Myprosody

Example usage

Gender recognition and mood of speech: Function *myspgend(p,c)*

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)

mysp.myspgend(p,c)

[] a female, mood of speech: Reading, p-value/sample size= :0.00 5
```

Pronunciation posteriori probability score percentage: Function *mysppron(p,c)*

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)

mysp.mysppron(p,c)

[]Pronunciation_posteriori_probability_score_percentage= :85.00
```

Detect and count number of syllables: Function *myspsyl(p,c)*

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)

mysp.myspsyl(p,c)

[]number_ of_syllables= 154
```

Detect and count number of fillers and pauses: Function *mysppaus(p,c)*

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)

mysp.mysppaus(p,c)

[]number_of_pauses= 22
```

Measure the rate of speech (speed): Function *myspsr(p,c)*

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)

mysp.myspsr(p,c)

[]rate_of_speech= 3 # syllables/sec original duration
```

Measure the articulation (speed): Function *myspatc(p,c)*

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)

mysp.myspatc(p,c)

[]articulation_rate= 5 # syllables/sec speaking duration
```

Measure speaking time (excl. fillers and pause): Function *myspst(p,c)*

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)

mysp.myspst(p,c)

[]speaking_duration= 31.6 # sec only speaking duration without pauses
```

Measure total speaking duration (inc. fillers and pauses): Function *myspod(p,c)*

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)

mysp.myspod(p,c)

[]original_duration= 49.2 # sec total speaking duration with pauses
```

Measure ratio between speaking duration and total speaking duration: Function *myspbala(p,c)*

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)

mysp.myspbala(p,c)

[]balance= 0.6 # ratio (speaking duration)/(original duration)
```

Measure fundamental frequency distribution mean: Function *myspf0mean(p,c)*

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)

mysp.myspf0mean(p,c)

[]f0_mean= 212.45 # Hz global mean of fundamental frequency distribution
```

Measure fundamental frequency distribution SD: Function *myspf0sd(p,c)*

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)

mysp.myspf0sd(p,c)

[]f0_SD= 57.85 # Hz global standard deviation of fundamental frequency
distribution
```

Measure fundamental frequency distribution median: Function *myspf0med(p,c)*

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)

mysp.myspf0med(p,c)

[]f0_MD= 205.7 # Hz global median of fundamental frequency distribution
```

Measure fundamental frequency distribution minimum: Function
myspf0min(p,c)

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)

mysp.myspf0min(p,c)

[]f0_min= 77 # Hz global minimum of fundamental frequency distribution
```


Measure fundamental frequency distribution maximum: Function
myspf0max(p,c)

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)

mysp.myspf0max(p,c)

[]f0_max= 414 # Hz global maximum of fundamental frequency distribution
```


Measure 25th quantile fundamental frequency distribution: Function
myspf0q25(p,c)

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)

mysp.myspf0q25(p,c)

[]f0_quan25= 171 # Hz global 25th quantile of fundamental frequency distribution
```


Measure 75th quantile fundamental frequency distribution: Function
myspf0q75(p,c)

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)

mysp.myspf0q75(p,c)

[]f0_quan75= 244 # Hz global 75th quantile of fundamental frequency distribution
```

Overview: Function *mysptotal(p,c)*

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)

mysp.mysptotal(p,c)
```

```
number_ of_syllables      154

number_of_pauses           22

rate_of_speech              3

articulation_rate           5

speaking_duration        31.6

original_duration        49.2

balance                   0.6

f0_mean                212.45

f0_std                  57.85

f0_median               205.7

f0_min                     77

f0_max                    414

f0_quantile25             171

f0_quan75                 244
```

Compared to native speech, here are the prosodic features of your speech: Function *mysp.myprosody(p,c)*

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)
```

mysp.myprosody(p,c)

Compared to native speech, here are the prosodic features of your speech:

average_syll_pause_duration: (:Out of Range)

No._long_pause: 25.000000 (% percentile)

speaking_time: 25.000000 (% percentile)

```
ave_No._of_words_in_minutes:          (:Out of Range)

articulation_rate:            66.666667 (% percentile )

No._words_in_minutes:         25.000000 (% percentile )

formants_index:               25.000000 (% percentile )

f0_index:                     66.666667 (% percentile )

f0_quantile_25_index:         (:Out of Range)

f0_quantile_50_index:         (:Out of Range)

f0_quantile_75_index:          33.333333 (% percentile )

f0_std:                       25.000000 (% percentile )

f0_max:                       33.333333 (% percentile )

f0_min:                       (:Out of Range)

No._detected_vowel:           25.000000 (% percentile )

perc%._correct_vowel:         66.666667 (% percentile )

(f2/f1)_mean:                 66.666667 (% percentile )

(f2/f1)_std:                  33.333333 (% percentile )

no._of_words:                 25.000000 (% percentile )

no._of_pauses:                25.000000 (% percentile )

intonation_index:             33.333333 (% percentile )

(voiced_syll_count)/(no_of_pause): 66.666667 (% percentile )

TOEFL_Scale_Score:             66.666667 (% percentile )

Score_Shannon_index:           66.666667 (% percentile )

speaking_rate:                 25.000000 (% percentile )
```

Spoken Language Proficiency Level estimator, based on **Machine Learning models** of the prosodic features of your speech:

Function mysp.mysplev(p,c)

```python
import myprosody as mysp

p="Walkers" # Audio File title
c=r"C:\Users\Shahab\Desktop\Mysp" # Path to the Audio_File directory (Python 3.7)


mysp.mysplev(p,c)

[out]

    58% accuracy    ['c']
```

```
65% accuracy      ['c']

70% accuracy      ['c']

67% accuracy      ['c']

64% accuracy      ['c']

63% accuracy      ['c']
```

Development

MYPROSODY was developed by Mysolution Lab in Japan. It is part of New Generation of Voice Recognition and Acoustic & Language modelling Project in Mysolution Lab. That is planned to enrich the functionality of Myprosody by adding more advanced functions.