Για να μετρήσουμε το πόσο φυσική είναι η παραγόμενη ομιλία χρησιμοποιούμε διάφορες μετρικές, επικρατέστερη των οποίων είναι το MOS (Mean Opinion Score) [SWH16]. Σύμφωνα με τη μετρική αυτή ένας βαθμολογητής αρχικά ακούει τη συνθετική φωνή και ύστερα δίνει μια βαθμολογία στην κλίμακα 1 έως 5, όπου υψηλότερες τιμές υποδηλώνουν ότι η φωνή που παράγει το σύστημα ακούγεται πιο φυσικά και μοιάζει με ανθρώπινη ομιλία. Σαφώς η μετρική αυτή σε ένα βαθμό βασίζεται στην υποκειμενικότητα κάθε βαθμολογητή, αφού για παράδειγμα δύο άνθρωποι μπορεί να αντιλαμβάνονται διαφορετικά την ποιότητα μιας συνθετικής φωνής και συνεπώς να τη βαθμολογούν διαφορετικά. Παρ' όλ' αυτά αναμένεται οι αξιολογήσεις τους στην κλίμακα MOS να μη διαφέρουν κατά πολύ μεταξύ τους.

Το κύριο ζήτημα όμως που προκύπτει σχετικά με την αξιολόγηση τέτοιων συστημάτων, είναι ότι η βαθμολόγηση εκφωνήσεων μεμονωμένων προτάσεων τις περισσότερες φορές δεν είναι κατάλληλη για να αναδείξει την καταλληλότητα ενός συστήματος παραγωγής ανθρώπινης φωνής. Μια πρόταση συνήθως αποτελεί μέρος μιας παραγράφου σε ένα κείμενο. Βαθμολογώντας λοιπόν μεμονωμένες προτάσεις είναι πιθανό να μη λάβουμε υπόψιν τη ροή και το περιεχόμενο της παραγράφου. Για παράδειγμα η πρόταση «Σήμερα έκανα έναν περίπατο στο πάρκο» μπορεί να ειπωθεί και να ερμηνευτεί με διαφορετικό τρόπο αν είναι μέρος των προτάσεων «Σήμερα έκανα έναν περίπατο στο πάρκο. Έβρεχε όμως πολύ, οπότε δεν το ευχαριστήθηκα όσο θα ήθελα». Στην πρώτη περίπτωση ένα σύστημα μπορεί να συνθέσει την πρόταση με έναν πιο ευδιάθετο τόνο, ενώ στην πραγματικότητα θα έπρεπε να τη συνθέσει με έναν τόνο μέτριας διάθεσης λαμβάνοντας υπόψιν και τη δεύτερη πρόταση. Υπάρχει λοιπόν ανάγκη εύρεσης εναλλακτικών τεχνικών για την αξιολόγηση συστημάτων που παράγουν ομιλία όχι μόνο από μεμονωμένες προτάσεις, αλλά και από παραγράφους ή ολόκληρα κείμενα.

Η εργασία με τίτλο *"Evaluating Long-form Text-to-Speech: Comparing the Ratings of Sentences and Paragraphs"* [Cla+19] επεκτείνει την αξιολόγηση συστημάτων text to speech συνδυάζοντας τρεις τρόπους. Ο πρώτος αφορά τη συνήθη αξιολόγηση μεμονωμένων προτάσεων, ο δεύτερος την αξιολόγηση ολόκληρων παραγράφων και ο τρίτος αφορά μια ενδιάμεση προσέγγιση των δύο πρώτων. Στη συνέχεια παρουσιάζουμε τις βασικές ιδέες πίσω από κάθε τρόπο αξιολόγησης.

### Αξιολόγηση μεμονωμένων προτάσεων

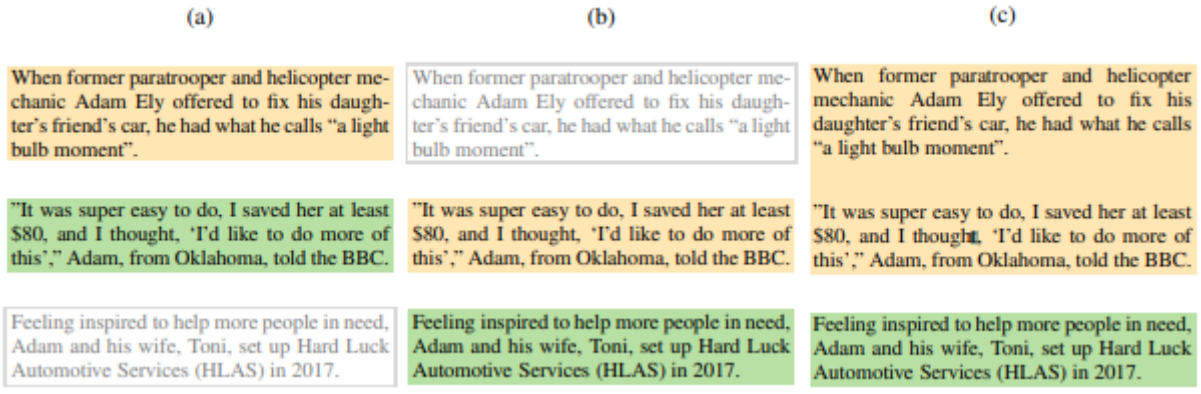
Ο πιο συνηθισμένος τρόπος για την αξιολόγηση ενός συστήματος σύνθεσης φωνής από κείμενο είναι η αξιολόγηση της παραγόμενης φωνής που αντιστοιχεί σε μεμονωμένες προτάσεις. Αυτό συμβαίνει γιατί τα περισσότερα σύνολα δεδομένων (π.χ. LJ Speech [IJ17a]), στα οποία εκπαιδεύονται τέτοιου είδους συστήματα αποτελούνται από ηχογραφήσεις μεμονωμένων προτάσεων ή μικρών φράσεων. Όπως αναφέραμε αυτή η προσέγγιση έχει το μειονέκτημα ότι δε λαμβάνεται υπόψιν η ροή και το περιεχόμενο (context) της πρότασης, όταν αυτή ανήκει σε μια παράγραφο. Έτσι ένα μοντέλο μπορεί να συνθέσει με διαφορετικό τρόπο το κείμενο που αντιστοιχεί σε μια πρόταση όταν αυτή είναι μεμονωμένη ή όταν βρίσκεται σε μια παράγραφο.

### Αξιολόγηση ολόκληρης παραγράφου

Ένας εναλλακτικός τρόπος που προτείνεται είναι η αξιολόγηση της παραγόμενης φωνής που αντιστοιχεί σε μια ολόκληρη παράγραφο ή ένα σύνολο προτάσεων. Σε αυτή την περίπτωση ένας βαθμολογητής καλείται να αξιολογήσει ολόκληρη την παράγραφο κάτι που σαφώς αυξάνει το βαθμό στον οποίο πρέπει να είναι συγκεντρωμένος όταν ακούει τη συνθετική ομιλία. Αυτό επίσης σχετίζεται και με το μέγεθος κάθε παραγράφου που του δίνεται για βαθμολόγηση. Αντιθέτως είναι ευκολότερο για έναν βαθμολογητή να είναι συγκεντρωμένος όταν του δίνεται να ακούσει μια μεμονωμένη πρόταση. Το βασικό πλεονέκτημα όμως της αξιολόγησης ολόκληρης παραγράφου, είναι ότι μπορεί να γίνει αντιληπτό το περιεχόμενο, το συναίσθημα καθώς και η ροή του λόγου και συνεπώς η αξιολόγηση να είναι πιο αντιπροσωπευτική.

### Αξιολόγηση ζεύγους context-stimulus

Ο τρίτος τρόπος αξιολόγησης συνδυάζει τους δύο πρώτους και λειτουργεί ως εξής. Αρχικά παρουσιάζεται στον βαθμολογητή το context. Το context μπορεί να αποτελείται είτε από το κείμενο (text) είτε από την πραγματική ομιλία (speech) μίας ή περισσότερων προτάσεων μιας παραγράφου. Στη συνέχεια ο βαθμολογητής καλείται να αξιολογήσει τη συνθετική ομιλία (stimulus) ενός άλλου μέρους της παραγράφου. Με αυτό τον τρόπο ο βαθμολογητής αρχικά κατανοεί το περιεχόμενο (συναίσθημα, ροή κ.τ.λ.) βάσει του οποίου πρέπει να παραχθεί η ομιλία και στη συνέχεια βαθμολογεί με πιο αντικειμενικό τρόπο τη φυσικότητα και καταλληλότητα της ομιλίας που συνέθεσε το σύστημα. Τα πλεονεκτήματα αυτής της προσέγγισης είναι εμφανή αφού ο βαθμολογητής αξιολογεί ένα μικρό μέρος της παραγράφου αφού πρώτα έχει κατανοήσει το περιεχόμενο αυτής. Ένα ερώτημα που προκύπτει είναι το πόσες προτάσεις της παραγράφου πρέπει να επιλεγούν σαν context για να παρουσιαστούν στον βαθμολογητή και πόσες πρέπει να δοθούν σαν stimulus για αξιολόγηση. Στη Εικόνα 3.8 φαίνονται τρία παραδείγματα παραγράφων προς αξιολόγηση με την προσέγγιση του context-stimulus. Στον βαθμολογητή παρουσιάζεται πρώτα το context (προτάσεις με κίτρινο χρώμα) και έπειτα βαθμολογεί την ομιλία που παράγεται από το κείμενο stimulus (προτάσεις με πράσινο χρώμα).




Εικόνα 3.8: Αξιολόγηση συστήματος text to speech με την μέθοδο context-stimulus. Σε κάθε ένα από τα τρία παραδείγματα αρχικά παρουσιάζεται στον βαθμολογητή το context (κίτρινο χρώμα) που είναι είτε το κείμενο είτε η πραγματική εκφώνησή του. Στη συνέχεια βάσει του context αξιολογεί το stimulus, δηλαδή τη συνθετική ομιλία του κειμένου που παρουσιάζεται με πράσινο χρώμα. [Cla+19]

### 3.5.1 Δεδομένα

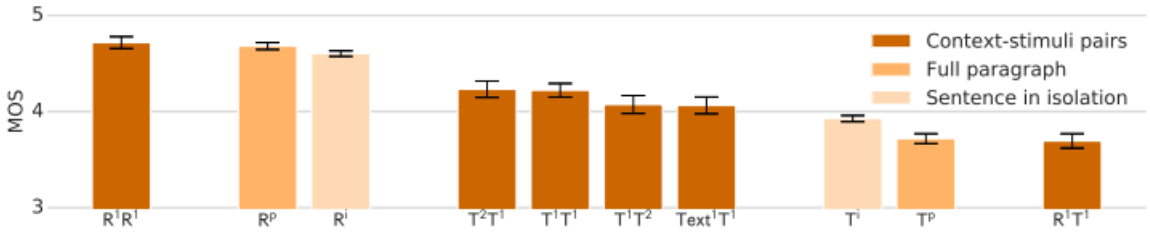
Για να γίνει η σύγκριση των τριών μεθόδων αξιολόγησης χρησιμοποιήθηκαν δύο σύνολα δεδομένων. Το πρώτο περιείχε εκφωνήσεις ανάγνωσης ειδήσεων από έναν ομιλητή μαζί με τα αντίστοιχα κείμενα. Κάθε παράγραφος προς εκφώνηση περιείχε τουλάχιστον δύο προτάσεις και όλο το σύνολο δεδομένων περιείχε συνολικά 103 παραγράφους, όπου κατά μέσο όρο κάθε παράγραφος περιείχε 3 προτάσεις. Το δεύτερο dataset αποτελούνταν από εκφωνήσεις ανάγνωσης συζητήσεων από δύο ομιλητές, όπου κάθε ένας σε σειρά εκφωνεί μία ή περισσότερες προτάσεις. Χρησιμοποιήθηκαν δύο ζευγάρια ομιλητών, όπου το πρώτο ηχογράφησε 42 συζητήσεις και το δεύτερο 71 συζητήσεις. Για όλες τις περιπτώσεις αρχικά χρησιμοποιήθηκε ένα μοντέλο [Ken+19] για την παραγωγή παραμέτρων όπως το $F_0, c_0$ και η διάρκεια. Στη συνέχεια το μοντέλο Parallel WaveNet [Oor+18] χρησιμοποιώντας αυτές τις παραμέτρους και άλλα γλωσσικά χαρακτηριστικά παρήγαγε την συνθετική ομιλία. Για την αξιολόγηση των παραγόμενων εκφωνήσεων από το πρώτο σύνολο δεδομένων, χρησιμοποιήθηκε ένα σύνολο παραγράφων στο οποίο το μοντέλο δεν εκπαιδεύτηκε αρχικά, ενώ για την δεύτερη περίπτωση, οι συγγραφείς χρησιμοποίησαν ένα μέρος των δεδομένων για αξιολόγηση το οποίο χρησιμοποιήθηκε και κατά την εκπαίδευση. Για όλες τις αξιολογήσεις έγινε χρήση της μετρικής MOS στην κλίμακα 1 έως 5 με ενδιάμεσες διαβαθμίσεις της τάξης του 0.5.

Για το πρώτο σύνολο δεδομένων ανάγνωσης ειδήσεων από έναν ομιλητή μελετήθηκαν οι εξής περιπτώσεις. Όσον αφορά την αξιολόγηση μεμονωμένων προτάσεων, σε κάθε βαθμολογητή δόθηκαν η πραγματική $\mathbf{R}^i$ και η συνθετική $\mathbf{T}^i$ εκφώνηση μιας πρότασης. Στην περίπτωση αξιολόγησης ολόκληρων παραγράφων δόθηκαν επίσης οι πραγματικές $\mathbf{R}^p$ και οι συνθετικές $\mathbf{T}^p$ εκφωνήσεις κάθε παραγράφου. Τέλος για τη μέθοδο context-stimulus εξετάστηκαν οι εξής περιπτώσεις: $(\mathbf{R}^1, \mathbf{R}^1)$, $(\mathbf{R}^1, \mathbf{T}^1)$, $(\mathbf{T}^1, \mathbf{T}^1)$, $(\mathbf{Text}^1, \mathbf{T}^1)$, $(\mathbf{T}^2, \mathbf{T}^1)$ και $(\mathbf{T}^1, \mathbf{T}^2)$. Σε κάθε περίπτωση το πρώτο στοιχείο αντιστοιχεί στο context και το δεύτερο στο stimulus. Για παράδειγμα η περίπτωση $(\mathbf{R}^1, \mathbf{T}^1)$ σημαίνει ότι στον βαθμολογητή δόθηκε αρχικά η πραγματική εκφώνηση μίας πρότασης $\mathbf{R}^1$ ως context και έπειτα αξιολόγησε τη συνθετική εκφώνηση μιας άλλης πρότασης $\mathbf{T}^1$ που παρήγαγε το σύστημα. Η περίπτωση $(\mathbf{Text}^1, \mathbf{T}^1)$ υποδηλώνει ότι σαν context δόθηκε μία πρόταση σε μορφή κειμένου $\mathbf{Text}^1$, ενώ σαν stimulus αξιολογήθηκε η παραγόμενη ομιλία από μια άλλη πρόταση $\mathbf{T}^1$ της παραγράφου. Επίσης ο συμβολισμός $\mathbf{T}^2$ αντιστοιχεί στην εκφώνηση συνθετικής ομιλίας δύο προτάσεων. Το δεύτερο σύνολο δεδομένων περιείχε συζητήσεις από δύο ζευγάρια ομιλητών $(\mathbf{F1}, \mathbf{M1})$ και $(\mathbf{F2}, \mathbf{M2})$, όπου το $\mathbf{F}$ αντιστοιχεί σε γυναίκα και το $\mathbf{M}$ σε άντρα. Οι βαθμολογητές αξιολόγησαν αντίστοιχους συνδυασμούς περιπτώσεων πραγματικής και συνθετικής ομιλίας, όπως στο πρώτο σύνολο δεδομένων.

### 3.5.2 Αποτελέσματα

Στην ενότητα αυτή παρουσιάζουμε τα αποτελέσματα από κάθε περίπτωση αξιολόγησης. Στο Διάγραμμα 3.9 φαίνονται τα ραβδογράμματα που αντιστοιχούν στη μετρική MOS κάθε πειράματος για το πρώτο σύνολο δεδομένων.

Όπως είναι λογικό τα υψηλότερα αποτελέσματα δίνονται από τους βαθμολογητές στις εκφωνήσεις που αντιστοιχούν σε πραγματική ομιλία. Από τα πρώτα τρία ραβδογράμματα είναι εμφανής η διαφορά στις τρεις μεθόδους αξιολόγησης (context-stimulus $\mathbf{R}^1\mathbf{R}^1$, ολόκληρη παράγραφος $\mathbf{R}^p$ και μεμονωμένη πρόταση $\mathbf{R}^i$), με υψηλότερο MOS περίπου 4.7 να δίνει η περίπτωση $\mathbf{R}^1\mathbf{R}^1$. Στα επόμενα τέσσερα ραβδογράμματα δίνεται ως context μια συνθετική εκφώνηση ή το κείμενο μιας πρότασης ($\mathbf{Text}^1$) και ακολουθεί επίσης συνθετική εκφώνηση. Από αυτό το block τα δύο πρώτα ραβδογράμματα αντιστοιχούν περίπου στο ίδιο MOS με τιμή περίπου στο 4.3. Σε αυτή την περίπτωση

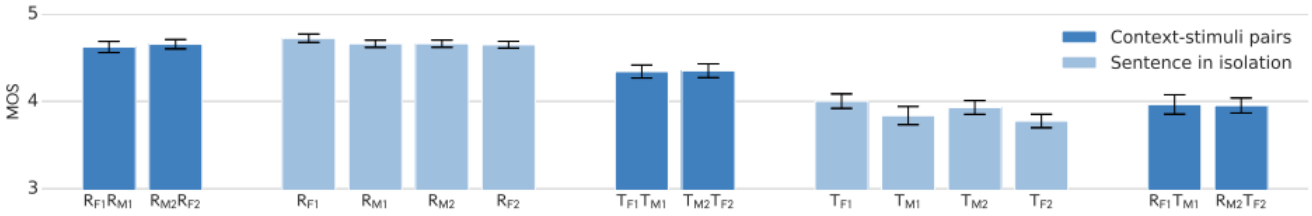



Διάγραμμα 3.9: Ραβδογράμματα που αντιστοιχούν στη μετρική του MOS για κάθε περίπτωση αξιολόγησης στο σύνολο δεδομένων ανάγνωσης ειδήσεων. Το γράμματα **R** και **T** αντιστοιχούν σε πραγματική και συνθετική εκφώνηση αντίστοιχα. [Cla+19]

το μέγεθος του context που παρουσιάζεται στον βαθμολογητή μειώνεται αρχικά από δύο εκφωνήσεις σε μία (από $\mathbf{T}^2$ γίνεται $\mathbf{T}^1$ στο δεύτερο ραβδόγραμμα). Τα δύο επόμενα ραβδογράμματα αντιστοιχούν σε ελαφρώς χαμηλότερη τιμή του MOS με τιμή περίπου στο 4.1. Εδώ το μέγεθος του context παραμένει σταθερό σε μία πρόταση (είτε είναι εκφώνηση $\mathbf{T}^1$ είτε κείμενο $\mathbf{Text}^1$), αλλά μειώνεται το μέγεθος του stimulus από δύο σε μία εκφώνηση. Επομένως παρατηρούμε ότι το μέγεθος του context δεν επηρεάζει σημαντικά το αποτέλεσμα (MOS $\approx 4.3$), σε αντίθεση όμως με το μέγεθος του stimulus, όπου η αύξησή του οδηγεί σε σχετική μείωση (MOS $\approx 4.1$). Έπειτα ακολουθούν δύο ραβδογράμματα που αντιστοιχούν στις αξιολόγησεις μεμονωμένων προτάσεων $\mathbf{T}^i$ και ολόκληρης παραγράφου $\mathbf{T}^p$. Εδώ παρατηρούμε ότι η συνθετική ομιλία που προκύπτει από ολόκληρη την παράγραφο δίνει μικρότερη τιμή για το MOS σε σχέση με την ομιλία που προκύπτει από μια μεμονωμένη πρόταση. Αυτό πιθανόν οφείλεται στο ότι ο βαθμολογητής χρειάζεται να είναι συγκεντρωμένος για περισσότερο χρόνο όταν βαθμολογεί παραγράφους και έτσι ενδέχεται να εστιάσει σε κάτι που δεν ακούστηκε φυσικά κατά την εκφώνηση και κατά συνέπεια να δώσει μικρότερη βαθμολογία για την εκφώνηση ολόκληρης της παραγράφου. Το τελευταίο ραβδόγραμμα αντιστοιχεί στην περίπτωση όπου ως context δίνεται η πραγματική ομιλία μιας πρότασης και ως stimulus η συνθετική ομιλία μιας πρότασης. Το αποτέλεσμα αυτό φαίνεται λογικό αφού όταν δοθεί πρώτα η πραγματική εκφώνηση ως context, τότε ο βαθμολογητής δίνει μεγάλη προσοχή στο αν η παραγόμενη φωνή είναι ίδιας ποιότητας με την αρχική εκφώνηση.

