Ονοματεπώνυμο: Παναγιώτης Κουνούδης

Αριθμός Μητρώου: 7115182100013

Συστήματα Σύνθεσης και Αναγνώρισης Φωνής (M907)

Στην παρούσα εργασία επιχειρήθηκε να μελετηθεί η επίδραση του α) **γλωσσικού μοντέλου** και β) **του θορύβου**, στην ακρίβεια αναγνώρισης του μοντέλου Mozilla DeepSpeech. Διενεργήθηκαν **δύο (2) πειράματα**, σχετικά με την επίδραση του θορύβου στην ακρίβεια αναγνώρισης του μοντέλου καθώς και με την επίδραση του βάρους του γλωσσικού μοντέλου στην ακρίβεια αναγνώρισης του μοντέλου.

1. DeepSpeech Model

Το μοντέλο DeepSpeech (A. Hannin et al., 2014) είναι ένα end-to-end σύστημα αυτόματης αναγνώρισης ομιλίας, που μετατρέπει τον προφορικό λόγο σε γραπτό κείμενο απευθείας -χωρίς για παράδειγμα την πρόβλεψη φωνημάτων σε ενδιάμεσο στάδιο, αξιοποιώντας νευρωνικά δίκτυα και τεχνικές deep learning. Κάποια από τα χαρακτηριστικά του είναι τα εξής:

- **Convolutional Neural Netoworks (CNN)**. Το μοντέλο χρησιμοποιεί ένα ακουστικό και ένα γλωσσικό μοντέλο. Το ακουστικό μοντέλο αποτελείται από μια στοίβα από convolutional layers, όπου γίνεται η επεξεργασία των ακουστικών χαρακτηριστικών (features). Το μοντέλο έχει γίνει trained ώστε να προβλέπει κατευθείαν τις πιθανότητες του κάθε γράμματος-χαρακτήρα, λαμβάνοντας ακουστικό input.
- **Recurrent Neural Network (RNN)**. Το γλωσσικό μοντέλο βασίζεται στα RNNs και χρησιμοποιείται για να βελτιώσει τις μεταγραφές που κάνει το ακουστικό μοντέλο. Έχει γίνει trained σε πολλά γλωσσικά δεδομένα για να μάθει τις πιθανότητες ακολουθιών των λέξεων και λειτουργεί βοηθητικά στο decoding process. Παίρνει σαν input spectrograms και εξάγει transcriptions.
- **Connectionist Temporal Classification (CTC) Training**. Το CTC χρησιμοποιείται για την εκπαίδευση του μοντέλου. Το μοντέλο μαθαίνει τα απαραίτητα alignments μεταξύ του audio input της μεταγραφής του σε γράμμα-χαρακτήρα, κατά τη διάρκεια της εκπαίδευσης, χωρίς κάποιο labeling των δεδομένων σε προηγούμενο στάδιο.
- **Greedy Decoding and Beam Search**. Κατά τη διάρκεια του inference το DeepSpeech μετατρέπει ακουστικές αναπαραστάσεις σε ακολουθίες γραμμάτων-χαρακτήρων. Στο decoding χρησιμοποιείται Greedy αλγόριθμος και έπειτα γίνεται beam search, ώστε να ληφθούν υπόψιν, πολλαπλές υποψήφιες ακολουθίες χαρακτήρων και να βελτιωθεί η ποιότητα της μεταγραφής.

- **Training data**. Το DeepSpeech έχει εκπαιδευτεί σε τεράστιες ποσότητες γλωσσικών δεδομένων σε διάφορες γλώσσες (Αγγλικά, Κινεζικά κ.α.). Τα δεδομένα εκπαίδευσης έχουν τη μορφή ζευγών audio και αντίστοιχων ακολουθιών χαρακτήρων.

2. Δεδομένα Φωνής και Δημιουργία Δεδομένων με θόρυβο

Το corpus που χρησιμοποιήθηκε είναι το Mini LibreSpeech. Τα αρχεία (αρχικά συμπιεσμένα ώς .flac) αποσυμπιέστηκαν σε .wav με sampling rate 16000, με το ffmpeg. Το dataset περιέχει ομιλίες από 26 ομιλητές. Χρησιμοποιήθηκαν 5 αρχεία από κάθε ομιλητή, συνολικά 130 αρχεία.

Κάθε αρχείο ομιλητή, ενσωματώθηκε με θόρυβο. Έτσι για κάθε αρχείο ομιλητή, προέκυψαν τρία αρχεία με θόρυβο και signal to noise ratio 3, 6 και 9 dB. Χρησιμοποιήθηκαν 5 τυχαία διαφορετικά αρχεία θορύβου από το dataset DEMAND. Συγκεκριμένα, χρησιμοποιήθηκαν τα DKITCHEN, NFIELD, NRIVER, PCAFETER, TMETRO, με το αρχείο και το τμήμα του θορύβου να είναι ίδιο για κάθε συγκεκριμένο αρχείο φωνής. Προέκυψαν 520 νέα αρχεία.

