ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ-ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Επεξεργασία Σημάτων Φωνής και Ήχου

Από τους φοιτητές:

Αρζουμάν Άγγελος Π17009

Κορώνης Ευάγγελος Π17050

Λαμπίρης Δημήτριος Π17061

Τσιάκας Δημήτριος Π17151

Περιεχόμενα

Παραδοτέα αρχεία 3
Θέμα 1ο 3
Εισαγωγή 3
Γλώσσα προγραμματισμού και βιβλιοθήκες που χρησιμοποιήθηκαν 3
Περιγραφή του συστήματος αλγοριθμικά 3
Ταξινομητής background vs foreground 4
Επίδοση ταξινομητή foreground vs background 5
Εξαγωγή χαρακτηριστικών 5
Αναγνώριση ψηφίων 6
Ακρίβεια KNN ταξινομητή 6
Θέμα 2ο 6

Παραδοτέα αρχεία

Για την παρούσα εργασία παραδίδονται:

- Το αρχείο πηγαίου κώδικα digit_recognition.py , που είναι συμπιεσμένο στο αρχείο source2021.zip
- Το συμπιεσμένο αρχείο auxiliary2021.zip, που περιλαμβάνει τους φακέλους bVSf_test_data, knn_training_data και recordings που περιέχουν αρχεία ήχου.
- Τα αρχείο ergasia-thema-2.csv με το θέμα 2.
- Το παρών pdf αρχείο με την τεκμηρίωση της εργασίας.

Για να τρέξετε το πρόγραμμα κάντε αποσυμπίεση τα αρχεία source2021.zip και auxiliary2021.zip στο ίδιο μέρος/φάκελο.

Θέμα 1ο

Εισαγωγή

Η ομάδα μας υλοποίησε ένα ASR σύστημα το οποίο δέχεται ως είσοδο μία ηχογράφηση κάθε φορά, η οποία συνιστά πρόταση αποτελούμενη από 5-10 ψηφία της Αγγλικής γλώσσας που έχουν ειπωθεί με αρκούντως μεγάλα διαστήματα παύσης. Στην συνέχεια το σύστημα προχωρά στην κατάτμηση της πρότασης χρησιμοποιώντας έναν ταξινομητή background vs foreground τον οποίο υλοποιήσαμε εμείς. Τα ψηφία που προκύπτουν από την ταξινόμηση τα αναγνωρίζουμε με την βοήθεια ενός KNN ταξινομητή της βιβλιοθήκης sklearn.

Γλώσσα προγραμματισμού και βιβλιοθήκες που χρησιμοποιήθηκαν

Η υλοποίηση έγινε σε γλώσσα python 3.7. Οι βιβλιοθήκες που χρησιμοποιούνται και πρέπει να είναι εγκατεστημένες είναι οι εξής:

- librosa
- matplotlib
- numpy
- sounddevice
- sklearn
- os
- scipy
- time
- sys

Περιγραφή του συστήματος αλγοριθμικά

Στην αρχή το πρόγραμμα ζητάει από τον χρήστη να διαλέξει ένα φωνητικό αρχείο για αναγνώριση. Τα αρχεία αυτά είναι τύπου wav και βρίσκονται στον φάκελο recordings. Ο ρυθμός δειγματοληψίας που χρησιμοποιούμε είναι ίσος με 44100.

Ταξινομητής background vs foreground

Το αρχείο που επίλεξε ο χρήστης για αναγνώριση, το χωρίζουμε σε παράθυρα μήκους 2048 δειγμάτων, με κάθε παράθυρο να απέχει 1024 δείγματα από το προηγούμενο. Για κάθε παράθυρο εξάγουμε το zero crossing rate, την ενέργεια του και την βασική συχνότητά του. Ο τύπος εύρεσης του zero crossing rate δίνεται παρακάτω.

$$zero\ crossing\ rate = \frac{1}{2 * \mu\acute{\eta}κος\ παραθύρου} * \sum_{i=1}^{\mu\acute{\eta}κος\ παραθύρου} |sgn(x[i]) - sgn(x[i-1])|$$

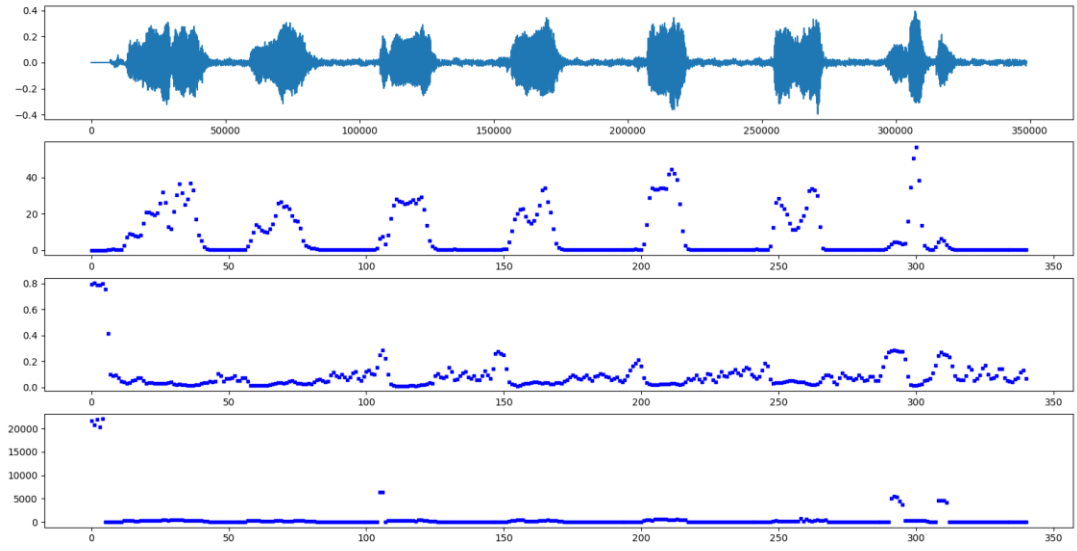
, όπου $sgn(x) = \begin{cases} 1, x \geq 0 \\ -1, x < 0 \end{cases}$