Στο Διάγραμμα 3.10 παρουσιάζονται τα αποτελέσματα από το δεύτερο σύνολο δεδομένων που περιέχει διαλόγους μεταξύ δύο ομιλητών. Τα ζευγάρια των ομιλητών είναι τα $(\mathbf{F1}, \mathbf{M1})$ και $(\mathbf{F2}, \mathbf{M2})$. Από κάθε διάλογο χρησιμοποιήθηκαν μόνο η πρώτη εκφώνηση από κάθε ομιλητή. Για παράδειγμα αν ο διάλογος αποτελούνταν από 4 προτάσεις, δύο για κάθε ομιλητή και είχε την εξής σειρά: $\mathbf{F}_{s_1}, \mathbf{M}_{s_1}, \mathbf{F}_{s_2}, \mathbf{M}_{s_2}$, όπου $\mathbf{F}_{s_i}, \mathbf{M}_{s_i}$ είναι η $i-$οστή πρόταση που εκφωνούν οι ομιλητές **F** και **M** αντίστοιχα, τότε κατά την αξιολόγηση λήφθηκαν υπόψιν μόνο οι δύο πρώτες εκφωνήσεις $\mathbf{F}_{s_1}$ και $\mathbf{M}_{s_1}$. Στην περίπτωση της μεθόδου context-stimulus ο βαθμολογητής άκουγε πρώτα την εκφώνηση από τον πρώτο ομιλητή και έπειτα αξιολογούσε την εκφώνηση του δεύτερου ομιλητή.

Από το Διάγραμμα 3.10 παρατηρούμε ότι οι πραγματικές εκφωνήσεις αντιστοιχούν σε υψηλότερη τιμή του MOS με τιμή περίπου στο 4.6, είτε όταν αξιολογούνται βάσει της μεθόδου context-stimulus $(\mathbf{R}_{F_1}\mathbf{R}_{M_1}, \mathbf{R}_{M_2}\mathbf{R}_{F_2})$ είτε ως μεμονωμένες εκφωνήσεις $(\mathbf{R}_{F_1}, \mathbf{R}_{M_1}, \mathbf{R}_{M_2}, \mathbf{R}_{F_2})$. Τα αποτελέσματα του MOS είναι παρόμοια, με μικρή διαφορά μεταξύ των περιπτώσεων $\mathbf{R}_{F_1}$ και $\mathbf{R}_{F_1}\mathbf{R}_{M_1}$. Έπειτα ακολουθούν οι περιπτώσεις που περιλαμβάνουν μόνο συνθετική φωνή είτε με τη μέθοδο context-stimulus $(\mathbf{T}_{F_1}\mathbf{T}_{M_1}, \mathbf{T}_{F_2}\mathbf{T}_{M_2})$ (MOS $\approx 4.4$), είτε αξιολογώντας συνθετικές εκφωνήσεις μεμονωμένων προτάσεων $(\mathbf{T}_{F_1}, \mathbf{T}_{M_1}, \mathbf{T}_{F_2}, \mathbf{T}_{M_2})$ (MOS $\approx 3.8$ με 4). Τέλος, όταν δίνεται ως context στο βαθμολογητή μία πραγματική εκφώνηση και το stimulus αποτελείται από μια συνθετική εκφώνηση $(\mathbf{R}_{F_1}\mathbf{T}_{M_1}, \mathbf{R}_{M_2}\mathbf{T}_{F_2})$, τότε οι τιμές για την μετρική του MOS κυμαίνονται στο




Διάγραμμα 3.10: Ραβδογράμματα που αντιστοιχούν στη μετρική του MOS για κάθε περίπτωση αξιολόγησης στο σύνολο δεδομένων ανάγνωσης συζητήσεων, από δύο ζευγάρια ομιλητών (**F1**, **M1**) και (**F2**, **M2**). [Cla+19]

3.9. Εδώ το αποτέλεσμα είναι σαφώς μικρότερο σε σχέση με τις περιπτώσεις που παρουσιάζεται ως context συνθετική ομιλία, π.χ. $\mathbf{T}_{F_1}\mathbf{T}_{M_1}$. Το γεγονός αυτό συμφωνεί και με την αντίστοιχη παρατήρηση στο πρώτο dataset και πιθανόν υποδηλώνει ότι οι βαθμολογητές εστιάζουν αρκετά στο αν η ποιότητα της εκφώνησης stimulus αντικατοπτρίζει την ποιότητα του context. Προφανώς μια συνθετική εκφώνηση θα έχει χαμηλότερη ποιότητα από μια πραγματική και αυτό δικαιολογεί τη μείωση του MOS για τις περιπτώσεις $\mathbf{R}_{F_1}\mathbf{T}_{M_1}$, $\mathbf{R}_{M_2}\mathbf{T}_{F_2}$.

Συμπερασματικά λοιπόν, είναι εμφανές ότι για την πλήρη αξιολόγηση ενός συστήματος σύνθεσης φωνής από κείμενο, χρειάζεται ένας συνδυασμός μεθόδων αξιολόγησης και όχι απλά η βαθμολόγηση μεμονωμένων προτάσεων, όπως συμβαίνει στα περισσότερα μοντέλα TTS. Αυτό είναι αναγκαίο κυρίως για την περίπτωση όπου η σύνθεση φωνής γίνεται χρησιμοποιώντας ολόκληρες παραγράφους ή κείμενα, όπως ενδεχομένως να χρειάζεται στην περίπτωση των audio books, αφού όπως είδαμε από τα σχετικά αποτελέσματα οι βαθμολογητές αξιολογούν με διαφορετικό τρόπο τις εκφωνήσεις απλών προτάσεων, ολόκληρων παραγράφων και αυτές που προκύπτουν με τη μέθοδο context-stimulus.

## 3.6 Συμπέρασμα

Κλείνοντας το κεφάλαιο αυτό παίρνουμε μια εικόνα για το πόσο ευρύ είναι το πεδίο που μελετάμε. Ειδικότερα σε σύγχρονες εφαρμογές που χρησιμοποιούν τη σύνθεση φωνής και χρησιμοποιούνται σε κάποιο παραγωγικό περιβάλλον, είναι σαφές ότι δεν αρκεί μόνο να μπορούμε να παράγουμε φωνή από ένα κείμενο. Πέραν αυτού θα πρέπει ένα τέτοιο σύστημα να είναι σε θέση για παράδειγμα να συνθέτει ομιλία με το επιθυμητό συναίσθημα, να συνθέτει ομιλία σε real time ταχύτητα αλλά και σε γλώσσες όπου δεν είναι εύκολο να εκπαιδευτεί ένα μοντέλο λόγω έλλειψης δεδομένων. Στο επόμενο μας κεφάλαιο θα παρουσιάσουμε τη μελέτη και τους πειραματισμούς μας πάνω στη σύνθεση φωνής από κείμενο για την ελληνική γλώσσα.

# Κεφάλαιο 4

# Σύνθεση φωνής στα ελληνικά

## 4.1 Εισαγωγή

Στο κεφάλαιο αυτό περιγράφουμε τα πειράματα που πραγματοποιήθηκαν για τη σύνθεση φωνής από κείμενο στην ελληνική γλώσσα. Τα ελληνικά είναι μία low resource γλώσσα με την έννοια ότι δεν υπάρχουν αρκετά δεδομένα (ηχογραφήσεις) καλής ποιότητας ώστε να εκπαιδευτεί ένα σύστημα σύνθεσης φωνής από την αρχή. Το γεγονός αυτό αποτέλεσε και ένα από τα εμπόδια στα πειράματά μας, το οποίο όμως ξεπεράστηκε αξιοποιώντας τεχνικές όπως η μεταφορά μάθησης και η συλλογή νέων δεδομένων. Τα βασικά μοντέλα που χρησιμοποιήθηκαν στη μελέτη μας είναι το Tacotron2 και το WaveGlow. Στο πρώτο μέρος του κεφαλαίου γίνεται αναφορά στα δεδομένα που αξιοποιήθηκαν στην ελληνική και ισπανική γλώσσα καθώς και στη διαδικασία επεξεργασίας αυτών ώστε να έρθουν στην επιθυμητή μορφή και να δοθούν ως είσοδος στο εκάστοτε μοντέλο. Έπειτα παρουσιάζονται οι αρχικές δοκιμές εκπαίδευσης των μοντέλων τόσο στα ισπανικά (ως γλώσσα με πολλά διαθέσιμα δεδομένα) όσο και στα ελληνικά. Στη συνέχεια περιγράφονται τα πειράματα στα οποία χρησιμοποιήθηκε η μεταφορά μάθησης και έδωσαν τελικά τα καλύτερα αποτελέσματα. Ως vocoder σε όλα τα πειράματα χρησιμοποιήθηκε ένα προεκπαιδευμένο μοντέλο WaveGlow στην αγγλική γλώσσα, αφού ύστερα από δοκιμές διαπιστώθηκε ότι έδινε ικανοποιητικά αποτελέσματα ως προς την ποιότητα του παραγόμενου ήχου και σε γλώσσες όπως τα ελληνικά και τα ισπανικά. Αυτό λοιπόν αποτέλεσε μια ένδειξη ότι το μοντέλο WaveGlow είναι ανεξάρτητο τόσο από τη γλώσσα ομιλίας όσο και από την ταυτότητα του ομιλητή. Συνεπώς δόθηκε βαρύτητα στο πρώτο τμήμα του συστήματος σύνθεσης φωνής, δηλαδή το μοντέλο Tacotron2. Για να βελτιώσουμε περαιτέρω την ποιότητα της παραγόμενης φωνής, προχωρήσαμε σε συλλογή νέων δεδομένων στην ελληνική γλώσσα από μια γυναίκα ομιλήτρια με ηχογραφήσεις συνολικής διάρκειας περίπου 19.5 ώρες. Τα αποτελέσματα των πειραμάτων μας αξιολογούνται ως προς τη φυσικότητα του παραγόμενου ήχου μέσω ερωτηματολογίου στη κλίμακα MOS. Αν και η ποιότητα των παραγόμενων ηχητικών δειγμάτων στα ελληνικά είναι σχετικά καλή με τιμή MOS $\approx 3.43$, όπως προκύπτει από τις απαντήσεις των βαθμολογητών, εντούτοις υπάρχει σαφώς περιθώριο βελτίωσης ώστε η συνθετική φωνή να είναι πιο κοντά στα ανθρώπινα επίπεδα. Κλείνουμε το κεφάλαιο αυτό παρουσιάζοντας ορισμένα συμπεράσματα αλλά και μελλοντικές επεκτάσεις της εργασίας που αφορούν τη βελτίωση των μοντέλων αλλά και την αξιοποίηση νέων μοντέλων με σκοπό την παραγωγή φωνής με ακόμα καλύτερη ποιότητα.

## 4.2 Δεδομένα

Στην ενότητα αυτή περιγράφουμε τα δεδομένα που χρησιμοποιούμε για την εκπαίδευση και αξιολόγηση των μοντέλων μας. Το πρώτο σύνολο δεδομένων που αξιοποιούμε είναι το M-AILABS Speech Dataset στην ισπανική γλώσσα. Το σύνολο αυτό περιέχει εκφωνήσεις βιβλίων από το LibriVox και το Project Gutenberg. Η μορφή του είναι παρόμοια με το σύνολο δεδομένων LJSpeech, δηλαδή αποτελείται από ζεύγη ηχητικών δειγμάτων και των αντίστοιχων κειμένων (transcriptions). Τα ηχητικά δείγματα είναι σε μορφή .wav, mono και ο ρυθμός δειγματοληψίας τους είναι στα 16000 Hz. Για τα πειράματά μας επιλέχθηκε ένας άνδρας ομιλητής με όνομα Tux. Ο ομιλητής αυτός αφηγείται συνολικά επτά βιβλία με συνολική διάρκεια ηχογραφήσεων περίπου 55 ώρες, αλλά για τα πειράματά μας χρησιμοποιήθηκαν μόνο τα ηχητικά δείγματα από τα βιβλία "Eneida" και "La Batalla de Los Arapiles", διότι είχαν ελαφρώς καλύτερη ποιότητα ήχου σε σχέση με τα υπόλοιπα βιβλία. Όλα τα ζεύγη audio-text είναι 10198 με συνολική διάρκεια ηχογραφήσεων 18 ώρες και 47 λεπτά. Ορισμένα από αυτά τα ηχητικά δείγματα έχουν μικρή διάρκεια και χρειάζεται να αφαιρεθούν, διότι διαφορετικά το σφάλμα του μοντέλου WaveGlow απειρίζεται κατά την εκπαίδευση με αποτέλεσμα να σταματά η εκμάθηση των παραμέτρων. Έτσι αφαιρώντας τα ηχητικά κλιπ που έχουν λιγότερο από 17500 δείγματα, δηλαδή διάρκεια μικρότερη των 1.09 δευτερολέπτων, καταλήγουμε τελικά σε 9799 ζεύγη ήχου-κειμένου. Στη συνέχεια χωρίζουμε τα δεδομένα αυτά σε train (9669), validation (100) και test (30) δείγματα, ώστε η συνολική διάρκεια ηχογραφήσεων που θα χρησιμοποιήσουμε για την εκπαίδευση των μοντέλων στα ισπανικά να είναι περίπου 18 ώρες και 27 λεπτά. Όσον αφορά τα ελληνικά δεδομένα, τα οποία παραχωρήθηκαν από το Ινστιτούτο Επεξεργασίας του Λόγου (ΙΕΛ) του Ερευνητικού Κέντρου Αθηνά, περιέχουν εκφωνήσεις από τέσσερις ομιλήτριες. Οι ομιλήτριες αυτές αφηγούνται τρία βιβλία με συνολική διάρκεια ηχογραφήσεων 15 ώρες και 27 λεπτά. Συγκεκριμένα οι πρώτες δύο εκφωνούν διαφορετικές σελίδες από το βιβλίο «Τα κακομαθημένα παιδιά της Ιστορίας» του Κώστα Κωστή. Για την πρώτη ομιλήτρια (speaker 1) δόθηκαν τρία mp3 αρχεία συνολικής διάρκειας 2 ώρες και 18 λεπτά ενώ για την δεύτερη ομιλήτρια (speaker 2) δόθηκαν δύο mp3 αρχεία διάρκειας 1 ώρα και 44 λεπτά. Η τρίτη ομιλήτρια (speaker 3) εκφωνεί το βιβλίο της Αγγελικής Νικολούλη «Θάνατος με χείλη κόκκινα» και το αρχείο mp3 που δόθηκε είχε διάρκεια 4 ώρες και 37 λεπτά, ενώ η τέταρτη ομιλήτρια εκφωνεί το βιβλίο του Πάμπλο Γκουτιέρεθ «Κομμένα Κεφάλια» με συνολική διάρκεια 6 ώρες και 48 λεπτά. Μαζί με τα ηχητικά δείγματα δόθηκαν και τα αντίστοιχα κείμενα σε μορφή εικόνας και pdf. Στη συνέχεια αφού επεξεργαστούμε τα ελληνικά δεδομένα καταλήγουμε σε 7494 ζεύγη ήχου-κειμένου και για τις τέσσερις ομιλήτριες με διάρκεια περίπου 14 ώρες και 10 λεπτά. Στον Πίνακα 4.1 συνοψίζονται τα στοιχεία για τη διάρκεια και το πλήθος των εκφωνήσεων ανά ομιλήτρια. Συγκεκριμένα, στην

| | Duration (h) | Utterances |
|---|---|---|
| speaker 1 | 2.08 | 707 |
| speaker 2 | 1.55 | 527 |
| speaker 3 | 4.05 | 3491 |
| speaker 4 | 6.48 | 2769 |
| Total | **14.16** | **7494** |

Πίνακας 4.1: Διάρκεια και πλήθος εκφωνήσεων ανά ομιλήτρια στα ελληνικά.

πρώτη ομιλήτρια αντιστοιχούν 707 ζεύγη audio-text διάρκειας 2 ώρες και 5 λεπτά, στη δεύτερη ομιλήτρια 527 ζεύγη διάρκειας 1 ώρα και 33 λεπτά, στη τρίτη ομιλήτρια 3491 ζεύγη διάρκειας 4

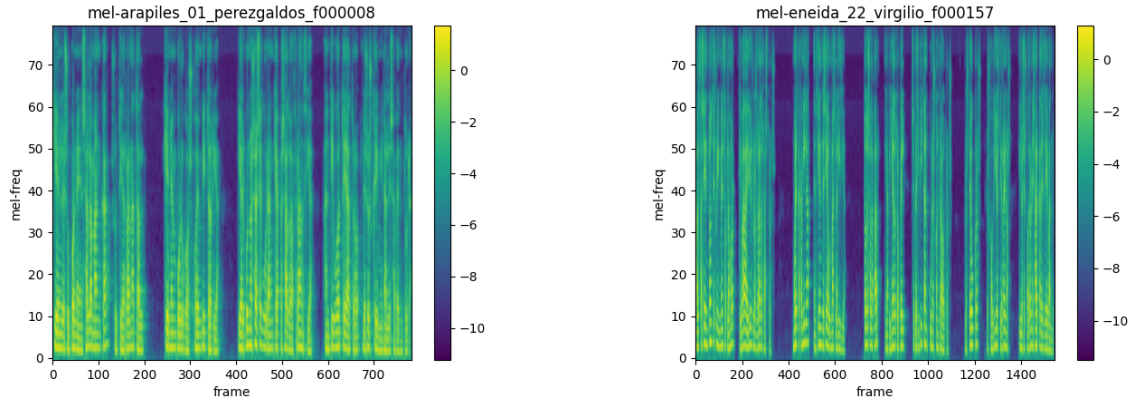
ώρες και 3 λεπτά ενώ στην τέταρτη ομιλήτρια 2769 ζεύγη συνολικής διάρκειας 6 ώρες και 29 λεπτά. Ύστερα από το διαχωρισμό σε train (7464), validation (30) και test (10) δείγματα, έχουμε ότι η συνολική διάρκεια των δεδομένων εκπαίδευσης στην ελληνική γλώσσα από όλους τους ομιλητές είναι 14 ώρες και 7 λεπτά. Στη συνέχεια παρουσιάζουμε τα βήματα επεξεργασίας των δεδομένων προκειμένου να έρθουν στην επιθυμητή μορφή.

## 4.3 Επεξεργασία των δεδομένων

Ως γνωστόν για την εκπαίδευση των μοντέλων Tacotron2 και WaveGlow που θα χρησιμοποιήσουμε, θα πρέπει τα δεδομένα μας να έχουν την κατάλληλη μορφή. Για παράδειγμα το μοντέλο Tacotron2 χρειάζεται ζεύγη κειμένου και του αντίστοιχου φασματογραφήματος στην κλίμακα mel, το οποίο εξάγεται από το ηχητικό δείγμα. Τα βήματα επεξεργασίας για τα ισπανικά και ελληνικά δεδομένα είναι παρόμοια. Στο σημείο αυτό αναφέρουμε ότι για την εξαγωγή των φασματογραφημάτων από τα ηχητικά δείγματα αλλά και για την εκπαίδευση των μοντέλων, χρησιμοποιούμε τον κώδικα που βρίσκεται στο repository της NVIDIA Tacotron 2 And WaveGlow v1.10 For PyTorch.

### Ισπανικά

Αρχικά κατεβάζουμε τα ισπανικά ηχητικά δείγματα χρησιμοποιώντας κώδικα που βρίσκεται στο repository Deep Mind - Tacotron2. Μαζί με τα ηχητικά δείγματα δημιουργείται και ένα txt αρχείο που περιέχει τα paths των ηχητικών δειγμάτων μαζί με τα αντίστοιχα transcriptions. Στη συνέχεια αφού αφαιρέσουμε τα ηχητικά δείγματα με πολύ μικρή διάρκεια όπως αναφέραμε στην προηγούμενη ενότητα, σειρά έχει η εξαγωγή των φασματογραφημάτων στην κλίμακα mel. Για να το επιτύχουμε αυτό εφαρμόζεται η μέθοδος STFT με τις εξής βασικές παραμέτρους: sampling rate 16000 Hz, filter length 1024, hop length 200 (12.5 ms), window length 800, mel-fmin 0 Hz, mel-fmax 8000 Hz και n-mels 80. Στην Εικόνα 4.1 φαίνονται δύο mel spectrograms που προκύπτουν από το σύνολο δεδομένων χρησιμοποιώντας τις παραπάνω παραμέτρους.