Για να γίνει η ενσωμάτωση θορύβου με το αρχείο του ομιλητή, υπολογίστηκε ο συντελεστής με τον οποίο πολλαπλασιάστηκε το τμήμα θορύβου με τον εξής τρόπο: Χρησιμοποιήθηκε η ενέργεια του σήματος ομιλίας (υπολογίστηκε βρίσκοντας το mean των squared values του), η ενέργεια του σήματος θορύβου και το SNR που θέλαμε κάθε φορά.

```
# Calculate the power of the speech and noise signals
power_speech = np.mean(wav1 ** 2)
power_noise = np.mean(wav2 ** 2)

# Calculate the desired power of the noise signal based on the
target_snr = 10 ** (target_snr_dB / 10)
target_power_noise = power_speech / target_snr

# Calculate the scale factor to achieve the desired SNR
scale_factor = np.sqrt(target_power_noise / power_noise)

# Multiply the noise signal by the scale factor
scaled_wav2 = wav2 * scale_factor

# Merge the two WAV files
merged_wav = wav1 + scaled_wav2
```

Διαιρούμε την επιθυμητή ενέργεια του τμήματος θορύβου με την αρχική ενέργεια του τμήματος θορύβου, και πολλαπλασιάζουμε το numpy array-αναπαράσταση του θορύβου με τον συντελεστή που προκύπτει. Έπειτα προσθέτουμε το scaled wav που προκύπτει με το wav της ομιλίας.

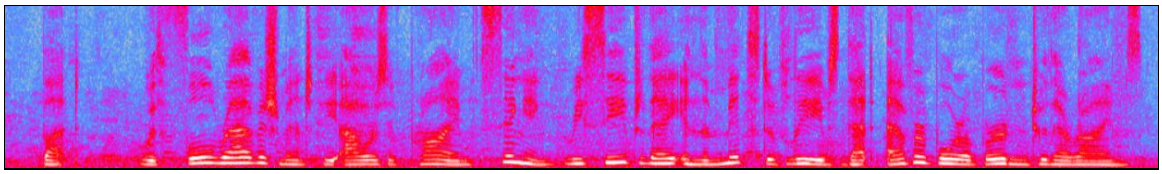

Figure 1 Φασματογράφημα ομιλίας με θόρυβο και Signal-to-noise ratio 9dB.

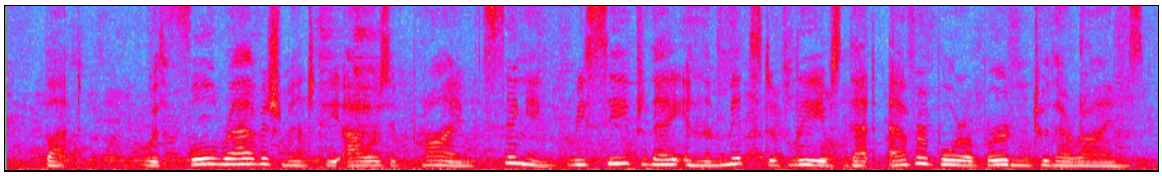

Figure 2 Φασματογράφημα ομιλίας με θόρυβο και Signal-to-noise ratio 6dB.

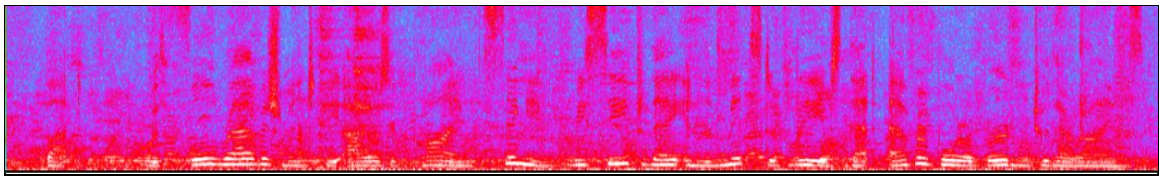

Figure 3 Φασματογράφημα ομιλίας με θόρυβο και Signal-to-noise ratio 3dB.

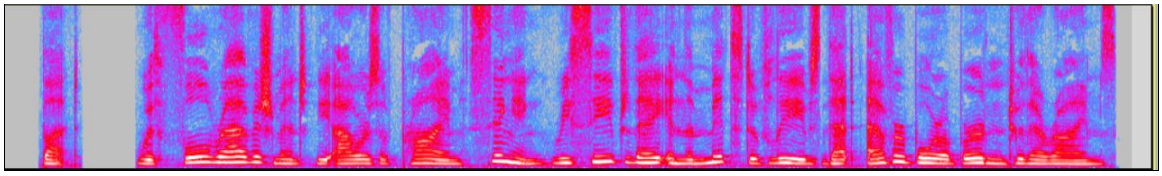

Figure 4 Φασματογράφημα ομιλίας χωρίς θόρυβο

3. Πειράματα

Α) Επίδραση θορύβου στην ακρίβεια αναγνώρισης

Έγινε ανάλυση των 130 αρχείων και υπολογίστηκε το WER καθώς και ο αριθμός των Substitutions, Deletions και Insertions, για τα αρχικά αρχεία και μετά για τα αρχεία με SNR 3dB, 6dB και 9dB.