Ο τύπος εύρεσης της ενέργειας δίνεται παρακάτω.

$$ενέργεια = \sum_{i=1}^{\mu\acute{\eta}κος\ παραθύρου} x[i]^2$$

Για την εύρεση της βασικής συχνότητας του παραθύρου βρίσκουμε τον DFT του, και παίρνουμε την συχνότητα που έχει την μεγαλύτερη ισχύς.

Παρακάτω βλέπουμε τα διαγράμματα με τις τιμές των τριών χαρακτηριστικών που εξάγουμε για κάθε παράθυρο. Στο πρώτο διάγραμμα βλέπουμε την απεικόνιση του σήματος στον χρόνο. Στο δεύτερο διάγραμμα βλέπουμε τις τιμές της ενέργειας του κάθε παραθύρου. Στο τρίτο διάγραμμα βλέπουμε τις τιμές του zero crossing rate για κάθε παράθυρο και στο τέταρτο την βασική συχνότητα κάθε παραθύρου.



Όπως φαίνεται από τα διαγράμματα, όπου υπάρχει ομιλία η ενέργεια είναι υψηλότερη από εκεί που δεν υπάρχει, ενώ το zero crossing rate είναι χαμηλό. Η συχνότητες των παραθύρων γενικά κυμαίνονται σε χαμηλά επίπεδα , αλλά σε παράθυρα με υψηλό zero crossing rate παρατηρούμε και υψηλές συχνότητες.

Έχοντας εξάγει τα τρία αυτά χαρακτηριστικά, κατηγοριοποιούμε κάθε παράθυρο είτε ως παράθυρο background ή ως παράθυρο foreground.

Ένα παράθυρο είναι foreground όταν ισχύουν και οι τρεις παρακάτω συνθήκες:

- το zero crossing rate του είναι μικρότερο από τα 3/2 του μέσου όρου των zero crossing rate όλων των παραθύρων,
- η ενέργεια του είναι μεγαλύτερη από το μισό του μέσου όρου της ενέργειας όλων των παραθύρων,
- η βασική του συχνότητα του είναι μικρότερη από 3400.

Αν κάποιο από αυτά δεν ισχύει τότε το παράθυρο θεωρείται ως παράθυρο background.

Αφού ταξινομήσουμε κάθε παράθυρο, παίρνουμε μία λίστα μήκους όσο και ο αριθμός των παραθύρων που χωρίσαμε το ηχητικό. Οι τιμές της λίστας είναι 0 και 1. Με 0 συμβολίζονται τα background παράθυρα και με 1 τα παράθυρα foreground.

Στην λίστα εφαρμόζουμε ένα φίλτρο μέσου με παράθυρο μήκους 5 ώστε να διορθώσουμε λάθη που μπορεί να έχουν προκύψει από την ταξινόμηση.

Στην συνέχεια, με βάση τα παράθυρα foreground που βρήκαμε, εξάγουμε τα μέρη του ηχητικού στα οποία υπάρχει ομιλία.

Ως τελευταίο μέτρο αποφυγής λαθών, απορρίπτουμε όσα κομμάτια foreground ήχου έχουν διάρκεια μικρότερη από 100 ms .

Επίδοση ταξινομητή foreground vs background

Για να μετρήσουμε την επίδοση του ταξινομητή δημιουργήσαμε 10 ηχητικά, ένα για κάθε ψηφίο, τα οποία βρίσκονται στον φάκελο bVSf_test_data. Σε κάθε ηχητικό υπάρχει διάστημα επαρκούς χρόνου με background ήχο. Τα ονόματα των αρχείων των ηχητικών είναι της μορφής X_Y.wav, όπου X είναι ο αριθμός του δείγματος στο οποίο ξεκινάει να προφέρεται το ψηφίο και Y ο αριθμός του δείγματος που τελειώνει η εκφώνηση του ψηφίου.

Ταξινομούμε κάθε δείγμα του κάθε ηχητικού ως background ή ως foreground με την διαδικασία που περιεγράφηκε προηγουμένως και βρίσκουμε το ποσοστό επιτυχών προβλέψεων με βάση τον X και Y αριθμό.