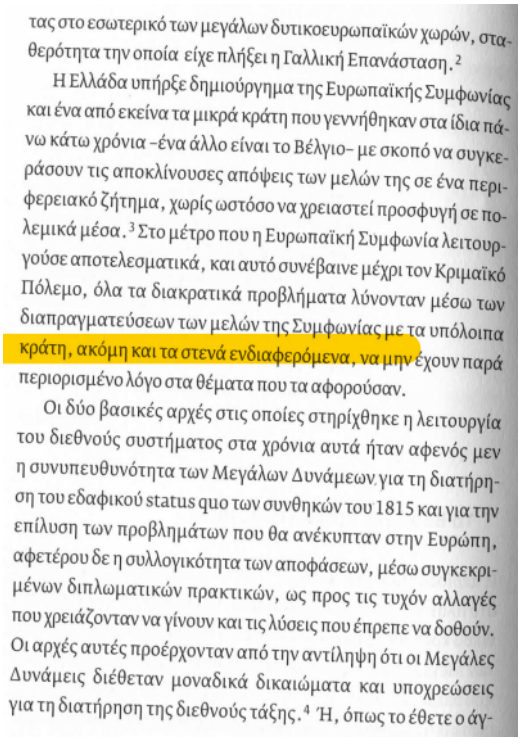

Εικόνα 4.1: Φασματογραφήματα στην κλίμακα mel για δύο ηχητικά δείγματα στα ισπανικά.

Αφού εξάγουμε τα φασματογραφήματα στην κλίμακα mel, εν συνεχεία δημιουργούμε ένα νέο αρχείο txt, το οποίο περιέχει ζεύγη κειμένου και του αντίστοιχου path στο οποίο βρίσκεται το mel-spectrogram (mel-text filelist). Το αρχείο αυτό χρησιμοποιείται κατά την εκπαίδευση του μοντέλου Tacotron2, ώστε να διαβάζονται τα mel spectrograms και τα αντίστοιχα transcriptions

από τα paths τους. Επιπλέον δημιουργείται και το αρχείο που περιέχει ζεύγη audio path-mel path (audio-mel filelist) που χρησιμοποιείται κατά την εκπαίδευση του μοντέλου WaveGlow. Τέλος χωρίζουμε τα filelists που προκύπτουν ώστε να περιέχουν ξεχωριστά τα paths και τα transcriptions για κάθε διαχωρισμό train, validation και test. Για παράδειγμα καθένα από τα τρία αρχεία audio-text filelist, mel-text filelist, audio-mel filelist παράγει τρία επιμέρους αρχεία για κάθε σύνολο διαχωρισμού (π.χ. από το mel-text filelist προκύπτουν τα mel-text-train, mel-text-val, mel-text-test).

### Ελληνικά

Όσον αφορά τα ελληνικά δεδομένα χρειάστηκε να γίνουν περισσότερα βήματα προκειμένου να έρθουν στην επιθυμητή μορφή και στη συνέχεια να εξάγουμε τα mel spectrograms και τα τελικά filelists. Σε πρώτη φάση θα πρέπει να επεξεργαστούμε τα αρχεία pdf που περιέχουν ολόκληρο το κείμενο ενός βιβλίου και να το χωρίσουμε σε μικρές φράσεις ώστε να μπορούν να δοθούν ως είσοδος στα μοντέλα μας. Για το πρώτο βιβλίο «Τα κακομαθημένα παιδιά της ιστορίας» που εκφωνείται από τις δύο πρώτες ομιλήτριες, η διαδικασία που ακολουθήσαμε έχει ως εξής. Επειδή η μορφή που μας δόθηκε για το βιβλίο αυτό ήταν σε εικόνα (και όχι κανονικό pdf), αρχικά λάβαμε στιγμιότυπα (screenshots) από κάθε σελίδα του και στη συνέχεια χρησιμοποιήσαμε το εργαλείο OCR-Greek, που κάνει οπτική αναγνώριση χαρακτήρων, ώστε να μετατρέψουμε το στιγμιότυπο κάθε σελίδας στο αντίστοιχο txt αρχείο. Στην Εικόνα 4.2 φαίνεται το αποτέλεσμα εξαγωγής του κειμένου που αντιστοιχεί στο στιγμιότυπο μιας συγκεκριμένης σελίδας από το πρώτο βιβλίο.



τας στο εσωτερικό των μεγάλων δυτικοευρωπαϊκών χωρών, στα-
θερότητα την οποία είχε πλήξει η Γαλλική Επανάσταση.[2]
Η Ελλάδα υπήρξε δημιούργημα της Ευρωπαϊκής Συμφωνίας
και ένα από εκείνα τα μικρά κράτη που γεννήθηκαν στα ίδια πά-
νω κάτω χρόνια –ένα άλλο είναι το Βέλγιο– με σκοπό να συγκε-
ράσουν τις αποκλίνουσες απόψεις των μελών της σε ένα περι-
φερειακό ζήτημα, χωρίς ωστόσο να χρειαστεί προσφυγή σε πο-
λεμικά μέσα.[3] Στο μέτρο που η Ευρωπαϊκή Συμφωνία λειτουρ-
γούσε αποτελεσματικά, και αυτό συνέβαινε μέχρι τον Κριμαϊκό
Πόλεμο, όλα τα διακρατικά προβλήματα λύνονταν μέσω των
διαπραγματεύσεων των μελών της Συμφωνίας με τα υπόλοιπα
κράτη, ακόμη και τα στενά ενδιαφερόμενα, να μην έχουν παρά
περιορισμένο λόγο στα θέματα που τα αφορούσαν.
Οι δύο βασικές αρχές στις οποίες στηρίχθηκε η λειτουργία
του διεθνούς συστήματος στα χρόνια αυτά ήταν αφενός μεν
η συνυπευθυνότητα των Μεγάλων Δυνάμεων για τη διατήρη-
ση του εδαφικού status quo των συνθηκών του 1815 και για την
επίλυση των προβλημάτων που θα ανέκυπταν στην Ευρώπη,
αφετέρου δε η συλλογικότητα των αποφάσεων, μέσω συγκεκρι-
μένων διπλωματικών πρακτικών, ως προς τις τυχόν αλλαγές
που χρειάζονταν να γίνουν και τις λύσεις που έπρεπε να δοθούν.
Οι αρχές αυτές προέρχονταν από την αντίληψη ότι οι Μεγάλες
Δυνάμεις διέθεταν μοναδικά δικαιώματα και υποχρεώσεις
για τη διατήρηση της διεθνούς τάξης.[4] Ή, όπως το έθετε ο άγ-

(a)

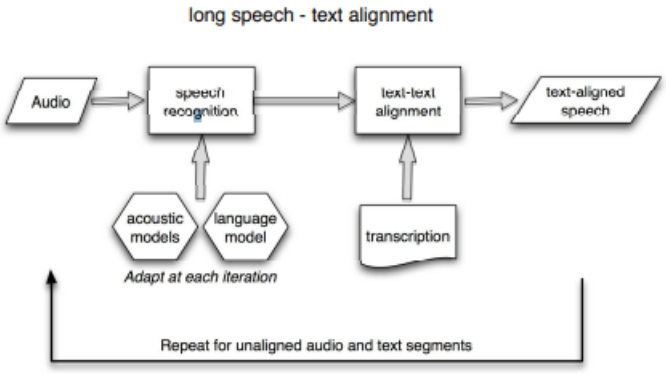
τας στο εσωτερικό των μεγάλων δυτικοευρωπαϊκών χωρών , στα
θερότητα την οποία είχε πλήξει ηΓαλλική Επανάσταση .2
Η Ελλάδα υπήρξε δημιούργημα της Ευρωπαϊκής Συμφωνίας
και ένα από εκείνα ταμικρά κράτη που γεννήθηκαν στα ίδιαπά
νω κάτω χρόνια - ένα άλλο είναιτο Βέλγιο- με σκοπό να συγκε
ράσουν τις αποκλίνουσες απόψεις των μελών της σε ένα περι
φερειακό ζήτημα ,χωρίς ωστόσο να χρειαστεί προσφυγή σε πο
λεμικά μέσα.3 Στο μέτρο που ηΕυρωπαϊκή Συμφωνία λειτουρ
γούσε αποτελεσματικά , και αυτό συνέβαινε μέχρι τον Κριμαϊκό
Πόλεμο , όλα τα διακρατικά προβλήματα λύνονταν μέσω των
διαπραγματεύσεων των μελών της Συμφωνίας με τα υπόλοιπα
κράτη , ακόμη και τα στενά ενδιαφερόμενα , να μην έχουν παρά
περιορισμένο λόγο στα θέματα που τα αφορούσαν .
Οι δύο βασικές αρχές στις οποίες στηρίχθηκε ηλειτουργία
του διεθνούς συστήματος στα χρόνια αυτά ήταν αφενός μεν
η συνυπευθυνότητα των Μεγάλων Δυνάμεων για τη διατήρη
ση του εδαφικού status quo των συνθηκών του 1815 και για την
επίλυση των προβλημάτων που θα ανέκυπταν στην Ευρώπη ,
αφετέρου δε η συλλογικότητα των αποφάσεων ,μέσω συγκεκρι
μένων διπλωματικών πρακτικών , ως προς τιςτυχόν αλλαγές
που χρειάζονταν να γίνουν και τις λύσεις που έπρεπε να δοθούν .
Οι αρχές αυτές προέρχονταν από την αντίληψη ότιοι Μεγάλες
Δυνάμεις διέθεταν μοναδικά δικαιώματα και υποχρεώσεις
για τη διατήρηση της διεθνούς τάξης Ή ,όπως το έθετε ο άγ

(b)

Εικόνα 4.2: (Αριστερά) Στιγμιότυπο σελίδας από το πρώτο βιβλίο. (Δεξιά) Εξαγωγή του αντίστοιχου κειμένου με χρήση του εργαλείου OCR-Greek.

Έπειτα χρειάστηκε να γίνουν ορισμένες διορθώσεις διότι στις περισσότερες σελίδες τα κείμενα

που προέκυπταν περιείχαν ενωμένες ή κομμένες λέξεις. Για παράδειγμα μπορεί στην πραγματική εικόνα που δίναμε ως είσοδο να είχαμε τις λέξεις «είμαι εδώ» και το αποτέλεσμα που προέκυπτε να ήταν «είμαιεδώ». Επίσης έπρεπε να ενωθούν λέξεις οι οποίες χωρίζονταν με το σύμβολο της παύλας κάθε φορά που άλλαζε κάποια γραμμή (π.χ. η λέξη «ανά-σταση» να διορθωθεί σε «ανάσταση»). Οι παραπάνω διορθώσεις δεν ήταν απαραίτητες στα άλλα δύο βιβλία που εκφωνούν οι ομιλήτριες 3 και 4, διότι σε αυτή την περίπτωση τα pdf αρχεία που μας δόθηκαν είχαν καλύτερη μορφή και το κείμενο μπορούσε να προκύψει απευθείας χωρίς τη χρήση κάποιου online tool (π.χ. με απλή αντιγραφή ολόκληρου του κειμένου σε ένα αρχείο txt). Αφού λάβουμε τα κείμενα στη συνέχεια γίνεται μετατροπή των αριθμών και των συντομογραφιών σε λέξεις, όπου αυτό είναι δυνατό. Για παράδειγμα φράσεις όπως «ο κος Παπαδόπουλος γεννήθηκε το 1978» μετατρέπονταν στην φράση «ο κύριος Παπαδόπουλος γεννήθηκε το χίλια εννιακόσια εβδομήντα οκτώ». Στη συνέχεια αφαιρούμε τους χαρακτήρες που δεν βρίσκονται στο ελληνικό αλφάβητο και τα σημεία στίξης και μετατρέπουμε τις λέξεις ώστε να περιλαμβάνουν μόνο μικρά γράμματα (lowercase). Τέλος για να αποθηκεύσουμε τα κείμενα χρησιμοποιούμε την κωδικοποίηση ISO-8859-7. Αυτή η κωδικοποίηση είναι απαραίτητη προκειμένου τα κείμενα να δοθούν ως είσοδος στο open-source λογισμικό sail align. Το Sail Align [Kat+11] είναι ένα εργαλείο μέσω του οποίου επιτυγχάνεται το alignment (ευθυγράμμιση) μεταξύ ενός ηχητικού δείγματος ομιλίας και ενός κειμένου. Πρόκειται για μια επαναληπτική διαδικασία που στηρίζεται στην αναγνώριση φωνής. Πιο συγκεκριμένα, σε πρώτη φάση δίνεται ως είσοδος ένα ηχητικό δείγμα μαζί με ένα transcription. Το ηχητικό δείγμα χωρίζεται σε μικρότερα τμήματα από τα οποία εξάγονται ορισμένα ακουστικά χαρακτηριστικά και στη συνέχεια δίνονται ως είσοδος σε ένα σύστημα αναγνώρισης φωνής. Το σύστημα αυτό παράγει υποθετικά transcriptions για κάθε τμήμα του ηχητικού, τα οποία αφού συνενωθούν θα πρέπει να ευθυγραμμιστούν με το πραγματικό transcription. Έτσι το πρόβλημα του speech-text alignment μετατρέπεται σε ένα πρόβλημα ευθυγράμμισης κειμένων. Η διαδικασία αυτή ακολουθείται επαναληπτικά αφήνοντας σε κάβε βήμα εκτός τα τμήματα εκείνα από το ηχητικό δείγμα καθώς και από το αρχικό transcription, τα οποία έχουν ευθυγραμμιστεί σωστά (θα πρέπει να έχουν προβλεφθεί σωστά τουλάχιστον τρεις συνεχόμενες λέξεις). Επιπλέον γίνεται εφαρμογή ενός ακουστικού και γλωσσικού μοντέλου για να βελτιωθεί το alignment στην περίπτωση που το ηχητικό δείγμα είναι θορυβώδες. Η μέθοδος που ακολουθείται παρουσιάζεται στην Εικόνα 4.3. Στην περίπτωσή μας το




Εικόνα 4.3: Επαναληπτική μέθοδος που εφαρμόζεται από το Sail Align για την «ευθυγράμμιση» μεταξύ ηχητικών δειγμάτων ομιλίας και κειμένων. [Kat+11]

sail align χρησιμεύει διότι μπορούμε να επιτύχουμε alignment ακόμα και μεταξύ μεγάλων ηχητικών δειγμάτων και κειμένων. Έτσι λοιπόν σε πρώτη φάση χωρίζουμε τα ηχητικά δείγματα από όλα τα βιβλία με χρήση του προγράμματος Audacity, καθώς και τα κείμενα που έχουμε στη διάθεσή

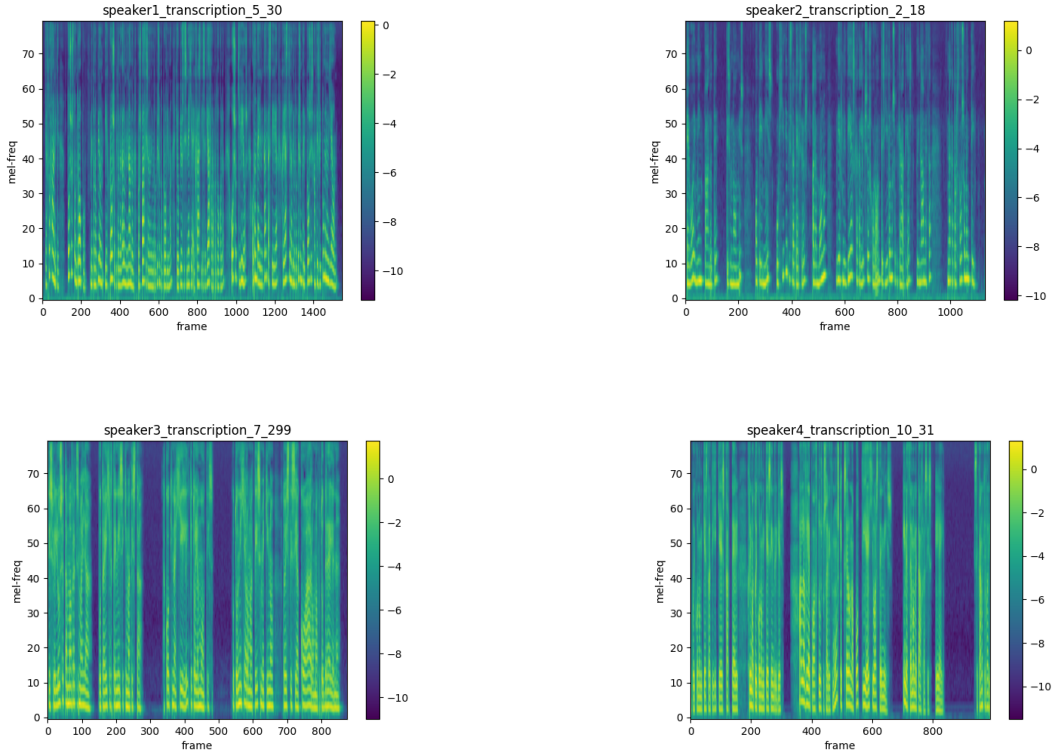
μας σε μικρότερα τμήματα, τα οποία δίνονται μαζί ως είσοδος στο sail align (ζεύγη audio.wav και transcription.txt). Θα πρέπει επίσης να έχουμε προσθέσει τις λέξεις εκείνες από τα κείμενά μας, οι οποίες δεν βρίσκονται στο ήδη υπάρχον λεξικό του sail align. Στη συνέχεια λαμβάνουμε το αποτέλεσμα σε ένα αρχείο που μας δίνει τους χρόνους διάρκειας σε δευτερόλεπτα κάθε λέξης μέσα στο ηχητικό δείγμα. Για παράδειγμα αν υπάρχει η πρόταση «καλημέρα τι κάνεις» μέσα στο κείμενο, τότε εντοπίζονται οι αντίστοιχοι χρόνοι στο ηχητικό δείγμα και το αποτέλεσμα έχει για παράδειγμα την εξής μορφή: $[1.5, 1.96] \rightarrow$ καλημέρα, $[1.96, 2.15] \rightarrow$ τι, $[2.15, 2.44] \rightarrow$ κάνεις. Αξίζει να σημειωθεί ότι για να λειτουργήσει αποτελεσματικά η μέθοδος θα πρέπει το ηχητικό δείγμα να περιλαμβάνει περίπου ολόκληρο το κείμενο που εκφωνείται μέσα σε αυτό. Αν παραδείγματος χάριν δοκιμάσουμε να δώσουμε ένα ηχητικό κλιπ διάρκειας μιας ώρας και ως transcription μόνο μια μικρή πρόταση από αυτό, τότε δε θα γίνει το σωστό alignment αφού όπως είναι αναμενόμενο οι λέξεις της πρότασης θα βρίσκονται και σε άλλα σημεία του ηχητικού και έτσι η ακριβής αναγνώριση του χρόνου τους θα είναι δυσκολότερη.

Αφού λάβουμε τα χρονικά διαστήματα των λέξεων στο κείμενο, σειρά έχει ο διαχωρισμός τόσο των ηχητικών δειγμάτων όσο και των κειμένων σε μικρότερα τμήματα. Για να διαχωρίσουμε λοιπόν ένα κείμενο σε μικρότερες φράσεις χρησιμοποιούμε τον greek tokenizer από τη βιβλιοθήκη nltk της python. Από τις φράσεις που προκύπτουν χωρίζουμε σε επιπλέον μικρότερα τμήματα εκείνες που έχουν πάνω από 40 λέξεις. Αυτό είναι χρήσιμο ώστε να μην υπάρχουν θέματα μνήμης στις GPUs και να μπορέσουμε να χρησιμοποιήσουμε μεγαλύτερο batch size κατά την εκπαίδευση των μοντέλων. Έπειτα λαμβάνουμε και τα αντίστοιχα τμήματα των ηχητικών δειγμάτων και για τις τέσσερις ομιλήτριες. Στη συνέχεια η διαδικασία που ακολουθείται για την εξαγωγή των φασματογραφημάτων στην κλίμακα mel αλλά και των υπόλοιπων filelists (mel-text, audio-mel) είναι ίδια με τη διαδικασία που ακολουθήθηκε για τα ισπανικά δεδομένα. Για κάθε speaker εξάγουμε τα mel spectrograms με τη μέθοδο STFT χρησιμοποιώντας ίδιες παραμέτρους με πριν και τέλος διαχωρίζουμε τα δεδομένα σε train, validation και test sets με την αναλογία που αναφέραμε στην προηγούμενη ενότητα. Στην Εικόνα 4.4 παρουσιάζονται ορισμένα φασματογραφήματα που αντιστοιχούν σε ηχητικά δείγματα από τις τέσσερις ομιλήτριες στην ελληνική γλώσσα. Εφόσον τα δεδομένα μας έχουν έρθει στην επιθυμητή μορφή σειρά έχει η εκπαίδευση των μοντέλων στην ισπανική και ελληνική γλώσσα.