Καταρτίστηκε πίνακας, τμήμα του οποίου αναφέρεται εδώ. Στην αριστερή στήλη φαίνεται ο αριθμός κάθε ομιλητή, και το σύνολο WER, Subs, Dels και Ins, για κάθε 5άδα αρχείων/ομιλητή. Υπολογίστηκε, και ο μέσος όρος των WER, Subs, Dels και Ins σε όλα τα αρχεία και καταρτίστηκε αντίστοιχο πινακάκι. Τέλος, δημιουργήθηκε διάγραμμα με το αντίστοιχο WER, για τα καθαρά αρχεία και τα αρχεία με θόρυβο. Παρατηρείται ότι ο θόρυβος επιδρά σημαντικά στην ακρίβεια αναγνώρισης του μοντέλου με το WER να είναι σημαντικά πιο υψηλό στα αρχεία με SNR 3dB, σε αντίθεση με τα αρχεία χωρίς θόρυβο ή στα αρχεία με SNR 6dB και 9dB.

7850				
Clean files	0.073	4	0	0
3dB files	0.073	2	2	0
6dB files	0.018	0	1	0
9dB files	0.036	1	1	0
7976				
Clean files	0.034	5	0	0
3dB files	0.096	11	2	1
6dB files	0.034	5	0	0
9dB files	0.055	6	0	2
8297				
Clean files	0.04	3	1	0
3dB files	0.051	3	2	0
6dB files	0.04	2	2	0
9dB files	0.04	3	1	0
8842				
Clean files	0.218	15	4	0
3dB files	0.264	18	5	0
6dB files	0.241	19	2	0
9dB files	0.23	18	2	0
Average Clean files	**0.0702**	**5.4**	**0.84**	**0.6**
Average 3db files	**0.1276**	**9.12**	**3.16**	**0.68**
Average 6db files	**0.08664**	**6.8**	**1.28**	**0.48**
Average 9db files	**0.07648**	**5.96**	**0.88**	**0.64**

Figure 5 Πίνακας με αποτελέσματα ανά ομιλητή και ανά βαθμίδα θορύβου.

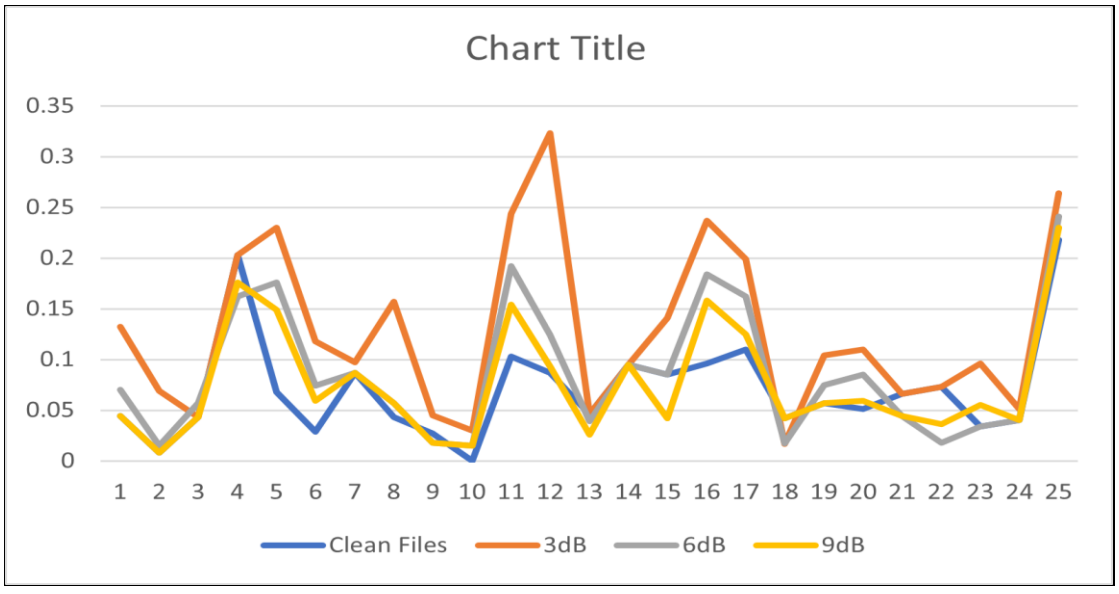

Figure 6 Διάγραμμα με το WER και πως αυτό αλλάζει ανάλογα με τη βαθμίδα θορύβου.