Στο τέλος βρίσκουμε το μέσο ποσοστό επιτυχίας για όλα τα αρχεία, το οποίο είναι περίπου 86%. Το αποτέλεσμα εμφανίζεται στο τέλος της εκτέλεσης του προγράμματος.

```
Accuracy of background vs foreground: 0.8631441157195209
```

Εξαγωγή χαρακτηριστικών

Για κάθε ψηφίο που βρήκαμε από την background vs foreground ταξινόμηση εφαρμόζουμε ένα ζωνοπερατό φίλτρο ώστε να εξαλείψουμε τις συχνότητες κάτω των 400HZ και πάνω των 3400HZ και εξάγουμε το mel spectrogram. Για το mel spectrogram χρησιμοποιούμε παράθυρο μήκους 2048 δειγμάτων με κάθε παράθυρο να απέχει 1024 δείγματα από το προηγούμενο. Επίσης ορίζουμε ως ανώτατη συχνότητα τα 5000 HZ και χωρίζουμε τις

συχνότητες σε 128 mel bins. Επιπλέον η ισχύς των συχνοτήτων μετριέται σε μονάδες ντεσιμπέλ.

Για κάθε συχνότητα του mel spectrogram βρίσκουμε την μέγιστη τιμή της. Έτσι καταλήγουμε με έναν μονοδιάστατο πίνακα με 128 τιμές ,όσες είναι και οι συχνότητες που μετρήσαμε. Αυτός ο πίνακας αποτελεί και το διάνυσμα των χαρακτηριστικών που θα χρησιμοποιήσουμε για την αναγνώριση του κάθε ψηφίου που πρόφερε ο χρήστης.

Στην συνέχεια δημιουργούμε τα δεδομένα εκπαίδευσης του KNN ταξινομητή που θα χρησιμοποιήσουμε για αναγνώριση. Για την εκπαίδευση δημιουργήσαμε 160 ηχητικά. Σε κάθε ηχητικό ακούγεται ένα ψηφίο, με κάθε μέλος της ομάδας να έχει ηχογραφήσει το κάθε ψηφίο τουλάχιστον 3 φορές. Τα ηχητικά βρίσκονται στον φάκελο knn_training_data.

Για την εξαγωγή χαρακτηριστικών από τα δεδομένα εκπαίδευσης, ακολουθούμε την ίδια διαδικασία που ακολουθήσαμε για την εξαγωγή χαρακτηριστικών από τα ψηφία που προορίζονται για αναγνώριση. Δηλαδή για κάθε ηχητικό εφαρμόζουμε ένα ζωνοπερατό φίλτρο και εξάγουμε το mel spectrogram με τις ίδιες παραμέτρους που χρησιμοποιήσαμε και πριν. Για κάθε mel συχνότητα του spectrogram εξάγουμε την μέγιστη τιμή.

Τέλος κανονικοποιούμε τα δεδομένα εκπαίδευσης και αναγνώρισης ώστε να πετύχουμε μεγαλύτερη απόδοση στην ταξινόμηση.

Αναγνώριση ψηφίων

Τα κανονικοποιημένα, πλέον, δεδομένα εκπαίδευσης τα εισάγουμε στον ταξινομητή για την εκπαίδευση του.

Για την ταξινόμηση χρησιμοποιούμε έναν KNN ταξινομητή της βιβλιοθήκης sklearn. Ο ταξινομητής αυτός ταξινομεί σε αύξουσα σειρά τα διανύσματα εκπαίδευσης με βάση την απόσταση Minkowski που έχουν με το προς αναγνώριση διάνυσμα. Στο διάνυσμα αυτό ανατίθεται η κλάση η οποία επικρατεί στα 3 πιο κοντινά προς αυτό διανύσματα.

Ακρίβεια KNN ταξινομητή

Για να υπολογίσουμε την ακρίβεια του ταξινομητή χρησιμοποιούμε την μέθοδο K Fold cross validation. Συγκεκριμένα ανακατεύουμε τα δεδομένα εκπαίδευσης και τα χωρίζουμε σε 3 ομάδες. Στην συνέχεια εκτελούμε την διαδικασία της εκπαίδευσης και της ταξινόμησης 3 φορές, ώστε κάθε ομάδα να έχει γίνει μια φορά η ομάδα προς ταξινόμηση ενώ οι άλλες δύο χρησιμοποιούνται για την εκπαίδευση. Για κάθε επανάληψη βρίσκουμε την ακρίβεια της ταξινόμησης και στο τέλος βρίσκουμε τον μέσο όρο των επαναλήψεων. Το αποτέλεσμα εμφανίζεται στο τέλος της εκτέλεσης του προγράμματος και κυμαίνεται μεταξύ 46-56%

```
Knn prediction score: 0.48765432098765427
```

Θέμα 2ο

Για το θέμα 2 ακούσαμε 100 ηχητικά προκειμένου να εντοπίσουμε τα συμβάντα ήχου που ακούγονται. Τα συμβάντα αυτά τα καταγράψαμε στο αρχείο ergasia-thema-2.csv.