## 4.4 Εκπαίδευση των μοντέλων

Στην ενότητα αυτή περιγράφουμε τη διαδικασία που ακολουθήθηκε για την εκπαίδευση των μοντέλων σύνθεσης φωνής από κείμενο. Στηριζόμενοι στην υλοποίηση της nvidia[1] που συνδυάζει τα μοντέλα Tacotron2 και WaveGlow, εκτελέσαμε ορισμένα πειράματα χρησιμοποιώντας τα δεδομένα από την ισπανική και ελληνική γλώσσα. Τα πειράματά μας χωρίζονται σε δύο φάσεις. Η πρώτη φάση αφορά την εκπαίδευση του μοντέλου Tacotron2 προκειμένου να μετατρέψουμε ένα κείμενο (ακολουθία χαρακτήρων) στο αντίστοιχο φασματογράφημα στην κλίμακα mel. Η δεύτερη φάση αφορά την εκπαίδευση του μοντέλου WaveGlow ώστε να λάβουμε το ηχητικό δείγμα από το φασματογράφημα που αντιστοιχεί στο κείμενο εισόδου. Όσον αφορά την εκπαίδευση, χρησιμοποιήθηκαν δύο NVIDIA GeForce RTX 2080 Ti GPUs στον server "milos" του ΙΕΛ.

[1] https://github.com/NVIDIA/DeepLearningExamples/tree/master/PyTorch/SpeechSynthesis/Tacotron2




Εικόνα 4.4: Φασματογραφήματα στην κλίμακα mel για ηχητικά δείγματα από τις τέσσερις ομιλήτριες στα ελληνικά.

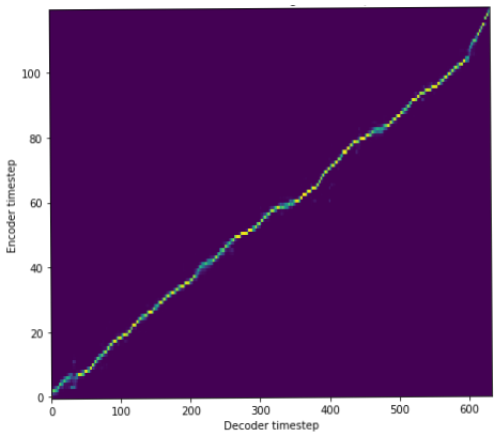
### 4.4.1 Tacotron2

#### Πείραμα 1: Εκπαίδευση στα ελληνικά δεδομένα

Ξεκινώντας από το Tacotron2 η πρώτη ιδέα ήταν να εκπαιδεύσουμε το συγκεκριμένο μοντέλο απευθείας πάνω στα ελληνικά δεδομένα. Συνήθως τα μοντέλα σύνθεσης φωνής από κείμενο εκπαιδεύονται με ηχογραφήσεις από έναν μόνο speaker, όπως για παράδειγμα στην περίπτωση των μοντέλων Tacotron2 και WaveGlow για την αγγλική γλώσσα, τα οποία εκπαιδεύτηκαν στο σύνολο δεδομένων LJ Speech που περιέχει ηχογραφήσεις μόνο από μια ομιλήτρια. Αναλύοντας τα ηχητικά δείγματα από όλες τις ομιλήτριες στα ελληνικά, είδαμε ότι η τέταρτη είχε τις ηχογραφήσεις με την καλύτερη ποιότητα. Επιπλέον το σύνολο των διαθέσιμων ωρών για την τέταρτη ομιλήτρια ήταν το μεγαλύτερο (6.48 ώρες) συγκριτικά με τις υπόλοιπες τρεις. Παρ' όλ' αυτά το αποτέλεσμα από το μοντέλο Tacotron2 δεν ήταν ικανοποιητικό και ο σημαντικότερος λόγος είναι ότι χρησιμοποιήσαμε αρκετά λίγες ώρες ηχητικών δειγμάτων για την εκπαίδευση. Ενδεικτικά αναφέρουμε ότι το LJ Speech περιέχει 24 ώρες ηχογραφήσεων, οπότε χρησιμοποιήθηκε λιγότερο από το 30% των συνολικών ωρών (συγκριτικά με το σύνολο LJ Speech). Για το λόγο αυτό επιλέξαμε στο δεύτερο πείραμά μας να εκπαιδεύσουμε το μοντέλο Tacotron2 αξιοποιώντας τα δεδομένα από όλες τις ομιλήτριες στα ελληνικά (14.16 ώρες). Και σε αυτή την περίπτωση δεν είχαμε το επιθυμητό αποτέλεσμα, γεγονός που επίσης οφείλεται στην έλλειψη επαρκών δεδομένων τόσο όσον αφορά τις διαθέσιμες ώρες ηχητικών δειγμάτων όσο και στο ότι το μοντέλο εκπαιδεύτηκε με δεδομένα από πολλούς ομιλητές.

Για το πρώτο πείραμα το μοντέλο εκπαιδεύτηκε για 990 εποχές με batch size 28, ρυθμό εκμάθησης (learning rate) $10^{-3}$ και βελτιστοποιητή (optimizer) Adam [KB14]. Επιπλέον ο ρυθμός

εκμάθησης μειώθηκε σε $10^{-4}$ στις 500 εποχές. Ομοίως για το δεύτερο πείραμα το μοντέλο εκπαιδεύτηκε με χρήση του βελτιστοποιητή Adam με weight decay $10^{-6}$ (παράγοντας ομαλοποίησης L2) για 1500 εποχές, βήμα εκμάθησης $10^{-3}$ που μειώθηκε σε $10^{-4}$ και $10^{-5}$ στις 500 και στις 1000 εποχές αντίστοιχα. Το batch size σε αυτή την περίπτωση ήταν ίσο με 26. Ο συνολικός χρόνος εκπαίδευσης για το πρώτο πείραμα ήταν περίπου 4 μέρες ενώ για το δεύτερο περίπου 17 μέρες. Αφού ολοκληρώθηκε η εκπαίδευση του μοντέλου, στη συνέχεια πήραμε ορισμένα αποτελέσματα λαμβάνοντας υπόψιν τα βάρη του στην εποχή με το μικρότερο validation loss. Έπειτα εξετάσαμε τα γραφήματα των alignments που προέκυψαν στις δύο περιπτώσεις. Τα alignments περιγράφουν το πόσο καλά αντιστοιχίζονται τα frames που εξάγει ο decoder με τους χαρακτήρες από το κείμενο εισόδου. Όπως είδαμε στο Κεφάλαιο 2, το μοντέλο Tacotron2 περιέχει το μηχανισμό προσοχής location sensitive attention που παράγει ορισμένα alignment scores[2] για κάθε διάνυσμα αναπαράστασης που προκύπτει από τον encoder. Τα scores αυτά δείχνουν σε ποιο encoder state πρέπει να εστιάσει την προσοχή του ο decoder σε ένα συγκεκριμένο timestep, προκειμένου να εξάγει το αντίστοιχο frame από το φασματογράφημα. Έτσι λοιπόν τα alignments σχηματίζουν έναν πίνακα με τόσες γραμμές όσοι και οι χαρακτήρες του κειμένου εισόδου (όσα είναι δηλαδή και τα encoder states) και στήλες όσο και το πλήθος των frames που παράγει ο decoder. Συνεπώς το στοιχείο που ανήκει στην γραμμή $i$ και τη στήλη $j$ περιγράφει το πόσο καλά γίνεται aligned ο χαρακτήρας που αντιστοιχεί στη θέση $i$ με το frame που αντιστοιχεί στη θέση $j$. Προφανώς για να είναι το αποτέλεσμα ορθό θα πρέπει το τελικό γράφημα των alignments να παρουσιάζει μια διαγώνιο, όπως φαίνεται και στην Εικόνα 4.5. Αυτό σημαίνει ότι κάθε επόμενο frame που



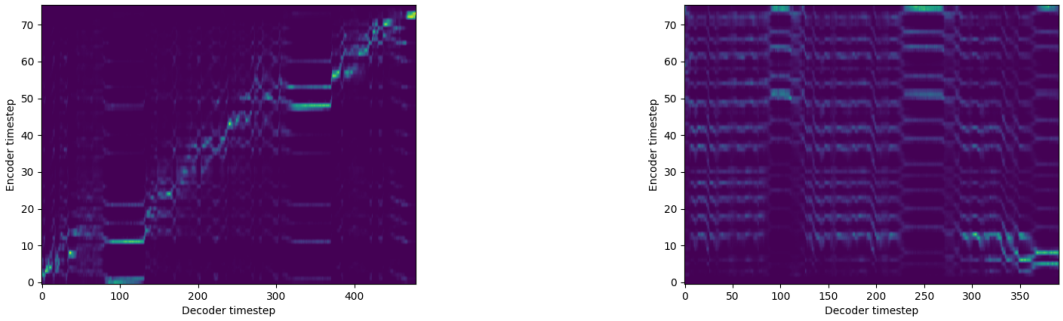

Εικόνα 4.5: Παράδειγμα γραφήματος ενός ορθού alignment από το μοντέλο Tacotron2.

εξάγεται θα πρέπει ιδανικά να είναι aligned με κάποιον από τους αμέσως επόμενους χαρακτήρες του κειμένου και όχι με κάποιον προηγούμενο.

Στην Εικόνα 4.6 παρατηρούμε τα alignments που προέκυψαν από την εκπαίδευση του μοντέλου Tacotron2 στα ελληνικά, αριστερά για την τέταρτη ομιλήτρια και δεξιά για όλες τις ομιλήτριες. Μπορούμε εύκολα να διακρίνουμε ότι το μοντέλο και στις δύο περιπτώσεις δεν μπορεί να κάνει align τους χαρακτήρες με τα αντίστοιχα frames. Το πρόβλημα είναι σαφώς εντονότερο στη δεξιά εικόνα, ενώ στην αριστερή φαίνεται μια ένδειξη για σχετικά διαγώνια μορφή που όμως δεν είναι επαρκής.

Επιπλέον στην Εικόνα 4.7 φαίνονται τα φασματογραφήματα στην κλίμακα mel για το ίδιο κείμενο εισόδου. Τα πρώτα δύο (speaker 4 και all speakers) προκύπτουν από το μοντέλο Tacotron2,

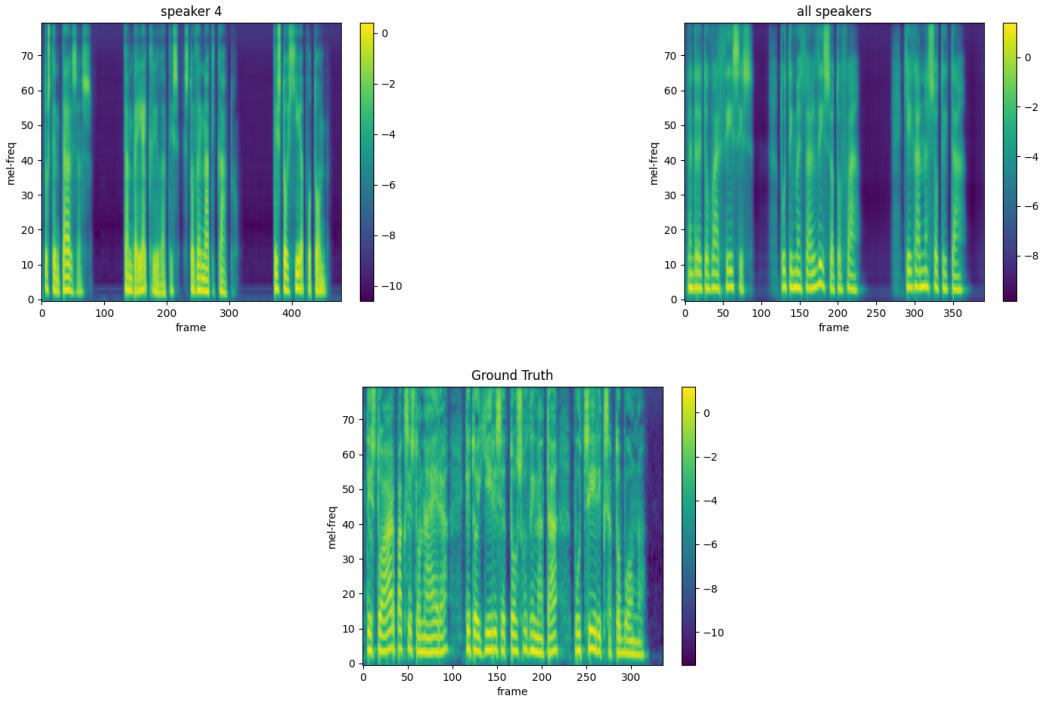
[2]πρόκειται για το attention context vector




(a) speaker 4 (b) all speakers

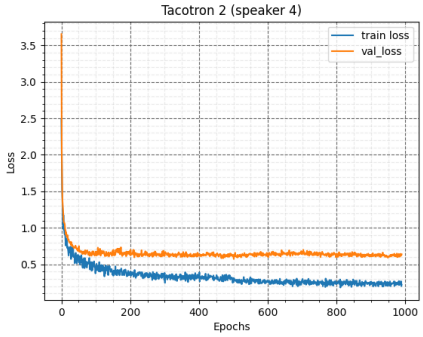
Εικόνα 4.6: Τα alignments που προκύπτουν από το μοντέλο Tacotron2 ύστερα από εκπαίδευση στα ελληνικά δεδομένα. Αριστερά φαίνονται τα alignments που προέκυψαν από την τέταρτη ομιλήτρια και δεξιά από όλες τις ομιλήτριες για μία συγκεκριμένη φράση.

ενώ κάτω παρουσιάζεται το πραγματικό φασματογράφημα (Ground Truth). Παρατηρούμε ότι υπάρχει σημαντική διαφορά όσον αφορά τις προβλέψεις του μοντέλου και του πραγματικού φασματογραφήματος. Συγκεκριμένα και στα δύο φασματογραφήματα που προκύπτουν από το μοντέλο φαίνονται δύο περιοχές με βαθύ μωβ χρώμα (που αντιστοιχεί σε πολύ χαμηλή ένταση για όλες τις συχνότητες - άρα και παύση σε εκείνα τα σημεία), οι οποίες δεν υπάρχουν στο αντίστοιχο ground truth φασματογράφημα.

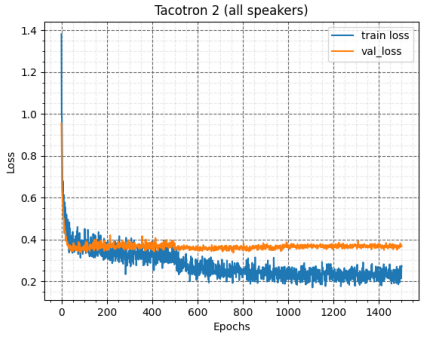



Εικόνα 4.7: Τα φασματογραφήματα στην κλίμακα mel από το μοντέλο Tacotron2 για την τέταρτη ομιλήτρια (speaker 4), για όλες μαζί (all speakers), καθώς και το πραγματικό (Ground Truth) φασματογράφημα για μία συγκεκριμένη φράση.

Τα παραπάνω αποτελέσματα είναι αναμενόμενα αν παρατηρήσουμε και το Διάγραμμα 4.8 που απεικονίζει τα σφάλματα εκπαίδευσης και επικύρωσης του μοντέλου Tacotron2 για τις δύο περιπτώσεις που εξετάζουμε. Είναι εμφανές ότι το μοντέλο παρουσιάζει το φαινόμενο της υπερεκπαίδευσης (overfitting) μετά από ένα συγκεκριμένο αριθμό εποχών αφού υπάρχει κενό μεταξύ των σφαλμάτων train και validation. Οι τιμές των σφαλμάτων για ορισμένες εποχές εκπαίδευσης φαίνονται αναλυτικά στους Πίνακες 4.2 και 4.3.




(a) speaker 4




(b) all speakers

Διάγραμμα 4.8: Σφάλματα εκπαίδευσης (train) και επικύρωσης (validation) του μοντέλου Tacotron2 για τις περιπτώσεις της τέταρτης ομιλήτριας (αριστερά) και όλων μαζί (δεξιά).

| Epoch | Train loss | Val loss |
|---|---|---|
| 1 | 1.94 | 3.65 |
| 250 | 0.37 | 0.63 |
| 500 | 0.31 | 0.60 |
| 750 | 0.25 | 0.65 |

Πίνακας 4.2: speaker 4

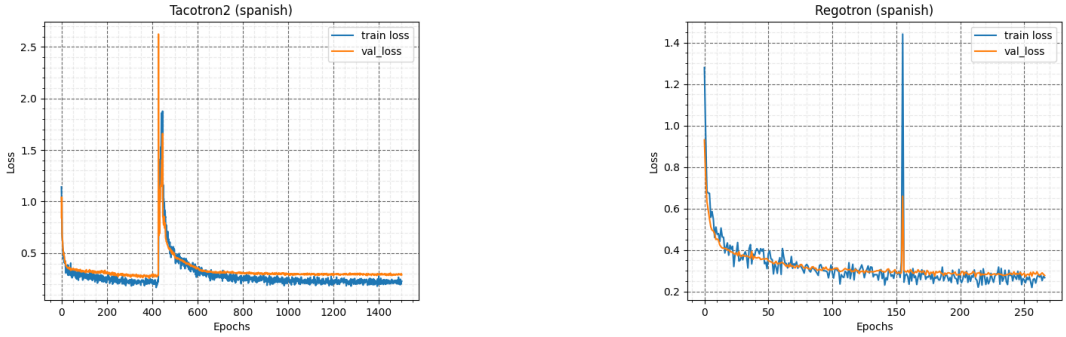
| Epoch | Train loss | Val loss |
|---|---|---|
| 1 | 0.99 | 0.85 |
| 250 | 0.29 | 0.39 |
| 500 | 0.30 | 0.35 |
| 750 | 0.22 | 0.35 |

Πίνακας 4.3: all speakers

### Πείραμα 2: Εκπαίδευση στα ισπανικά δεδομένα

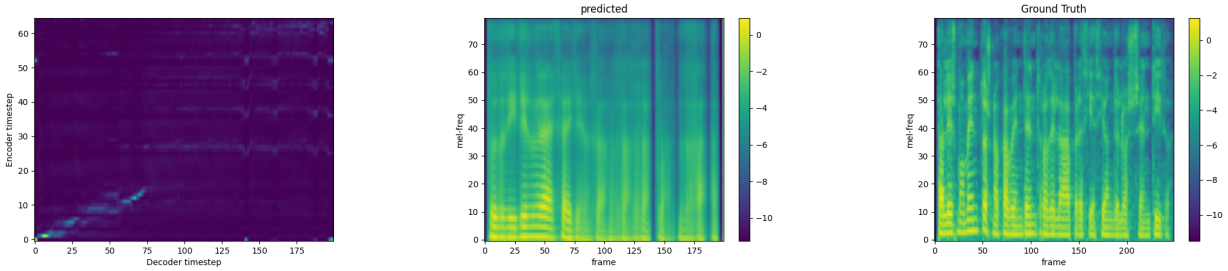
Όπως είδαμε η εκπαίδευση του μοντέλου Tacotron2 χρησιμοποιώντας μόνο ελληνικά δεδομένα είτε από μια ομιλήτρια είτε από όλες μαζί δεν έδωσε τα επιθυμητά αποτελέσματα. Για το λόγο αυτό σαν δεύτερο πείραμα δοκιμάσαμε να εκπαιδεύσουμε πρώτα το μοντέλο στα ισπανικά, όπου διαθέτουμε περισσότερες ώρες από έναν μόνο ομιλητή και στη συνέχεια να αξιοποιήσουμε την τεχνική της μεταφοράς μάθησης για τα ελληνικά δεδομένα. Η επιλογή της ισπανικής γλώσσας οφείλεται στο γεγονός ότι υπάρχει μεγαλύτερη ομοιότητα με την ελληνική όσον αφορά τον τρόπο προφοράς και εκφώνησης των λέξεων. Κατά συνέπεια αναμένουμε η μεταφορά μάθησης να έχει καλύτερα αποτελέσματα συγκριτικά με άλλες γλώσσες, όπως για παράδειγμα τα αγγλικά ή τα γερμανικά. Χρησιμοποιώντας λοιπόν παρόμοιο setup με πριν, δηλαδή μέγεθος batch size 26, ρυθμό εκμάθησης $10^{-3}$, βελτιστοποιητή Adam με παράγοντα ομαλοποίησης L2 ίσο με $10^{-6}$, το μοντέλο εκπαιδεύτηκε για 1500 εποχές. Ο συνολικός χρόνος εκπαίδευσης ήταν περίπου 16 μέρες και στις δύο GPUs. Στο Διάγραμμα 4.9 (αριστερά) παρουσιάζονται τα σφάλματα εκπαίδευσης και επικύρωσης για το μοντέλο Tacotron2 στα ισπανικά δεδομένα. Παρατηρούμε ότι περίπου στην εποχή 420 υπάρχει μια

κατακόρυφη αύξηση στις τιμές των δύο σφαλμάτων. Στη συνέχεια οι τιμές τους επανέρχονται, όμως τελικά το μοντέλο δεν οδηγείται σε σύγκλιση. Αυτό είναι εμφανές και από την Εικόνα 4.10,




Διάγραμμα 4.9: Σφάλματα εκπαίδευσης (train) και επικύρωσης (validation) του μοντέλου Tacotron2 (αριστερά) και του μοντέλου Regotron (δεξιά) στα ισπανικά δεδομένα.