Β) Επίδραση βάρους του γλωσσικού μοντέλου

Λόγω έλλειψης χρόνου χρησιμοποιήθηκαν για την ανάλυση τα αρχεία χωρίς θόρυβο μόνο από πέντε (5) ομιλητές. Για τον ίδιο λόγο τροποποιήθηκαν μόνο με μια τιμή τα lm_alpha και lm_beta. Στην πρώτη περίπτωση έμεινε σταθερό το lm_beta και το lm_alpha έγινε 0.20, ενώ στη δεύτερη περίπτωση έμεινε σταθερό το lm_alpha και το lm_beta έγινε 0.20. Καταρτίστηκε το παρακάτω πινακάκι, με τους πέντε (5) ομιλητές.

84	WER	Subs	Dels	Ins
a_020	0.096	7	0	4
b_020	0.018	1	1	0
174	WER	Subs	Dels	Ins
a_020	0.015	2	0	0
b_020	0.008	1	0	0
251	WER	Subs	Dels	Ins
a_020	0.1	4	0	3
b_020	0.043	2	0	1
777	WER	Subs	Dels	Ins
a_020	0.257	15	1	3
b_020	0.189	10	3	1
1272	WER	Subs	Dels	Ins
a_020	0.122	7	1	1
b_020	0.081	5	1	0
Average a_020	**0.118**	**7**	**0.4**	**2.2**
Average b_020	**0.0678**	**3.8**	**1**	**0.4**

Figure 7 Πίνακας με αποτελέσματα 2ου πειράματος.

Επιπλέον, δημιουργήθηκε διάγραμμα με το WER, για κάθε περίπτωση. Όταν το lm_alpha είναι ίσο με 0.20 και σταθερό το lm_beta, και όταν το lm_beta είναι ίσο με 0.20 και σταθερό το lm_alpha.

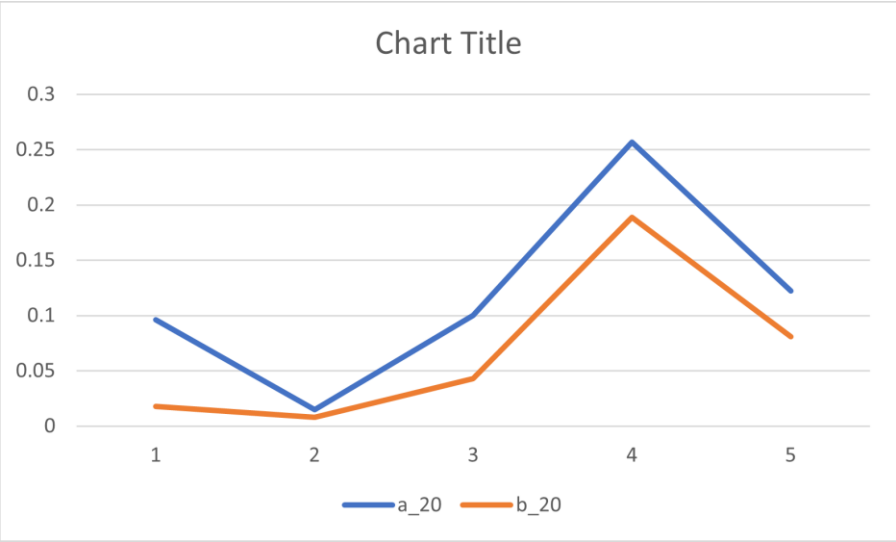

Figure 8 Διάγραμμα με το WER ανά τροποποίηση σε lm_alpha και lm_beta.

Μειώνοντας την τιμή του lm_alpha, περιμένουμε να επηρεαστεί το γλωσσικό μοντέλο του DeepSpeech. Το transcription βασίζεται περισσότερο στο ακουστικό μοντέλο, και υπάρχει μεγαλύτερη πιθανότητα να δημιουργηθούν λάθη η λιγότερο fluent μεταγραφές. Παρατηρώντας τον πίνακα, επαληθεύουμε την παραπάνω υπόθεση, εφόσον έχουμε ένα σχετικά υψηλό WER (Average: 0.118).

Αντιθέτως, παρατηρούμε ότι μειώνοντας το lm_beta, δεν παρατηρούμε τόσο σημαντική αύξηση στις τιμές του WER. Το μπόνους insertion των λέξεων, δεν φαίνεται να επηρεάζει τόσο σημαντικά την δυνατότητα του μοντέλου να παράγει ακριβείς μεταγραφές, όσο επηρεάζει η μείωση της βαρύτητας του γλωσσικού μοντέλου.

ΤΕΛΟΣ