ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Επεξεργασία Φωνής και Φυσικής Γλώσσας

Προπαρασκευή 2ου Εργαστηρίου:

Αναγνώριση φωνής με το KALDi TOOLKiT

Αναφορά

Μαρκος Γιαννόπουλος 03118103

Νικόλαος Καραφύλλης 03119890

Μάιος 2023

ΠΡΟΠΑΡΑΣΚΕΥΗ

Αφού εγκαταστήσουμε το εργαλείο Kaldi, κατεβάσουμε τα δεδομένα και δημιουργήσουμε τους φακέλους που ζητούνται, δημιουργούμε τα αρχεία:

- utt2spk μέσω του bash script make_utt2spk.sh:

```bash
#!/bin/bash

# Define the list of input and output file pairs
input_files=("./data/train/uttids" "./data/test/uttids" "./data/dev/uttids")
output_files=("./data/train/utt2spk" "./data/test/utt2spk" "./data/dev/utt2spk")

# Loop through each input/output file pair
for i in "${!input_files[@]}"; do

  # Initialize line counter
  line_num=1

  # Loop through each line in the input file
  while read line; do

    # Take the first two characters of the line
    speaker_id="${line:0:2}"

    # Combine line number with "utterance_id"
    utterance_id="utterance_id_${line_num}"

    # Write the new line to the output file
    echo "$utterance_id $speaker_id" >> "${output_files[$i]}"

    # Increment the line number counter
    line_num=$((line_num+1))

  done < "${input_files[$i]}"

done
```

οπότε δημιουργούνται αρχεία της μορφής :

```
utterance_id_1 f1
utterance_id_2 f1
utterance_id_3 f1
utterance_id_4 f1
```

- wav.scp μέσω του script make_wav.scp.sh:

```bash
#!/bin/bash

# Define the list of input and output file pairs
input_files=("./data/train/uttids" "./data/test/uttids" "./data/dev/uttids")
output_files=("./data/train/wav.scp" "./data/test/wav.scp" "./data/dev/wav.scp")

# Loop through each input/output file pair
for i in "${!input_files[@]}"; do

  # Initialize line counter
  line_num=1

  # Loop through each line in the input file
  while read line; do

    # Split the line into an array of strings
    IFS=' ' read -r -a strings <<< "$line"

    # Combine line number with "utterance_id"
    utterance_id="utterance_id_${line_num}"
    path="./wav/${strings[0]}.wav"

    # Write the new line to the output file
    echo "$utterance_id $path" >> "${output_files[$i]}"

    # Increment the line number counter
    line_num=$((line_num+1))

  done < "${input_files[$i]}"

done
```

οπότε δημιουργούνται αρχεία της μορφής :

```
utterance_id_1 ./wav/f1_003.wav
utterance_id_2 ./wav/f1_004.wav
utterance_id_3 ./wav/f1_005.wav
utterance_id_4 ./wav/f1_007.wav
utterance_id_5 ./wav/f1_008.wav
utterance_id_6 ./wav/f1_009.wav
utterance_id_7 ./wav/f1_010.wav
```

- text μέσω του script make_text.sh:

```
#!/bin/bash

# Define the list of input and output file pairs
input_files=("./data/train/uttids" "./data/test/uttids" "./data/dev/uttids")
output_files=("./data/train/text" "./data/test/text" "./data/dev/text")

transcriptions_file=("./transcriptions.txt")

# Loop through each input/output file pair
for i in "${!input_files[@]}"; do

  # Initialize line counter
  line_num=1

  # Loop through each line in the input file
  while read line; do

    # Split the line into an array of strings
    IFS='_' read -r -a strings <<< "$line"

    # Combine line number with "utterance_id"
    utterance_id="utterance_id_${line_num}"

    search_string="${strings[1]}"

    # Search for lines where the first string is the search_string
    result=$(grep "^${search_string}" "${transcriptions_file}")

    # If a matching line is found, extract the rest of the string
    if [ -n "${result}" ]; then
        text=$(echo "${result}" | cut -d$'\t' -f2-)
    else
        echo "No matching line found"
    fi

    # Write the new line to the output file
    echo "$utterance_id $text" >> "${output_files[$i]}"

    # Increment the line number counter
    line_num=$((line_num+1))

  done < "${input_files[$i]}"

done
```

οπότε δημιουργούνται αρχεία της μορφής :

```
utterance_id_1 She is thinner than I am.
utterance_id_2 Bright sunshine shimmers on the ocean.
utterance_id_3 Nothing is as offensive as innocence.
```

και για το τελευταίο βήμα μετατρέπουμε τις λέξεις των προτάσεων των αρχειων text σε φωνήματα μέσω του script words2phonemes.sh :

```bash
#!/bin/bash

# Define the list of input and output file pairs
input_files=("./data/train/text" "./data/test/text" "./data/dev/text")
output_files=("./data/train/phonemes" "./data/test/phonemes" "./data/dev/phonemes")

lexicon_file=("./lexicon.txt")

# Loop through each input/output file pair
for i in "${!input_files[@]}"; do

    # Loop through each line in the input file
    while read -r line; do

        # Split line into utterance ID and sentence
        utterance_id=${line%% *}
        sentence=${line#* }

        # Make sentence uppercase
        sentence=$(echo "$sentence" | tr '[:lower:]' '[:upper:]')
        # Remove special characters except single quotes
        sentence=$(echo "$sentence" | tr -cd "[:alnum:][:space:]'\-'"| sed 's/-/ /g')
        # Split sentence into words
        words=($sentence)

        # Replace each word with its corresponding phonemes from the lexicon
        phonemes=()
        for word in "${words[@]}"; do
            #echo ${word}
            # Search for lines where the first string is the search_string
            result=$(grep "^${word}[[:space:]]" "${lexicon_file}")

            # If a matching line is found, extract the rest of the string
            if [ -n "${result}" ]; then
                phonemes+=$(echo "${result}" | cut -d$'\t' -f2-)" "
            else
                echo "No matching line found for word: " $word
            fi
        done

        # Write new sentence to output file
        echo "$utterance_id "sil" "${phonemes[@]}" "sil"" >> "${output_files[$i]}"

    done < "${input_files[$i]}"

done
```

οπότε δημιουργούνται τα αρχεία phonemes που έχουν την μορφή:

```
utterance_id_1 sil  sh iy ih z th ih n er dh ae n ay ae m  sil
utterance_id_2 sil  b r ay t s ah n sh ay n sh ih m er z aa n dh ah dh iy ow sh ah n  sil
utterance_id_3 sil  n ah th ih ng ih z ae z ah f eh n s ih v ae z ih n ah s ah n s  sil
```

ΤΕΛΟΣ ΠΡΟΠΑΡΑΣΚΕΥΗΣ