που όπως φαίνεται ούτε το alignment, αλλά ούτε και η πρόβλεψη για το φασματογράφημα είναι ικανοποιητική.




Εικόνα 4.10: (Αριστερά) alignments, (κέντρο) φασματογράφημα από το μοντέλο Tacotron2, (δεξιά) ground truth φασματογράφημα για μια φράση στα ισπανικά.

Για το λόγο αυτό δοκιμάσαμε να εκπαιδεύσουμε μια παραλλαγή του μοντέλου Tacotron2 που ονομάζεται Regotron [Geo+22]. Σύμφωνα με τους συγγραφείς, το regotron μπορεί να επιλύσει θέματα αστάθειας κατά την εκπαίδευση και να βελτιώσει το alignment, προσθέτοντας έναν επιπλέον όρο ομαλοποίησης (regularization) στη συνάρτηση ελαχιστοποίησης του Tacotron2. Πιο συγκεκριμένα αναφέρεται ότι η εύρεση ορθών alignments από το μηχανισμό προσοχής του μοντέλου έχει μεγάλη σημασία τόσο στην ομαλότερη εκπαίδευση, όσο και στην παραγωγή φασματογραφημάτων των οποίων τα αντίστοιχα παραγόμενα ηχητικά δείγματα ακούγονται πιο φυσικά. Η βασική ιδέα στηρίζεται στην ομαλοποίηση του σφάλματος εκπαίδευσης με έναν όρο που λαμβάνει υπόψιν τη διαφορά των γειτονικών alignments. Συγκεκριμένα για κάθε frame $j$ ορίζεται η ποσότητα:

$$\langle \boldsymbol{\alpha_j} \rangle = \sum_{i=1}^{N} \alpha_{i,j} \cdot i, \tag{4.4.1}$$

που ονομάζεται "mean attended position" και εκφράζει που περίπου δίνεται βαρύτητα στους χαρακτήρες του κειμένου για την εξαγωγή του frame $j$. Όπως είναι λογικό θα πρέπει να ισχύει η συνθήκη αύξουσας μονοτονίας μεταξύ δύο διαδοχικών θέσεων, δηλαδή να έχουμε $\langle \boldsymbol{\alpha_{j+1}} \rangle \geq \langle \boldsymbol{\alpha_j} \rangle$ για κάθε

frame $j = 1, \ldots, M$. Λαμβάνοντας υπόψιν τον παραπάνω συλλογισμό, ο όρος ομαλοποίησης του μοντέλου Regotron γράφεται ως:

$$L_A = \sum_{j=1}^{M-1} \max \left\{ \frac{\langle \boldsymbol{\alpha_j} \rangle - \langle \boldsymbol{\alpha_{j+1}} \rangle + \delta \frac{N}{M}}{N}, 0 \right\}, \tag{4.4.2}$$

όπου $N$ είναι το πλήθος των χαρακτήρων στο κείμενο εισόδου και $\delta$ ο παράγοντας που ελέγχει το πόσο θα εφαρμοστεί η ποινή για τη συνθήκη μονοτονίας. Για παράδειγμα μεγαλύτερη τιμή του παράγοντα $\delta$ σημαίνει και μεγαλύτερη ποινή όταν δεν ικανοποιείται η μονοτονία. Αν τελικά ο αριθμητής σε κάποιον όρο του αθροίσματος είναι αρνητικός, τότε ο όρος αυτός λαμβάνει την τιμή 0 και παραλείπεται από το συνολικό άθροισμα. Έτσι λοιπόν δίνεται ποινή μόνο στους όρους εκείνους που δεν ικανοποιούν τη συνθήκη μονοτονίας. Τελικά η νέα συνάρτηση ελαχιστοποίησης για το μοντέλο Regotron θα γράφεται ως:

$$L_R = L_T + \lambda L_A, \tag{4.4.3}$$

όπου το $\lambda$ είναι ο παράγοντας ομαλοποίησης. Στην πράξη αντί για τον παράγοντα $\delta$ ρυθμίζεται η τιμή του $\lambda$.

Όσον αφορά τα πειράματά μας, επιλέχθηκαν οι τιμές $\delta = 0.01$ και $\lambda = 10^{-4}$. Το μοντέλο εκπαιδεύτηκε χρησιμοποιώντας τις ίδιες παραμέτρους όπως το Tacotron2 στα ισπανικά για 267 εποχές (περίπου 3 μέρες). Το αποτέλεσμα φαίνεται στο Διάγραμμα 4.9 (δεξιά), όπου και σε αυτή την περίπτωση λίγο μετά την εποχή 150 παρατηρείται κατακόρυφη αύξηση στις τιμές των σφαλμάτων. Έτσι λοιπόν παραμένει το πρόβλημα της αστάθειας κατά την εκπαίδευση ακόμη και με τη χρήση του μοντέλου Regotron. Το γεγονός αυτό ενδεχομένως να μπορεί να βελτιωθεί με την επιλογή καλύτερων υπερπαραμέτρων στο μοντέλο Regotron, είτε με καλύτερη προεπεξεργασία των ηχητικών δειγμάτων. Γενικότερα παρατηρήσαμε ότι στα δεδομένα που λάβαμε στα ισπανικά τα ηχητικά δείγματα «κόβονταν» στο τέλος της πρότασης, δηλαδή η τελευταία λέξη (σύμφωνα με το αντίστοιχο transcription) έλειπε είτε ολόκληρη, είτε ένα μέρος αυτής από το ηχητικό δείγμα. Αυτό όπως είναι λογικό μπορεί να οδηγήσει σε προβλήματα αναφορικά με την ορθότητα των alignments και συνεπώς το μοντέλο να μην οδηγείται σε σύγκλιση.

### 4.4.2 WaveGlow

Ένα σημαντικό μέρος των πειραμάτων αφορά επίσης την εκπαίδευση του μοντέλου WaveGlow, το οποίο χρησιμοποιήθηκε ως vocoder για την μετατροπή των φασματογραφημάτων στο αντίστοιχο ηχητικό δείγμα. Όπως και στο μοντέλο Tacotron2, για την εκπαίδευση αξιοποιήσαμε τα ισπανικά δεδομένα συνολικής διάρκειας περίπου 18 ώρες. Το μοντέλο WaveGlow εκπαιδεύτηκε για 460 εποχές με βήμα εκμάθησης $10^{-4}$, batch size 2 και βελτιστοποιητή Adam χωρίς ομαλοποίηση για τα βάρη του δικτύου. Παρατηρούμε ότι το batch size είναι πολύ μικρότερο συγκριτικά με το αντίστοιχο batch size στο μοντέλο Tacotron2 (26 ή 32) και αυτό συμβαίνει διότι τώρα το μοντέλο δέχεται ως είσοδο τόσο το mel spectrogram όσο και ολόκληρη την κυματομορφή ήχου (audio). Συνεπώς το batch size χρειάζεται να μειωθεί αρκετά, ώστε να μην υπάρχουν θέματα μνήμης. Ο συνολικός χρόνος εκπαίδευσης για το μοντέλο WaveGlow και στις δύο GPUs ήταν πάνω από ένα μήνα. Όσον αφορά την αντίστοιχη υλοποίηση της nvidia στην Εικόνα 4.11 παρουσιάζονται οι αναμενόμενοι χρόνοι εκπαίδευσης για τη σύγκλιση του μοντέλου WaveGlow με χρήση δύο ειδών GPUs. Αριστερά φαίνονται οι χρόνοι εκπαίδευσης σε 1,4 και 8 NVIDIA V100 16GB gpus ενώ δεξιά σε 1,4 και 8 NVIDIA A100 40GB gpus. Παρατηρείται ότι στην περίπτωση των

τεσσάρων V100 GPUs ο χρόνος εκπαίδευσης είναι σχεδόν 10 μέρες (233 ώρες) ενώ στην αντίστοιχη περίπτωση των τεσσάρων A100 GPUs είναι περίπου 5 μέρες (122) ώρες. Κατανοούμε λοιπόν ότι για το συγκεκριμένο μοντέλο απαιτείται αρκετά μεγάλος χρόνος εκπαίδευσης αλλά και μεγάλη υπολογιστική ισχύς.

| Number of GPUs | Batch size per GPU | Time to train with mixed precision (Hrs) | Time to train with FP32 (Hrs) | Speed-up with mixed precision |
|---|---|---|---|---|
| 1 | 10@FP16, 4@FP32 | 249 | 793 | 3.18 |
| 4 | 10@FP16, 4@FP32 | 78 | 233 | 3.00 |
| 8 | 10@FP16, 4@FP32 | 48 | 127 | 2.98 |

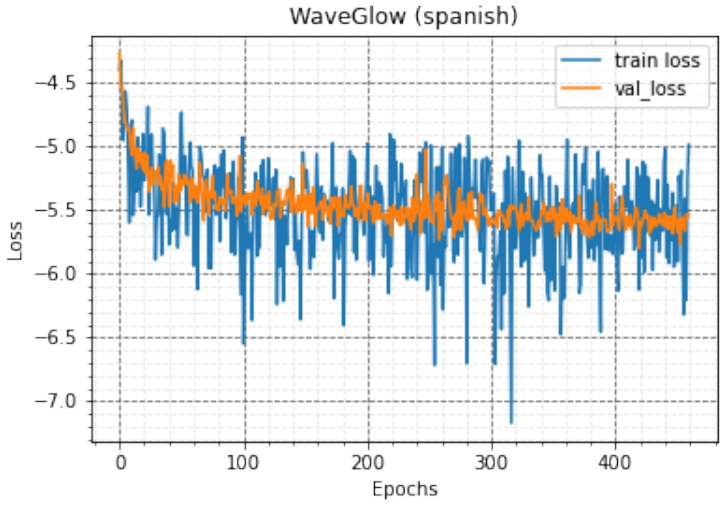
(a) NVIDIA V100 16GB

| Number of GPUs | Batch size per GPU | Time to train with mixed precision (Hrs) | Time to train with TF32 (Hrs) | Speed-up with mixed precision |
|---|---|---|---|---|
| 1 | 10@FP16, 4@TF32 | 188 | 416 | 2.21 |
| 4 | 10@FP16, 4@TF32 | 54 | 122 | 2.27 |
| 8 | 10@FP16, 4@TF32 | 33 | 75 | 2.29 |

(b) NVIDIA A100 40GB

Εικόνα 4.11: Αναμενόμενοι χρόνοι εκπαίδευσης για τη σύγκλιση (1001 εποχές) του μοντέλου WaveGlow σε δύο διαφορετικές GPUs. [nvidia-waveglow]

Παρ' όλα αυτά ύστερα από την εκπαίδευση του μοντέλου σε μόλις μισές εποχές από τη σύγκλιση, λάβαμε ορισμένα αρχικά αποτελέσματα όσον αφορά τον παραγόμενο ήχο. Στο Διάγραμμα 4.12 παρουσιάζονται τα σφάλματα εκπαίδευσης και επικύρωσης για το μοντέλο WaveGlow στα ισπανικά δεδομένα. Παρατηρούμε ότι το μοντέλο εκπαιδεύεται ομαλά χωρίς ιδιαίτερα προβλή-




Διάγραμμα 4.12: Σφάλματα εκπαίδευσης (train) και επικύρωσης (validation) για το μοντέλο WaveGlow στα ισπανικά δεδομένα.

ματα αστάθειας. Ήδη από την εποχή 460, δοκιμάζοντας το μοντέλο για inference πάνω σε ground truth φασματογραφήματα που εξάγαμε από τα ηχητικά δείγματα στα ισπανικά, είδαμε ότι ο παραγόμενος ήχος ακούγεται σχετικά καλά αλλά ελαφρώς «ρομποτικά» και σίγουρα επιδέχεται περαιτέρω βελτίωση. Επιπλέον δοκιμάσαμε να κάνουμε inference χρησιμοποιώντας το συγκεκριμένο μοντέλο απευθείας πάνω σε ground truth φασματογραφήματα από τα ελληνικά. Ο παραγόμενος ήχος και σε αυτή την περίπτωση ήταν σχετικά καλός, δηλαδή η φράση που προέκυπτε έμοιαζε με την φράση που περιείχε το ηχητικό δείγμα από το οποίο είχε εξαχθεί το φασματογράφημα. Εδώ να σημειώσουμε ότι τα ισπανικά περιέχουν ηχητικά δείγματα που προέρχονται μόνο από έναν άντρα ομιλητή, ενώ τα ελληνικά από τέσσερις γυναίκες ομιλήτριες. Η παρατήρηση αυτή, ότι δηλαδή το μοντέλο WaveGlow μπορεί να παράγει σχετικά καλά ηχητικά δείγματα και στις

δύο γλώσσες ακόμα και από ένα αρχικό στάδιο της εκπαίδευσης, μας οδήγησε στο συμπέρασμα ότι το WaveGlow είναι ανεξάρτητο τόσο από τη γλώσσα ομιλίας όσο και από την ταυτότητα του ομιλητή. Για το λόγο αυτό αντί να συνεχίσουμε την εκπαίδευση του μοντέλου μέχρι τη σύγκλιση, κάτι που θα απαιτούσε πολύ παραπάνω χρόνο, αξιοποιήσαμε το checkpoint του μοντέλου που είχε εκπαιδευτεί στην αγγλική γλώσσα. Δοκιμάζοντας το «αγγλικό» μοντέλο με τα συγκεκριμένα βάρη σε φασματογραφήματα από τα ελληνικά δεδομένα, είδαμε ότι η ποιότητα του παραγόμενου ήχου ήταν αρκετά καλή. Εδώ θα πρέπει να σημειωθεί ότι ο ρυθμός δειγματοληψίας των ηχητικών δειγμάτων από την αγγλική γλώσσα, στα οποία εκπαιδεύτηκε το μοντέλο WaveGlow και λάβαμε το συγκεκριμένο checkpoint, ήταν στα 22050 Hz. Για τα ελληνικά και τα ισπανικά δεδομένα ο ρυθμός δειγματοληψίας ήταν στα 16000 Hz. Επίσης για την εξαγωγή των φασματογραφημάτων στα αγγλικά με τη μέθοδο STFT χρησιμοποιήθηκαν οι εξής παράμετροι: filter length = 1024, hop length = 256, window length = 1024 ενώ για τα ελληνικά και τα ισπανικά οι αντίστοιχες τιμές ήταν filter length = 1024, hop length = 200 και window length = 800. Στη συνέχεια των πειραμάτων λάβαμε υπόψιν και αυτές τις διαφοροποιήσεις προκειμένου να μπορεί να χρησιμοποιηθεί αποτελεσματικά το προεκπαιδευμένο «αγγλικό» μοντέλο WaveGlow για την παραγωγή ηχητικών δειγμάτων στα ελληνικά.

## 4.5 Μεταφορά Μάθησης

Από τα πειράματα που έχουμε εκτελέσει μέχρι στιγμής βλέπουμε ότι το μοντέλο WaveGlow μπορεί να χρησιμοποιηθεί ως vocoder σε ένα σύστημα σύνθεσης φωνής από κείμενο για τα ελληνικά. Έτσι θα πρέπει να εστιάσουμε στο πρώτο μέρος του συστήματος που θα παράγει το φασματογράφημα στην κλίμακα mel από το κείμενο εισόδου. Εφόσον τα πειράματα για το μοντέλο Tacotron2 δεν έχουν δώσει μέχρι στιγμής κάποιο επιθυμητό αποτέλεσμα, αποφασίσαμε να χρησιμοποιήσουμε την τεχνική της μεταφοράς μάθησης από ένα προεκπαιδευμένο μοντέλο Tacotron2 και στη συνέχεια να κάνουμε fine tuning στα ελληνικά δεδομένα. Ύστερα από αναζήτηση καταλήξαμε στο μοντέλο Catotron [Kül+20], το οποίο αποτελεί μια παραλλαγή του Tacotron2 που έχει εκπαιδευτεί στα καταλανικά. Τα καταλανικά είναι μια γλώσσα που έχει πολλά κοινά με τα ισπανικά, οπότε αναμένουμε η μεταφορά μάθησης από τα καταλανικά στα ελληνικά να παράγει ικανοποιητικά αποτελέσματα λόγω του ότι και αυτές οι δύο γλώσσες θα μοιάζουν μεταξύ τους. Όσον αφορά το Catotron, οι συγγραφείς αναφέρουν ότι εκπαιδεύτηκε με ηχητικά δεδομένα που συλλέχθηκαν από το διαδίκτυο και τα οποία προέρχονταν από κοινοβουλευτικές συνεδριάσεις της καταλανικής κυβέρνησης (ParlamentParla). Από τα συγκεκριμένα ηχητικά δείγματα επιλέχθηκε ένα υποσύνολο αυτών που περιείχε ηχητικά δείγματα από έναν άντρα ομιλητή. Επιπλέον χρησιμοποίησαν και δύο ομιλητές από το FestCat corpus [Bon+08], μία γυναίκα και έναν άντρα με 10 ώρες ηχογραφήσεων ο καθένας. Έτσι λοιπόν εκπαιδεύτηκαν τρία μοντέλα για κάθε έναν ομιλητή. Επειδή και σε αυτή την περίπτωση το πλήθος των διαθέσιμων ηχογραφήσεων δεν επαρκούσε για πλήρη εκπαίδευση από την αρχή, έγινε μεταφορά μάθησης από το μοντέλο Tacotron2 που είχε εκπαιδευτεί στην αγγλική γλώσσα. Για την εκπαίδευση του Catotron (και για τους τρεις ομιλητές) χρησιμοποιήθηκε βήμα εκμάθησης $10^{-3}$, batch size 64 και αυξήθηκε η τιμή του dropout στα LSTM layers του decoder από 0.1 σε 0.3. Το μοντέλο τελικά συνέκλινε ύστερα από περίπου 40 εποχές. Ως vocoders δοκιμάστηκαν τα μοντέλα WaveGlow και MelGAN. Τα αποτελέσματα καθώς και τα αντίστοιχα checkpoints από τα μοντέλα στα καταλανικά βρίσκονται στη σελίδα Catotron. Για την περίπτωσή μας επιλέξαμε το checkpoint που αντιστοιχεί στη γυναίκα ομιλήτρια Ona αφού τα ελληνικά δεδομένα που διαθέτουμε αποτελούνται αποκλειστικά από γυναίκες ομιλήτριες. Στα παρακάτω πειράματα έγινε επιλογή διαφορετικών παραμέτρων όσον αφορά την εξαγωγή των φασματογραφημάτων στα ελλη-

νικά, ώστε να είναι ίδιες με τις αντίστοιχες που είχαν χρησιμοποιηθεί στο μοντέλο Tacotron2 για την αγγλική γλώσσα. Συγκεκριμένα χρειάστηκε να γίνει upsampling των ηχητικών μας δειγμάτων από τα 16000 Hz στα 22050 Hz. Επίσης για τη μέθοδο STFT χρησιμοποιήθηκαν οι παράμετροι filter length 1024, hop length 256, window length 1024, mel-fmin 0 Hz, mel-fmax 8000 Hz και n-mels 80. Για την εκπαίδευση των μοντέλων χρησιμοποιήθηκε μια μόνο GPU. Το batch size ήταν 32, ο ρυθμός εκμάθησης $10^{-3}$ και για την εκμάθηση των παραμέτρων χρησιμοποιήθηκε ο βελτιστοποιητής Adam με παράμετρο ποινής (weight decay) ίση με $10^{-6}$.

**Πείραμα 3: Catotron Fine Tuning στα ελληνικά**

Αφού λάβουμε το checkpoint από το μοντέλο Catotron στη συνέχεια δοκιμάζουμε να κάνουμε fine tuning στα ελληνικά δεδομένα από όλες τις ομιλήτριες. Σε πρώτη φάση χρειάστηκε να γίνει μετατροπή όλων των ελληνικών χαρακτήρων σε χαρακτήρες ascii (greeklish) ώστε το Embedding layer από το μοντέλο Tacotron2 να έχει ίδια διάσταση εισόδου (πλήθος χαρακτήρων) για τις δύο γλώσσες. Έπειτα το μοντέλο εκπαιδεύτηκε για 338 εποχές και όπως θα δούμε στη συνέχεια έδωσε τα πρώτα καλά αποτελέσματα.

### 4.5.1 Συλλογή νέου συνόλου δεδομένων στα ελληνικά

Ως γνωστόν ένα μοντέλο σύνθεσης φωνής από κείμενο, όπως το Tacotron2 είναι ικανό να παράγει ηχητικά δείγματα πολύ υψηλής ποιότητας όταν εκπαιδεύεται με μεγάλο πλήθος δεδομένων από έναν ομιλητή. Τα δεδομένα που διαθέτουμε στα ελληνικά προέρχονται από διάφορες ομιλήτριες, με την τέταρτη από αυτές (speaker 4) να διαθέτει τις περισσότερες ώρες (περίπου 6.5 ώρες). Για το λόγο αυτό επιχειρήθηκε η συλλογή νέων δεδομένων που αντιστοιχούν στη συγκεκριμένη ομιλήτρια προκειμένου να κατασκευαστεί ένα σύνολο ηχογραφήσεων μαζί με τα αντίστοιχα κείμενα το οποίο θα περιέχει αρκούντως μεγάλο πλήθος εκφωνήσεων σε καλή ποιότητα. Σκοπός αυτού είναι είτε η εκπαίδευση ενός μοντέλου απευθείας στα ελληνικά, είτε η μεταφορά μάθησης από κάποιο προεκπαιδευμένο μοντέλο, όμως αυτή τη φορά με χρήση πολύ περισσότερων δεδομένων. Για να το επιτύχουμε, σε πρώτη φάση έγινε αναζήτηση στην εφαρμογή JukeBooks που περιέχει συλλογή από audiobooks στην ελληνική γλώσσα από διάφορους αφηγητές. Έπειτα έγινε αναζήτηση για βιβλία τα οποία εκφωνούνται από τη συγκεκριμένη ομιλήτρια (speaker 4). Έτσι λοιπόν καταλήξαμε στο βιβλίο με τίτλο «Το κουτί», όπου όλη η αφήγησή του είχε διάρκεια περίπου 21 ώρες. Η αποθήκευση της συνολικής αφήγησης σε αρχείο .wav, πραγματοποιήθηκε με εγγραφή οθόνης (screen recording) από κινητό τηλέφωνο και στη συνέχεια έγινε μετατροπή του βίντεο εγγραφής σε αρχείο ήχου. Εδώ θα πρέπει να σημειωθεί ότι η εγγραφή της οθόνης έγινε με ρυθμό δειγματοληψίας 48000 Hz, ενώ τα αρχεία που είχαμε λάβει μέχρι στιγμής από τη συγκεκριμένη ομιλήτρια είχαν ρυθμό δειγματοληψίας 16000 Hz. Επιπλέον λαμβάνοντας υπόψιν ότι τα μοντέλα Tacotron2 και WaveGlow έχουν εκπαιδευτεί με ηχογραφήσεις στα 22050 Hz, τελικά έγινε ένα resampling όλων των ηχογραφήσεων από την τέταρτη ομιλήτρια στα 22050 Hz. Όσον αφορά τη συλλογή του κειμένου για τις νέες ηχογραφήσεις, το βιβλίο αγοράστηκε σε μορφή pdf ώστε να γίνει η απαραίτητη επεξεργασία και να χωριστεί σε μικρότερα τμήματα. Για την επεξεργασία του κειμένου αλλά και των ηχητικών δειγμάτων ακολουθήθηκε η διαδικασία που περιγράψαμε στην ενότητα «Επεξεργασία των Δεδομένων». Μέσω του λογισμικού sail align έγινε ο διαχωρισμός του μεγάλου ηχητικού δείγματος από το βιβλίο σε μικρότερες φράσεις, ώστε να μπορούν να δοθούν ως είσοδος στο μοντέλο Tacotron2. Επίσης έγιναν οι απαραίτητες διορθώσεις και μετατροπές στο κείμενο (μετατροπή αριθμών σε λέξεις, μετατροπή λατινικών χαρακτήρων σε ελληνικούς, αφαίρεση κομματιών που δεν εκφωνούνται στο ηχητικό δείγμα π.χ. υποσημειώσεις κτλ.). Ο διαχωρισμός του κειμένου

έγινε πάλι με χρήση του greek tokenizer από τη βιβλιοθήκη nltk της python. Έτσι το κείμενο (περίπου 600 σελ.) διασπάστηκε σε μικρότερες φράσεις που αντιστοιχούσαν σε μικρότερα ηχητικά δείγματα. Το νέο σύνολο ηχογραφήσεων που προκύπτει ύστερα από την παραπάνω διαδικασία περιέχει τελικά περίπου 26 ώρες από την τέταρτη ομιλήτρια (19.5 ώρες από το νέο βιβλίο και 6.5 ώρες από το ήδη υπάρχον). Η ποιότητα των τελικών δειγμάτων είναι αρκετά καλή και επαρκής για την εκπαίδευση ενός μοντέλου σύνθεσης φωνής στα ελληνικά.

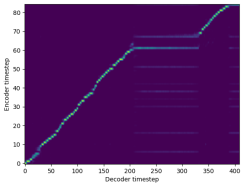
### Πείραμα 4: Fine Tuning στα νέα δεδομένα από την τέταρτη ομιλήτρια

Αξιοποιώντας πλέον τα νέα δεδομένα που προέρχονται αποκλειστικά από την τέταρτη ομιλήτρια δοκιμάσαμε να κάνουμε fine tuning από το προεκπαιδευμένο μοντέλο Tacotron2 στην αγγλική γλώσσα. Το fine tuning του μοντέλου Tacotron2 έγινε για 49 εποχές με ίδιες υπερπαραμέτρους όπως στο προηγούμενο πείραμα.

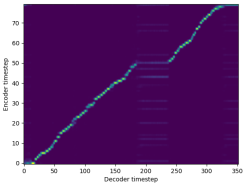
### Πείραμα 5: Fine Tuning στα δεδομένα από την τρίτη και τέταρτη ομιλήτρια

Στο τελικό μας πείραμα δοκιμάσαμε να χρησιμοποιήσουμε τα διαθέσιμα δεδομένα από την τρίτη και την τέταρτη ομιλήτρια μαζί. Η τρίτη ομιλήτρια είχε επίσης πολύ καλή ποιότητα εκφωνήσεων με διάρκεια περίπου 4 ώρες. Ως προεκπαιδευμένο μοντέλο επιλέχθηκε το Tacotron2 από τα αγγλικά και συνεπώς το fine tuning έγινε πάνω σε δεδομένα συνολικής διάρκειας 30 ώρες για 64 εποχές, πάλι με τις ίδιες υπερπαραμέτρους που είχαμε στα δύο προηγούμενα πειράματα.

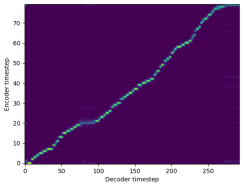
Στην Εικόνα 4.13 παρατηρούμε τα γραφήματα των alignments που προκύπτουν από τα πειράματα 3 (**catotron all speakers**), 4 (**tacotron speaker 4**) και 5 (**tacotron speakers 3,4**) για τη φράση «*Ο Νίκολα και ο Μίλαν έκαναν δυο βήματα πίσω για να μην εμποδίζουν τον εκσκαφέα.*» που εκφωνείται από την τέταρτη ομιλήτρια. Παρατηρούμε ότι και στα τρία γραφήματα υπάρχει διαγώνια μορφή που σημαίνει ότι ο μηχανισμός προσοχής στα μοντέλα λειτουργεί αποτελεσματικά και έτσι επιτυγχάνεται η ευθυγράμμιση μεταξύ των frames του παραγόμενου φασματογραφήματος και των χαρακτήρων της φράσης. Στα δύο πρώτα γραφήματα περίπου στο timestep 200, δεν είναι απόλυτα ξεκάθαρο σε ποιους χαρακτήρες πρέπει να εστιάσει την προσοχή του ο decoder, ενώ στο τρίτο γράφημα το alignment είναι αρκετά καλύτερο. Αυτό που παρατηρούμε στα γραφήματα των alignments γίνεται πιο κατανοητό αν ακούσουμε τα ηχητικά δείγματα που παράγονται από τη συγκεκριμένη φράση. Συγκεκριμένα τα ηχητικά κλιπ που προέκυψαν από τα πειράματα 3 και 4 έχουν μια μικρή παύση, η οποία διακόπτει ελαφρώς τη ροή του λόγου προς το τέλος της ομιλίας. Η συγκεκριμένη παρατήρηση προκύπτει επίσης και από την Εικόνα 4.14, στην οποία παρουσιάζονται τα παραγόμενα φασματογραφήματα στην κλίμακα mel από τα πειράματα 3,4,5 καθώς και το πραγματικό (ground truth) φασματογράφημα. Στα δύο πάνω φασματογραφήματα φαίνεται ξεκάθαρα η παύση που γίνεται στην παραγόμενη ομιλία ενώ το φασματογράφημα από το πείραμα "tacotron speakers 3,4" φαίνεται να μην έχει τέτοιο θέμα. Επίσης στο ground truth φασματογράφημα οι τιμές της έντασης απεικονίζονται με περισσότερη λεπτομέρεια συγκριτικά με τα άλλα τρία φασματογραφήματα. Η λεπτομέρεια αυτή στην εικόνα μπορούμε να πούμε ότι συσχετίζεται με το πόσο καθαρά ακούγεται η φωνή που παράγεται. Είδαμε ότι για τη συγκεκριμένη φράση η συνθετική φωνή έχει λίγο χαμηλότερη ποιότητα και ακούγεται ελαφρώς «ρομποτική». Παρ' όλ' αυτά και από τα τρία πειράματα η φωνή που παράγεται είναι σχετικά ικανοποιητική όπως προκύπτει και από την αξιολόγηση που θα δούμε στην επόμενη ενότητα.




(a) catotron all speakers

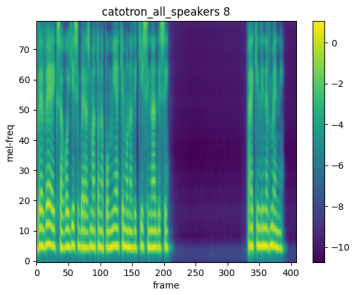



(b) tacotron speaker 4

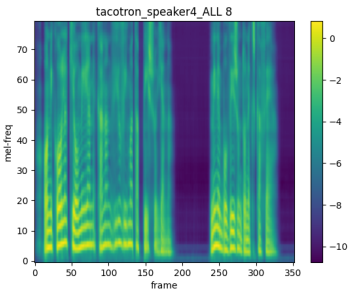



(c) tacotron speakers 3,4

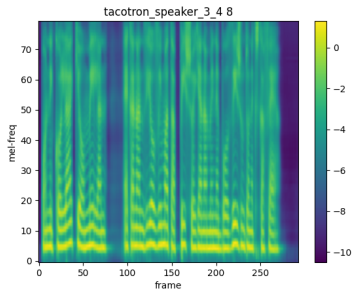
Εικόνα 4.13: Γραφήματα των alignments που προκύπτουν από τα πειράματα 3,4,5 για μια συγκεκριμένη φράση στα ελληνικά.



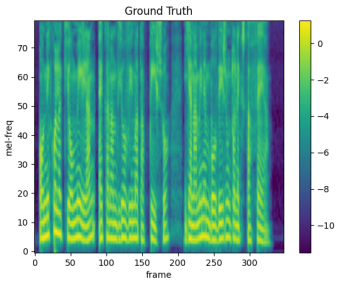

(a) catotron all speakers




(b) tacotron speaker 4




(c) tacotron speakers 3,4



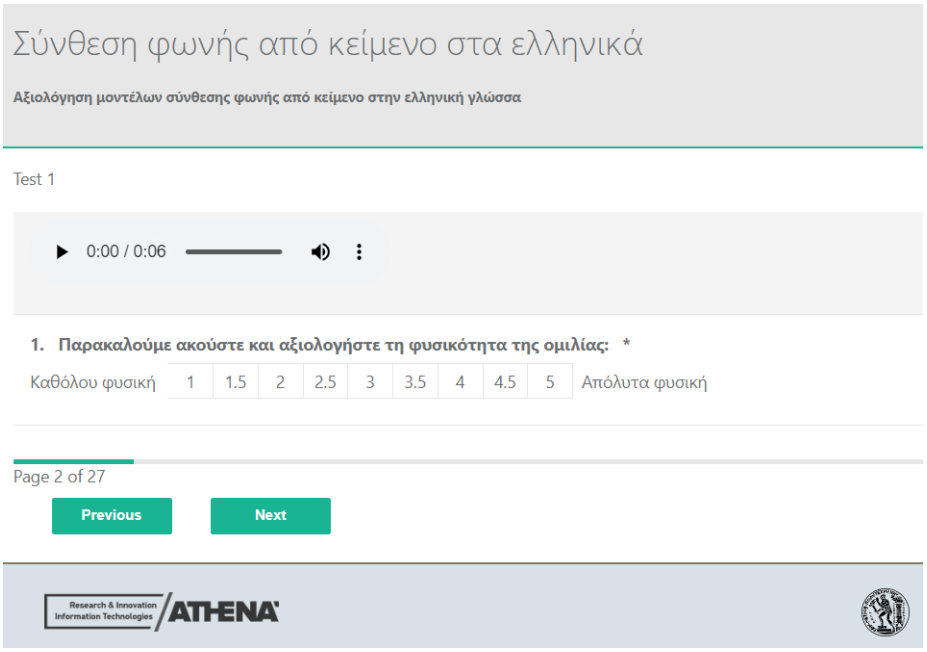

(d) Ground Truth

Εικόνα 4.14: Φασματογραφήματα σε κλίμακα mel που προκύπτουν από τα πειράματα 3,4,5 μαζί με το ground truth φασματογράφημα για μια συγκεκριμένη φράση στα ελληνικά.

## 4.6 Πειραματική Αξιολόγηση

Στην ενότητα αυτή περιγράφουμε τη διαδικασία αξιολόγησης των πειραμάτων και παρουσιάζουμε τα σχετικά αποτελέσματα. Για να αξιολογήσουμε τη φυσικότητα των παραγόμενων ηχητικών δειγμάτων από το εκάστοτε πείραμα, επιλέγουμε 25 ηχητικά δείγματα από το σύνολο δεδομένων μας, τα οποία παρουσιάζονται σε ορισμένους βαθμολογητές μέσω ενός ερωτηματολογίου. Κάθε βαθμολογητής καλείται να ακούσει τα ηχητικά κλιπ που του παρουσιάζονται και να τα βαθμολογήσει στην κλίμακα MOS (Mean Opinion Score) ή Likert [Jos+15], δίνοντας τους μία τιμή από το 1 έως το 5 με διαβαθμίσεις της τάξης του 0.5, όπως φαίνεται στην Εικόνα 4.15. Ο αριθμός 1 αντιστοιχεί σε ομιλία η οποία δεν είναι «Καθόλου φυσική», ενώ ο αριθμός 5 αντιστοιχεί σε «Απόλυτα φυσική» ομιλία.

Πέρα από τα τρία πειράματα που εξετάζουμε, στο ερωτηματολόγιο συμπεριλαμβάνονται και τα ηχητικά δείγματα που προκύπτουν από το μοντέλο WaveGlow δίνοντας ως είσοδο τα πραγματικά



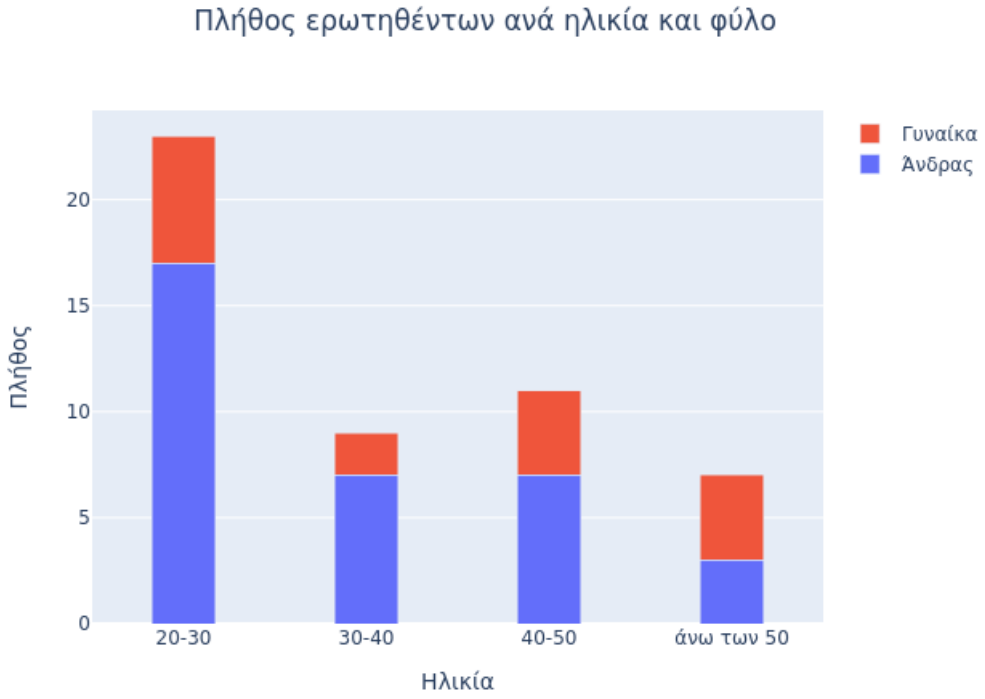

Εικόνα 4.15: Ερωτηματολόγιο αξιολόγησης μοντέλων σύνθεσης φωνής από κείμενο στα ελληνικά.

φασματογραφήματα (Mel + WaveGlow). Επίσης δίνονται και τα πραγματικά (ground truth) ηχητικά κλιπ από τα κείμενα που θέλουμε να μετατρέψουμε σε φωνή. Συνολικά δημιουργούνται 5 ερωτηματολόγια, καθένα από τα οποία περιλαμβάνει τις ίδιες 25 φράσεις (transcriptions), οι οποίες όμως παράγονται από διαφορετικό μοντέλο-πείραμα κάθε φορά. Κάθε ερωτηματολόγιο επιλέγεται με τυχαίο τρόπο και παρουσιάζεται σε κάποιον βαθμολογητή. Συνεπώς αν σε δύο βαθμολογητές τύχουν διαφορετικά ερωτηματολόγια, τότε και οι δύο θα ακούσουν τις ίδιες φράσεις, οι οποίες όμως θα έχουν προκύψει από διαφορετικό πείραμα σε κάθε περίπτωση. Έτσι αφού σε κάθε βαθμολογητή παρουσιάζεται τυχαία ένα από τα 5 ερωτηματολόγια, θεωρούμε ότι θα υπάρχει περισσότερη διαφάνεια σε όλη τη διαδικασία αξιολόγησης αφού οι ακροατές δεν θα γνωρίζουν εκ των προτέρων από ποιο πείραμα έχει προέλθει το κάθε ηχητικό δείγμα. Όλα τα ηχητικά κλιπ έχουν διάρκεια μέχρι 10 δευτερόλεπτα ώστε ο εκτιμώμενος χρόνος συμπλήρωσης του ερωτηματολογίου να είναι μέχρι 10 λεπτά και ο βαθμολογητής να μπορέσει να ακούσει προσεκτικά όλες τις φράσεις. Για όλα τα πειράματα ως vocoder για την μετατροπή των φασματογραφημάτων στο αντίστοιχο ηχητικό δείγμα χρησιμοποιήθηκε το μοντέλο WaveGlow που είχε εκπαιδευτεί στην αγγλική γλώσσα.

Συνολικά το ερωτηματολόγιο απαντήθηκε από 50 βαθμολογητές εκ των οποίων οι 34 ήταν άνδρες και οι 16 γυναίκες. Στο Διάγραμμα 4.16 φαίνεται και το πλήθος των ερωτηθέντων ανά ηλικία και φύλο. Η πλειοψηφία των απαντήσεων προήλθε από άτομα ηλικίας 20 έως 30 ετών (23 απαντήσεις). Ο Πίνακας 4.4 περιλαμβάνει τις βαθμολογίες στην κλίμακα MOS που προέκυψαν για κάθε πείραμα μαζί με το 95% διάστημα εμπιστοσύνης. Οι τιμές αυτές προκύπτουν ως μέσοι όροι των απαντήσεων που έδωσαν οι βαθμολογητές για κάθε φράση που προερχόταν από το εκάστοτε πείραμα[3]. Η ερμηνεία του 95% διαστήματος εμπιστοσύνης είναι ότι με πιθανότητα 0.95 το τυχαίο[4] διάστημα που προκύπτει για κάθε πείραμα περιέχει την άγνωστη παράμετρο μέσης τιμής του MOS.

[3]για το κάθε πείραμα προκύπτουν 250 τιμές (5 από κάθε ερωτηματολόγιο και συνολικά λάβαμε 50 ερωτηματολόγια).

[4]είναι τυχαίο διότι προκύπτει από κάποιο δείγμα ερωτηθέντων.




Διάγραμμα 4.16: Πλήθος ερωτηθέντων ανά ηλικία και φύλο.

Το διάστημα αυτό προκύπτει από την κατανομή Student με $n-1$ βαθμούς ελευθερίας, όπου το $n$ ισούται με το πλήθος των απαντήσεων από το δείγμα, δηλαδή 250. Έτσι οι τιμές του Πίνακα 4.4 προκύπτουν από τον τύπο:

$$\overline{x} \pm t_{n-1,\frac{\alpha}{2}} \frac{s}{\sqrt{n}}, \tag{4.6.1}$$

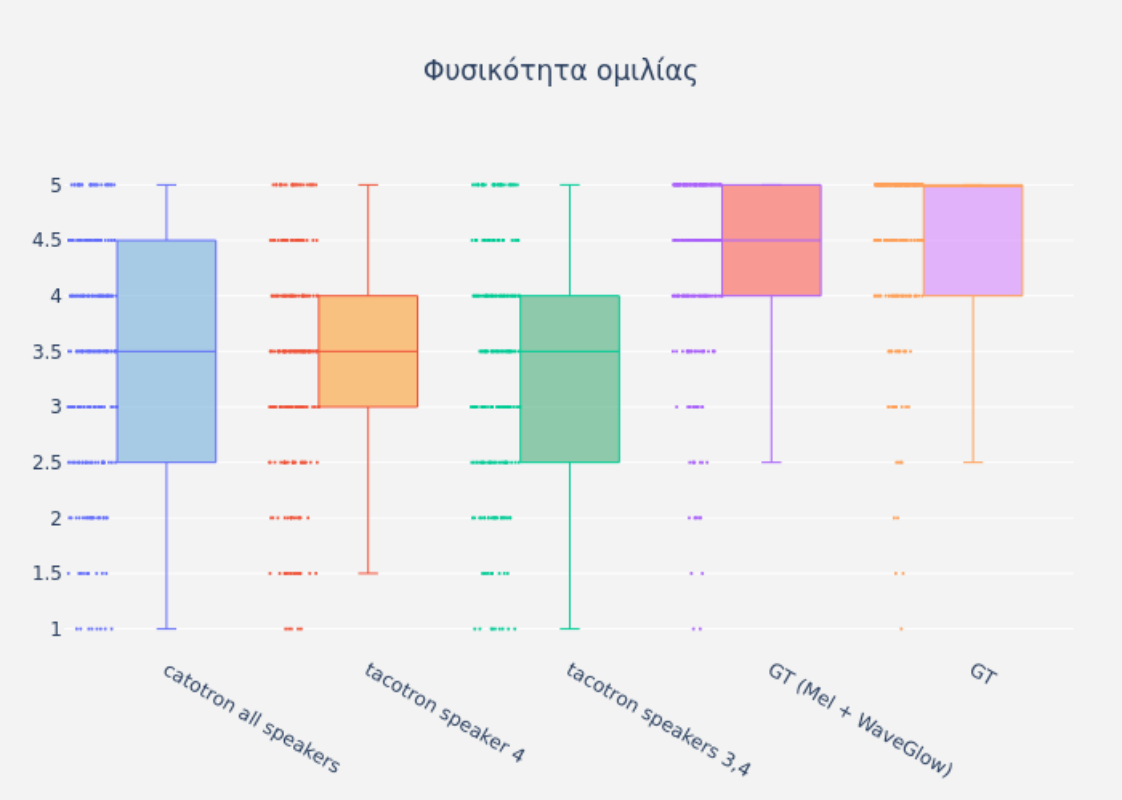
όπου $\overline{x}$ είναι η δειγματική μέση τιμή του MOS, $t_{n-1,\alpha/2}$ η τιμή από την κατανομή Student με $\alpha = 1 - 0.95 = 0.05$ και $s$ η δειγματική τυπική απόκλιση των 250 βαθμολογιών. Όπως είναι

| **Model** | **MOS** |
|---|---|
| catotron all speakers | **3.434 ± 0.131** |
| tacotron speaker 4 | 3.392 ± 0.126 |
| tacotron speakers 3,4 | 3.400 ± 0.134 |
| GT (Mel + WaveGlow) | 4.228 ± 0.103 |
| GT | 4.496 ± 0.092 |

Πίνακας 4.4: Βαθμολογίες στην κλίμακα MOS με 95% διάστημα εμπιστοσύνης.

αναμενόμενο παρατηρούμε ότι η υψηλότερη μέση βαθμολογία αντιστοιχεί στις πραγματικές (GT) εκφωνήσεις με τιμή 4.496. Στη συνέχεια ακολουθεί η βαθμολογία που προκύπτει από την μετατροπή των πραγματικών φασματογραφημάτων σε ηχητικό δείγμα από το μοντέλο WaveGlow (Mel + WaveGlow) με τιμή 4.228. Η τιμή αυτή μας υποδηλώνει ότι το προεκπαιδευμένο «αγγλικό»

μοντέλο WaveGlow που χρησιμοποιούμε στα πειράματά μας μπορεί να παράγει ηχητικά δείγμα τα οποία είναι αρκετά κοντά στα πραγματικά. Επίσης επιβεβαιώνει σε ένα βαθμό το γεγονός ότι το μοντέλο αυτό είναι ανεξάρτητο από τη γλώσσα και την ταυτότητα του ομιλητή. Στη συνέχεια παρατηρούμε ότι οι τιμές που προκύπτουν από τα τρία πειράματα είναι αρκετά κοντά μεταξύ τους με ελαφρώς υψηλότερη τιμή 3.434 να δίνει το πρώτο πείραμα "catotron all speakers". Έπειτα ακολουθεί το πείραμα "tacotron speakers 3,4" με τιμή 3.400 και τέλος το πείραμα "tacotron speaker 4" με τιμή 3.392. Τα παραπάνω αποτελέσματα απεικονίζονται και στο Διάγραμμα 4.17 που περιέχει τα αντίστοιχα θηκοδιαγράμματα (boxplots) κάθε πειράματος. Φαίνεται ότι η διάμεσος των βαθ-




Διάγραμμα 4.17: Μετρήσεις στην κλίμακα MOS για τη φυσικότητα των συνθετικών και πραγματικών εκφωνήσεων στα ελληνικά.

μολογιών MOS για τα τρία πρώτα πειράματα είναι στο 3.5, για το μοντέλο WaveGlow στο 4.5 και για τις πραγματικές εκφωνήσεις στο 5. Αξίζει επίσης να αναφέρουμε ορισμένα σχόλια που λάβαμε από τα ερωτηματολόγια αναφορικά με την ποιότητα των ηχητικών κλιπ. Τα σχόλια αυτά είναι τα εξής:

- Ο τρόπος βαθμολόγησης ενδεχομένως να αλλάζει λίγο καθώς προχωράει κανείς από δείγμα σε δείγμα γιατί ο ακροατής κάνει μια κανονικοποίηση.
- Πολύ ενδιαφέρον, αρκετά φυσικό. Σε όσα έβαλα 1 υπάρχει σφάλμα στον τονισμό.
- Κάποια δείγματα έχουν αφύσικες αρμονικές/ηχόχρωμα σαν να μιλάνε ταυτόχρονα δύο άτομα ή σαν να έχει φίλτρο flanger.
- Έχει ένα θεματάκι με τον τονισμό σε κάποιες λέξεις, αν το εντόπισα σωστά στις φωνολογικές

λέξεις που έχουν και εγκλιτικά (π.χ. το τετράδιό μου που χρειάζεται δύο τόνους) και σε κάποιες άλλες μάλλον πολυσύλλαβες.

- Η βραχνάδα σε κάποιες εκφωνήσεις προϊδεάζει για συνθετική φωνή. Σε κάποια κλιπ υπήρχε έντονος θόρυβος στο background.

- Σε αρκετά είναι σχεδόν φυσική. Είναι σχεδόν φυσική όταν οι τόνοι είναι πιο υψηλοί υπάρχει κάποιος επιτονισμός π.χ. προσταγή κάτι τέτοιο, υπάρχει και κάποια ταχύτητα στην εκφώνηση κι αυτό επίσης δίνει μια φυσικότητα. Εκεί που χάνει είναι στα πιο χαμηλά, πιο ήσυχα, επιτονισμός αφήγησης, επίσης κάπου δεν είναι καθαρό όταν υπάρχουν δίφθογγοι και κάποια συμπλέγματα συμφωνικά.

Όπως φαίνεται και από τα παραπάνω σχόλια, το αποτέλεσμα από τα πειράματά μας είναι σχετικά καλό, σαφώς όμως το σύστημα χρειάζεται περαιτέρω βελτίωση ώστε να διορθωθούν συγκεκριμένα ζητήματα που επηρεάζουν σημαντικά τη φυσικότητα της συνθετικής φωνής. Παραδείγματος χάριν παρατηρήθηκε ότι σε ορισμένα δείγματα υπήρχε λάθος επιτονισμός σε κάποιες λέξεις. Επίσης μπορεί κάποια λέξη μέσα σε μια φράση να εκφωνούνταν πιο σιγά από τις υπόλοιπες ή και καθόλου και συνεπώς αυτό να επηρέαζε τη ροή της ομιλίας. Επιπλέον όπως αναφέρεται στο τρίτο σχόλιο παρατηρήθηκε ότι ορισμένα δείγματα ακούγονταν σα να εκφωνούνται ταυτόχρονα από δύο φωνές. Η παρατήρηση αυτή συσχετίζεται με το γεγονός ότι στα πειράματά μας χρησιμοποιήσαμε δεδομένα που προέρχονταν από διάφορες ομιλήτριες. Συγκεκριμένα το πείραμα "catotron all speakers" εκπαιδεύτηκε με δεδομένα από τέσσερις ομιλήτριες συνολικής διάρκειας 14 ώρες και το πείραμα "tacotron speakers 3,4" με δεδομένα από δύο ομιλήτριες συνολικής διάρκειας περίπου 30 ώρες. Μόνο στο πείραμα "tacotron speaker 4" αξιοποιήθηκαν δεδομένα αποκλειστικά από την τέταρτη ομιλήτρια. Προφανώς η ταυτότητα του ομιλητή αλλά και η αναλογία σε ώρες των εκφωνήσεων που χρησιμοποιούνται κατά την εκπαίδευση επηρεάζει και το χρώμα-ταυτότητα της συνθετικής φωνής. Έτσι λοιπόν από το πείραμα "tacotron speaker 4" η φωνή που προκύπτει προέρχεται μόνο από την τέταρτη ομιλήτρια ενώ στα άλλα δύο πειράματα και από τις δύο. Αυτό όμως δεν επηρεάζει τη βαθμολογία ενός ακροατή εφόσον δε γνωρίζει εκ των προτέρων από ποιο μοντέλο προέρχεται η κάθε φωνή. Με άλλα λόγια μπορεί το πείραμα "tacotron speaker 4" να παράγει φωνή μόνο από την τέταρτη ομιλήτρια αλλά επειδή και τα άλλα δύο πειράματα παράγουν φωνή από την ίδια ομιλήτρια, ο ακροατής δε γνωρίζει την αντιστοιχία φωνή-πείραμα, ώστε να υπάρχει μεροληψία στον τρόπο αξιολόγησης των ηχητικών δειγμάτων.

Μια άλλη παρατήρηση που προκύπτει από τη σύγκριση των τιμών του MOS στα τρία πειράματα είναι η εξής. Αν και το πείραμα "tacotron speaker 4" χρησιμοποιεί αρκετές ώρες καλής ποιότητας από μια μόνο ομιλήτρια, εντούτοις λαμβάνει ελαφρώς χαμηλότερη βαθμολογία συγκριτικά με τα άλλα δύο πειράματα. Αυτό οφείλεται στο γεγονός ότι οι φράσεις που δόθηκαν στα μοντέλα για σύνθεση φωνής προέρχονταν από τις ομιλήτριες 3 και 4. Ακούγοντας τα ηχητικά κλιπ από το συγκεκριμένο πείραμα διαπιστώνουμε ότι το μοντέλο καταφέρνει να παράγει με βελτιωμένη ποιότητα τις φράσεις που προέρχονται από την τέταρτη ομιλήτρια ενώ οι υπόλοιπες που προέρχονται από την τρίτη ομιλήτρια, υστερούν λίγο σε χαρακτηριστικά όπως ο επιτονισμός, οι μικρές παύσεις και στο ότι η φωνή ακούγεται λίγο πιο «ρομποτική». Μια άλλη παρατήρηση που αξίζει να σημειωθεί είναι ότι το πείραμα "catotron all speakers" που χρησιμοποιεί ως προεκπαιδευμένο μοντέλο το Catotron που έχει εκπαιδευτεί στα καταλανικά δίνει την υψηλότερη βαθμολογία συγκριτικά με τα άλλα δύο πειράματα, στα οποία ως προεκπαιδευμένο μοντέλο χρησιμοποιήθηκε το Tacotron2 στην αγγλική γλώσσα. Βέβαια η διαφορά μεταξύ τους είναι αρκετά μικρή αλλά επιβεβαιώνει ως ένα βαθμό την αρχική πεποίθηση ότι επειδή τα καταλανικά μοιάζουν περισσότερο σαν γλώσσα με

τα ελληνικά, το προεκπαιδευμένο μοντέλο στα καταλανικά θα δίνει καλύτερα αποτελέσματα από το αντίστοιχο στα αγγλικά. Υπενθυμίζουμε επίσης ότι στο Catotron, όπως αναφέρεται από τους συγγραφείς, χρησιμοποιήθηκε ως προεκπαιδευμένο μοντέλο το Tacotron2 στην αγγλική γλώσσα. Οπότε στην πραγματικότητα στο πείραμά μας "catotron all speakers" ουσιαστικά έχει γίνει πρώτα fine tuning στα καταλανικά με αρχικό μοντέλο το «αγγλικό» Tacotron2 και έπειτα κάνουμε ξανά fine tuning στα ελληνικά. Επιπλέον αξίζει να αναφέρουμε ότι η ποιότητα των νέων δεδομένων που συλλέξαμε από την τέταρτη ομιλήτρια είναι ναι μεν αρκετά καθαρή αλλά λιγότερο καλή από την ποιότητα των αρχικών ηχογραφήσεων που είχαμε λάβει από την ίδια ομιλήτρια. Αυτό συμβαίνει διότι χρειάστηκε να γίνει downsampling των ηχογραφήσεων από τα 48000 Hz (ρυθμός δειγματοληψίας από το screen recording του τηλεφώνου) στα 22050 Hz, που είναι ο ρυθμός δειγματοληψίας των ηχητικών δειγμάτων στα οποία είχαν προεκπαιδευτεί τα μοντέλα Tacotron2 και WaveGlow. Αυτό ενδεχομένως να επηρεάζει σε κάποιο βαθμό την ποιότητα της παραγόμενης φωνής που προκύπτει από τα πειράματα "tacotron speaker 4" και "tacotron speakers 3,4", με την έννοια ότι αν όλα μας τα δεδομένα από την τέταρτη ομιλήτρια είχαν ίδιο ρυθμό δειγματοληψίας εξαρχής, ενδεχομένως να προέκυπταν ακόμα καλύτερα αποτελέσματα από αυτά τα δύο πειράματα.

Τέλος είναι σημαντικό να αναφέρουμε ότι ο χρόνος του fine tuning στα ελληνικά δεδομένα και για τα τρία πειράματα ήταν περίπου μία μέρα. Βλέπουμε λοιπόν ότι με τη μεταφορά μάθησης μειώνεται κατά πολύ ο χρόνος εκπαίδευσης, συγκριτικά πάντα με το χρόνο που χρειαζόταν το μοντέλο Tacotron2 για να εκπαιδευτεί από την αρχή είτε στα ελληνικά είτε στα ισπανικά δεδομένα. Επιπλέον μόνο με τη χρήση μιας GPU τα πειράματα έδωσαν καλά αποτελέσματα σε ένα εύλογο χρονικό διάστημα. Στην επόμενη ενότητα παρουσιάζουμε τα συμπεράσματα και ορισμένες μελλοντικές επεκτάσεις αναφορικά με τα πειράματά μας αλλά και τον τρόπο βελτίωσης των αποτελεσμάτων.

## 4.7 Συμπέρασμα - Μελλοντικές Επεκτάσεις

Όπως έχουμε ήδη αναφέρει τα ελληνικά είναι μια low resource γλώσσα και αυτό καθιστά το πρόβλημα της σύνθεσης φωνής από κείμενο ακόμα πιο σύνθετο, πόσο μάλλον όταν θέλουμε η συνθετική φωνή που παράγεται να είναι απόλυτα φυσική. Ένα επίσης σημαντικό ζήτημα που προκύπτει είναι ότι πέρα από το πλήθος των δεδομένων που απαιτούνται, χρειάζονται και αρκετοί υπολογιστικοί πόροι για να εκπαιδεύσουμε τέτοια μοντέλα. Στην περίπτωσή μας για παράδειγμα ακόμα και κατά τη διάρκεια του inference το μοντέλο WaveGlow χρειάζεται μία GPU για να παράγει τα ηχητικά δείγματα σε ένα λογικό χρονικό διάστημα. Παρ' όλ' αυτά όπως είδαμε και από τα πειράματά μας, το πρόβλημα του χρόνου εκπαίδευσης μπορεί να αντιμετωπιστεί αποτελεσματικά με χρήση της μεταφοράς μάθησης. Είδαμε για παράδειγμα ότι στα αρχικά μας πειράματα η εκπαίδευση απευθείας στα ελληνικά ή τα ισπανικά διαρκούσε από μερικές μέρες έως και μήνα για τα μοντέλα Tacotron2 και WaveGlow, χωρίς όμως να έχουμε το επιθυμητό αποτέλεσμα. Αντιθέτως με τη μεταφορά μάθησης, πέραν της βελτίωσης του χρόνου εκπαίδευσης καταφέραμε να παράγουμε ομιλία καλής ποιότητας στα ελληνικά ακόμη και με λιγότερα δείγματα, όπως για παράδειγμα στο πείραμα "catotron all speakers" που οι διαθέσιμες ώρες ηχογραφήσεων ήταν μόλις 14. Γενικότερα λοιπόν από τη μελέτη μας προκύπτουν τα εξής συμπεράσματα. Αρχικά όσον αφορά το μοντέλο WaveGlow είδαμε ότι είναι ικανό να παράγει συνθετική φωνή πολύ καλής ποιότητας ανεξάρτητα από τη γλώσσα ομιλίας και την ταυτότητα του ομιλητή, όπως προέκυψε και από την αντίστοιχη βαθμολογία στην κλίμακα MOS. Επίσης θεωρούμε σημαντική τη συλλογή νέων δεδομένων πολύ καλής ποιότητας από μία μόνο ομιλήτρια στα ελληνικά με συνολική διάρκεια περίπου 19.5 ώρες. Έτσι λοιπόν το τελικό σύνολο δεδομένων που προκύπτει από τη συγκεκριμένη ομιλήτρια περιλ-

αμβάνει περίπου 26 ώρες ηχογραφήσεων. Τα δεδομένα αυτά μπορούν να αξιοποιηθούν είτε για την εκπαίδευση ενός συστήματος σύνθεσης φωνής απευθείας στα ελληνικά, είτε όπως έγινε και στα πειράματά μας για fine tuning των μοντέλων Tacotron2 και WaveGlow πάνω σε περισσότερα δεδομένα. Επιπλέον τα αποτελέσματα της μελέτης μας δείχνουν ότι και από τα τρία πειράματα παράγεται φωνή σε σχετικά καλό επίπεδο (MOS $\approx$ 3.5), όπως αυτό κρίθηκε με τη βαθμολογία που προέκυψε από το δείγμα των 50 ερωτηθέντων. Σαφώς όμως υπάρχει περιθώριο βελτίωσης ώστε να αυξηθεί περαιτέρω η ποιότητα της συνθετικής φωνής και να πλησιάσει τη φυσικότητα των πραγματικών εκφωνήσεων.

Η βελτίωση λοιπόν της φυσικότητας στην παραγόμενη ομιλία αποτελεί μια βασική και άμεση επέκταση της μελέτης μας στο πρόβλημα της σύνθεσης φωνής στην ελληνική γλώσσα. Είδαμε και από τα σχόλια της αξιολόγησης των πειραμάτων μας, ότι αναφέρεται συχνά το θέμα του επιτονισμού, των παύσεων και του ότι σε ορισμένα ηχητικά κλιπ η παραγόμενη φωνή ακούγεται πιο «ρομποτική». Τα δύο πρώτα ζητήματα σχετίζονται με το πόσο επιτυχές είναι το alignment του μοντέλου μεταξύ των χαρακτήρων του κειμένου και των frames από το φασματογράφημα. Για να βελτιωθεί το alignment θα μπορούσαμε να χρησιμοποιήσουμε εναλλακτικά τη συνάρτηση ελαχιστοποίησης από το μοντέλο Regotron για το fine-tuning στα ελληνικά δεδομένα. Μια άλλη επιλογή θα ήταν να δοκιμάσουμε και άλλους μηχανισμούς προσοχής στο μοντέλο Tacotron2 πέραν του location sensitive attention. Όσον αφορά τη βελτίωση ποιότητας της φωνής θα μπορούσε να γίνει fine tuning και του μοντέλου WaveGlow πάνω στα ελληνικά δεδομένα. Σε όλα μας τα πειράματα χρησιμοποιήσαμε το προεκπαιδευμένο μοντέλο WaveGlow στην αγγλική γλώσσα, το οποίο αν και έδινε πολύ καλά αποτελέσματα, είναι πολύ πιθανόν η παραγόμενη φωνή να ήταν ακόμη καλύτερη αν το μοντέλο εκπαιδευόταν για λίγες παραπάνω εποχές και στα ελληνικά δεδομένα. Επιπλέον θα μπορούσε να γίνει δοκιμή και άλλων αρχιτεκτονικών όπως το Transformer TTS, το MelGAN και το WaveGrad2, ώστε να γίνει μια σύγκριση των αποτελεσμάτων που προκύπτουν από τα βασικότερα state of the art μοντέλα και να δούμε αν κάποιο από αυτά βελτιώνει τα ζητήματα της ποιότητας στη φωνή ή του χρόνου εκπαίδευσης και συμπερασματολογίας. Μια άλλη προσέγγιση που θα θέλαμε να μελετήσουμε είναι η προσθήκη χαρακτηριστικών όπως το συναίσθημα και η προσωδία στη συνθετική φωνή. Ήδη τα δεδομένα που διαθέτουμε περιέχουν αρκετή εκφραστικότητα μιας και οι εκφωνήτριες αφηγούνται τα συγκεκριμένα βιβλία με πολύ παραστατικό τρόπο. Θα ήταν λοιπόν ενδιαφέρον να επεκτείνουμε τα πειράματά μας αξιοποιώντας το μοντέλο Global Style Tokens μέσω του οποίου μπορεί να ρυθμιστεί η εκφραστικότητα και άλλα χαρακτηριστικά του ηχητικού δείγματος, όπως η ταχύτητα και γενικότερα το στυλ της ομιλίας. Ταυτόχρονα θα θέλαμε το σύστημα να μπορεί να παράγει φωνή και από περισσότερους ομιλητές άντρες ή γυναίκες με μελλοντικό στόχο να μπορεί να αξιοποιηθεί σε μια εμπορική εφαρμογή. Αυτό μπορεί να προκύψει είτε με τη συλλογή δεδομένων από περισσότερους ομιλητές είτε με την αξιοποίηση κάποιου μοντέλου μετατροπής φωνής όπως το CopyCat, προκειμένου να δημιουργηθούν δεδομένα από νέους ομιλητές και στη συνέχεια να εκπαιδευτεί ένα σύστημα σύνθεσης φωνής για τον εκάστοτε ομιλητή. Συγχρόνως στο ήδη υπάρχον σύστημα για τα ελληνικά, θα ήταν επιθυμητό να γίνεται επιλογή της ταυτότητας της ομιλήτριας που εκφωνεί μια συγκεκριμένη φράση. Αυτό συνήθως επιτυγχάνεται ενσωματώνοντας κάποιο speaker embedding στην αρχιτεκτονική του μοντέλου Tacotron2 ώστε να λαμβάνεται υπόψιν και η ταυτότητα του ομιλητή [Jia+18]. Τέλος μια επέκταση του προβλήματος της σύνθεσης φωνής από κείμενο που θα θέλαμε να μελετήσουμε, είναι η οπτικοακουστική σύνθεση φωνής από κείμενο στα ελληνικά. Στην πραγματικότητα το πρόβλημα αυτό επεκτείνει τη σύνθεση φωνής από κείμενο στη μετατροπή της παραγόμενης ομιλίας σε ανθρωπόμορφο βίντεο. Ορισμένες από τις βασικές εργασίες που μελετούν το συγκεκριμένο πρόβλημα είναι το "MakeItTalk" [Zho+21] και το "Synthesizing Obama" [SSK17].

Ανακεφαλαιώνοντας λοιπόν, είναι σαφές ότι το πρόβλημα της σύνθεσης φωνής από κείμενο για τα ελληνικά δε σταματάει εδώ. Ήδη το συγκεκριμένο πεδίο εξελίσσεται συνεχώς με όλο και περισσότερες δημοσιεύσεις και εργασίες. Επιπλέον εφαρμογές που στηρίζονται σε αυτό το πεδίο, αξιοποιούνται όλο και περισσότερο σε εταιρικά περιβάλλοντα και όχι μόνο. Έτσι λοιπόν η σύνθεση φωνής από κείμενο αποτελεί ένα πολύ σύγχρονο θέμα με μελλοντικές επεκτάσεις και πολλαπλές εφαρμογές, που σίγουρα στο κοντινό μέλλον θα χρησιμοποιούνται όλο και περισσότερο στην καθημερινότητά μας.

# Λίστα Σχημάτων

# Βιβλιογραφία


[BKH16] Ba, J. L., Kiros, J. R., and Hinton, G. E. "Layer normalization". In: *arXiv preprint arXiv:1607.06450* (2016).

[Bäc+22] Bäckström, T. et al. *Introduction to Speech Processing*. 2nd ed. 2022. DOI: 10.5281/zenodo.6821775. URL: https://speechprocessingbook.aalto.fi.

[BCB14] Bahdanau, D., Cho, K., and Bengio, Y. "Neural machine translation by jointly learning to align and translate". In: *arXiv preprint arXiv:1409.0473* (2014).

[Bon+08] Bonafonte, A. et al. "Corpus and Voices for Catalan Speech Synthesis". In: *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*. Marrakech, Morocco: European Language Resources Association (ELRA), May 2008. URL: http://www.lrec-conf.org/proceedings/lrec2008/pdf/835_paper.pdf.

[BB86] Bracewell, R. N. and Bracewell, R. N. *The Fourier transform and its applications*. Vol. 31999. McGraw-Hill New York, 1986.

[Che+20] Chen, N. et al. "WaveGrad: Estimating gradients for waveform generation". In: *arXiv preprint arXiv:2009.00713* (2020).

[Che+21] Chen, N. et al. "WaveGrad 2: Iterative Refinement for Text-to-Speech Synthesis". In: *arXiv preprint arXiv:2106.09660* (2021).

[Cho+15] Chorowski, J. et al. "Attention-based models for speech recognition". In: *arXiv preprint arXiv:1506.07503* (2015).

[Chu+14] Chung, J. et al. "Empirical evaluation of gated recurrent neural networks on sequence modeling". In: *arXiv preprint arXiv:1412.3555* (2014).

[Cla+19] Clark, R. et al. "Evaluating long-form text-to-speech: Comparing the ratings of sentences and paragraphs". In: *arXiv preprint arXiv:1909.03965* (2019).

[DLR77] Dempster, A. P., Laird, N. M., and Rubin, D. B. "Maximum likelihood from incomplete data via the EM algorithm". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 39.1 (1977), pp. 1–22.

[DSB16] Dinh, L., Sohl-Dickstein, J., and Bengio, S. "Density estimation using real nvp". In: *arXiv preprint arXiv:1605.08803* (2016).

[Dum+18] Dumoulin, V. et al. "Feature-wise transformations". In: *Distill* 3.7 (2018), e11.

[For73] Forney, G. D. "The viterbi algorithm". In: *Proceedings of the IEEE* 61.3 (1973), pp. 268–278.

[Geo+22] Georgiou, E. et al. "Regotron: Regularizing the Tacotron2 architecture via monotonic alignment loss". In: *arXiv preprint arXiv:2204.13437* (2022).

[Gon+16] Gonzalvo, X. et al. “Recent advances in Google real-time HMM-driven unit selection synthesizer”. In: (2016).

[Goo+14] Goodfellow, I. et al. “Generative adversarial nets”. In: *Advances in neural information processing systems* 27 (2014).

[GWD14] Graves, A., Wayne, G., and Danihelka, I. “Neural turing machines”. In: *arXiv preprint arXiv:1410.5401* (2014).

[GL84] Griffin, D. and Lim, J. “Signal estimation from modified short-time Fourier transform”. In: *IEEE Transactions on acoustics, speech, and signal processing* 32.2 (1984), pp. 236–243.

[He+16] He, K. et al. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[HO07] Hershey, J. R. and Olsen, P. A. “Approximating the Kullback Leibler divergence between Gaussian mixture models”. In: *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*. Vol. 4. IEEE. 2007, pp. IV–317.

[HJA20] Ho, J., Jain, A., and Abbeel, P. “Denoising diffusion probabilistic models”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 6840–6851.

[HS97] Hochreiter, S. and Schmidhuber, J. “Long short-term memory”. In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[HB96] Hunt, A. J. and Black, A. W. “Unit selection in a concatenative speech synthesis system using a large speech database”. In: *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*. Vol. 1. IEEE. 1996, pp. 373–376.

[Huy+21] Huybrechts, G. et al. “Low-resource expressive text-to-speech using data augmentation”. In: *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2021, pp. 6593–6597.

[HD05] Hyvärinen, A. and Dayan, P. “Estimation of non-normalized statistical models by score matching.” In: *Journal of Machine Learning Research* 6.4 (2005).

[IS15] Ioffe, S. and Szegedy, C. “Batch normalization: Accelerating deep network training by reducing internal covariate shift”. In: *International conference on machine learning*. PMLR. 2015, pp. 448–456.

[IJ17a] Ito, K. and Johnson, L. *The LJ Speech Dataset*. https://keithito.com/LJ-Speech-Dataset/. 2017.

[IJ17b] Ito, K. and Johnson, L. *The lj speech dataset*. 2017.

[ITU03] ITU-R, R. “1534-1, method for the subjective assessment of intermediate quality levels of coding systems (mushra),””. In: *International Telecommunication Union* (2003).

[Jia+18] Jia, Y. et al. “Transfer learning from speaker verification to multispeaker text-to-speech synthesis”. In: *Advances in neural information processing systems* 31 (2018).

[Jos+15] Joshi, A. et al. “Likert scale: Explored and explained”. In: *British journal of applied science & technology* 7.4 (2015), p. 396.

[Kar+20] Karlapati, S. et al. “Copycat: Many-to-many fine-grained prosody transfer for neural text-to-speech”. In: *arXiv preprint arXiv:2004.14617* (2020).

[Kat+11] Katsamanis, A. et al. “SailAlign: Robust long speech-text alignment”. In: *Proc. of workshop on new tools and methods for very-large scale phonetics research*. 2011.

[Kaw06] Kawahara, H. “STRAIGHT, exploitation of the other aspect of VOCODER: Perceptually isomorphic decomposition of speech sounds”. In: *Acoustical science and technology* 27.6 (2006), pp. 349–353.

[Ken+19] Kenter, T. et al. “CHiVE: Varying prosody in speech synthesis with a linguistically driven dynamic hierarchical conditional variational network”. In: *International Conference on Machine Learning*. PMLR. 2019, pp. 3331–3340.

[KB14] Kingma, D. P. and Ba, J. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014).

[KD18] Kingma, D. P. and Dhariwal, P. “Glow: Generative flow with invertible 1x1 convolutions”. In: *arXiv preprint arXiv:1807.03039* (2018).

[KW13] Kingma, D. P. and Welling, M. “Auto-encoding variational bayes”. In: *arXiv preprint arXiv:1312.6114* (2013).

[KSH12] Krizhevsky, A., Sutskever, I., and Hinton, G. E. “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems* 25 (2012), pp. 1097–1105.

[Kru+16] Krueger, D. et al. “Zoneout: Regularizing rnns by randomly preserving hidden activations”. In: *arXiv preprint arXiv:1606.01305* (2016).

[Kül+20] Külebi, B. et al. “CATOTRON–a neural text-to-speech system in Catalan”. In: *Proceedings of Interspeech 2020; 2020 Oct 25-29; Shanghai, China.[Baixas]: ISCA; 2020.* (2020).

[Kum+19] Kumar, K. et al. “Melgan: Generative adversarial networks for conditional waveform synthesis”. In: *arXiv preprint arXiv:1910.06711* (2019).

[Lar+16] Larsen, A. B. L. et al. “Autoencoding beyond pixels using a learned similarity metric”. In: *International conference on machine learning*. PMLR. 2016, pp. 1558–1566.

[LCH17] Lee, J., Cho, K., and Hofmann, T. “Fully character-level neural machine translation without explicit segmentation”. In: *Transactions of the Association for Computational Linguistics* 5 (2017), pp. 365–378.

[Li+19] Li, N. et al. “Neural speech synthesis with transformer network”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. 01. 2019, pp. 6706–6713.

[LY17] Lim, J. H. and Ye, J. C. “Geometric gan”. In: *arXiv preprint arXiv:1705.02894* (2017).

[MYO16] Morise, M., Yokomori, F., and Ozawa, K. “World: a vocoder-based high-quality speech synthesis system for real-time applications”. In: *IEICE TRANSACTIONS on Information and Systems* 99.7 (2016), pp. 1877–1884.

[Oor+18] Oord, A. et al. “Parallel wavenet: Fast high-fidelity speech synthesis”. In: *International conference on machine learning*. PMLR. 2018, pp. 3918–3926.

[Oor+16] Oord, A. v. d. et al. "Wavenet: A generative model for raw audio". In: *arXiv preprint arXiv:1609.03499* (2016).

[Pan+15] Panayotov, V. et al. "Librispeech: an asr corpus based on public domain audio books". In: *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE. 2015, pp. 5206–5210.

[Par+19] Park, D. S. et al. "Specaugment: A simple data augmentation method for automatic speech recognition". In: *arXiv preprint arXiv:1904.08779* (2019).

[PVC19] Prenger, R., Valle, R., and Catanzaro, B. "Waveglow: A flow-based generative network for speech synthesis". In: *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2019, pp. 3617–3621.

[Rec88] Recommendation, C. "Pulse code modulation (PCM) of voice frequencies". In: *ITU*. 1988.

[RM15] Rezende, D. and Mohamed, S. "Variational inference with normalizing flows". In: *International conference on machine learning*. PMLR. 2015, pp. 1530–1538.

[Rud16] Ruder, S. "An overview of gradient descent optimization algorithms". In: *arXiv preprint arXiv:1609.04747* (2016).

[SK16] Salimans, T. and Kingma, D. P. "Weight normalization: A simple reparameterization to accelerate training of deep neural networks". In: *Advances in neural information processing systems* 29 (2016), pp. 901–909.

[She+18] Shen, J. et al. "Natural tts synthesis by conditioning wavenet on mel spectrogram predictions". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2018, pp. 4779–4783.

[She+20] Shen, J. et al. "Non-attentive tacotron: Robust and controllable neural tts synthesis including unsupervised duration modeling". In: *arXiv preprint arXiv:2010.04301* (2020).

[SK19] Shorten, C. and Khoshgoftaar, T. M. "A survey on image data augmentation for deep learning". In: *Journal of big data* 6.1 (2019), pp. 1–48.

[Ske+18] Skerry-Ryan, R. et al. "Towards end-to-end prosody transfer for expressive speech synthesis with tacotron". In: *international conference on machine learning*. PMLR. 2018, pp. 4693–4702.

[Son+20] Song, W. et al. "Efficient WaveGlow: An Improved WaveGlow Vocoder with Enhanced Speed." In: *INTERSPEECH*. 2020, pp. 225–229.

[Sri+14] Srivastava, N. et al. "Dropout: a simple way to prevent neural networks from overfitting". In: *The journal of machine learning research* 15.1 (2014), pp. 1929–1958.

[SGS15] Srivastava, R. K., Greff, K., and Schmidhuber, J. "Highway networks". In: *arXiv preprint arXiv:1505.00387* (2015).

[SWH16] Streijl, R. C., Winkler, S., and Hands, D. S. "Mean opinion score (MOS) revisited: methods and applications, limitations and alternatives". In: *Multimedia Systems* 22.2 (2016), pp. 213–227.

[SVL14] Sutskever, I., Vinyals, O., and Le, Q. V. "Sequence to sequence learning with neural networks". In: *Advances in neural information processing systems*. 2014, pp. 3104–3112.

[SSK17] Suwajanakorn, S., Seitz, S. M., and Kemelmacher-Shlizerman, I. "Synthesizing obama: learning lip sync from audio". In: *ACM Transactions on Graphics (ToG)* 36.4 (2017), pp. 1–13.

[Sze+17] Szegedy, C. et al. "Inception-v4, inception-resnet and the impact of residual connections on learning". In: *Thirty-first AAAI conference on artificial intelligence*. 2017.

[TUA18] Tachibana, H., Uenoyama, K., and Aihara, S. "Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2018, pp. 4784–4788.

[Van+16] Van den Oord, A. et al. "Conditional image generation with pixelcnn decoders". In: *Advances in neural information processing systems* 29 (2016).

[Vas+17] Vaswani, A. et al. "Attention is all you need". In: *Advances in neural information processing systems*. 2017, pp. 5998–6008.

[Vin+15] Vinyals, O. et al. "Grammar as a foreign language". In: *Advances in neural information processing systems* 28 (2015), pp. 2773–2781.

[Wan+17] Wang, Y. et al. "Tacotron: Towards end-to-end speech synthesis". In: *arXiv preprint arXiv:1703.10135* (2017).

[Wan+18] Wang, Y. et al. "Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis". In: *International Conference on Machine Learning*. PMLR. 2018, pp. 5180–5189.

[Wen18] Weng, L. "Flow-based Deep Generative Models". In: *lilianweng.github.io/lil-log* (2018). URL: http://lilianweng.github.io/lil-log/2018/10/13/flow-based-deep-generative-models.html.

[Xie+21] Xie, J. et al. "Decomposition-Based Multistep Sea Wind Speed Forecasting Using Stacked Gated Recurrent Unit Improved by Residual Connections". In: *Complexity* 2021 (2021).

[Xu+20] Xu, J. et al. "Lrspeech: Extremely low-resource speech synthesis and recognition". In: *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2020, pp. 2802–2812.

[YSK20] Yamamoto, R., Song, E., and Kim, J.-M. "Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram". In: *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2020, pp. 6199–6203.

[Zei+10] Zeiler, M. D. et al. "Deconvolutional networks". In: *2010 IEEE Computer Society Conference on computer vision and pattern recognition*. IEEE. 2010, pp. 2528–2535.

[ZTB09] Zen, H., Tokuda, K., and Black, A. W. "Statistical parametric speech synthesis". In: *speech communication* 51.11 (2009), pp. 1039–1064.

[Zen+16] Zen, H. et al. “Fast, compact, and high quality LSTM-RNN based statistical parametric speech synthesizers for mobile devices”. In: *arXiv preprint arXiv:1606.06061* (2016).

[Zho+21] Zhou, H. et al. “Pose-Controllable Talking Face Generation by Implicitly Modularized Audio-Visual Representation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2021, pp. 4176–4186.