

Phonetics – 물리적 소리 그 자체

Phonology – 물리적 소리를 머리로 인지하는 과정(sound system)

Phonetics

articulatory : 소리를 만드는 원리에 관한 것 (from mouth)

acoustic : 소리가 공기를 타고 가는 것 (through air)

auditory : 소리를 귀로 듣는 것 (to ear)

articulatory

1. phonation process

Larynx open -> vocal cord vibration (x) -> voiceless

Larynx closed -> vocal cord vibration (o) -> voiced

2. oro-nasal process

Velum lowered (nasal tract open) -> 비음 (m, n, ŋ), 코로 숨쉴 때

Velum raised (nasal tract closed) -> 모든 모음, 비음을 뺀 모든 자음

3. articulatory process

3-1. constrictor

(※ 모든 모음의 constrictor -> tongue body)

lip / tongue tip / tongue body

3-2. constrictor location(CL – 앞뒤)

Lip – bilabial / labiodental

Tongue tip – dental / alveolar / retroflex / palato-alveolar

Tongue body – palatal / velar

3-3. constrictor degree(CD – 상하, 공기흐름이 막히는 정도)

Stop / fricative / approximant / vowel

Acoustic

Duration(Sec) – 소리의 시간

Pitch(Hz) – 소리의 높낮이

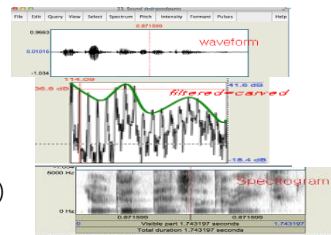
Intensity(dB) – 소리의 크기(볼륨)

Formant(Hz) – 가로줄 4줄(점찍힌 부분) -> 모음을 구별하는 지표

※<wave form> - x축 : 시간 / y축 : value(숫자값)

<spectrum> - x축 : frequency / y축 : amplitude

<spectrogram> - x축 : 시간 / y축 : frequency(숫자값)



(spectrogram -> spectrum을 왼쪽으로 90도 돌려 앞을 보게한 것)

Vowel Acoustics

Simplex tone (=pure tone) -> waveform : sine wave / spectrum : 한 개의 선

Complex tone -> spectrum : 여러 개의 등차간격 선

※Complex tone 의 spectrum 그래프에서 가장 첫 frequency(=fundamental frequency / F0)의 x값(Hz)

-> 나의 pitch

여자 – 남자보다 pitch 가 높음

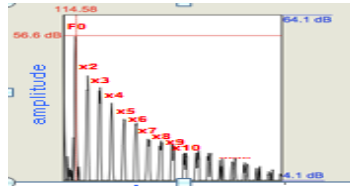
∴ spectrum 그래프가 더욱 등성등성 있을 것임

<Human voice Source> vs <Filtered by Vocal tract>

Human voice source

Spectrum – amplitude(y값) gradually decreasing

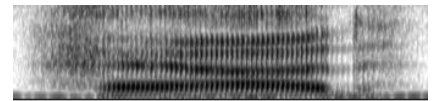
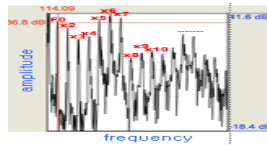
Harmonyx(배음구조) – (x값의 간격이 일정)



Filtered by vocal tract

Spectrum – gradually decreasing (x)

Harmonyx – 배음구조 그대로 (x값의 간격 일정)



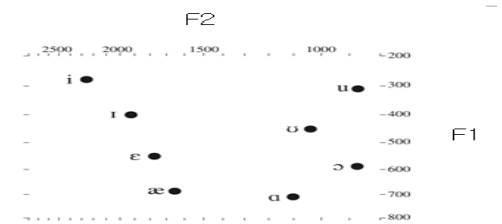
Formant – (spectrum 에서의 산맥 (첫 산맥 -> F1, 두 번째 산맥 -> F2))

(spectrogram 에서의 진한 가로선 (밑에서 첫 가로선 -> F1, 두 번째 가로선 -> F2))

➔ F1, F2를 통해 무슨 모음인지 결정 가능!!

➔ F1 : x축 / F2 : y축 으로 봤을 때 : 조음 위치와 비슷

(∴ F1 -> 모음의 높낮이 결정 / F2 -> 모음의 앞뒤 결정)



Coding -> 자동화 / Coding의 언어 – 다양 (C언어, 파이썬, java ...)

모든 language -> '단어' / '단어의 조합'으로 이루어짐

단어 -> 정보(의미)를 담는 그릇

Computer language에서 단어에 해당하는 것 -> 변수(variable)

Computer language의 문법

➔ 변수에 정보를 assign 하는 것 (variable assignment)

➔ (a 라면 b 하라) – 'if' 문법 (if conditioning)

➔ 여러 번 반복하는 것 – 'for' 문법(for loop)

➔ '함수' (어떠한 입력을 했을 때 특정한 출력 값이 나오는 것) -> 위의 3가지 개념을 packaging 하는 것 (함수 -> 위의 3가지 개념 / 또 다른 함수가 들어갈 수 있음)

Jupyter notebook -> 웹 브라우저 상에서 coding 을 할 수 있도록 해주는 것

컴퓨터에서의 variable -> 숫자, 문자

Computer language 에서의 '=' -> 오른쪽 정보를 왼쪽의 variable 로 assign 한다는 의미

(<ex> a=1 ➔ 1 이라는 정보를 a라는 변수에 assign 한다)

➔ Print(a)를 했을 때 '1'의 값이 나옴 ("함수(변수)" -> ("정보")

Print -> '함수' (함수 -> 누군가가 만든 것을 사용 or 내가 직접 만들어 사용)

(anaconda -> 이런 유용한 함수들을 packaging 해놓은 것)

'셀'을 선택하면 파란색으로 바뀜 -> 이때 a-위에 셀 형성 / b-밑에 셀 형성 / x- 셀 삭제

a=2 라고 한 다음 print(a)를 하면 print(a) -> 2 나옴 (print() -> 최신 정보 반영)

문자 그 자체를 입력하고 싶을 때(문자를 정보로 사용하려면) -> '사용 (ex -> 'love')

'run'의 단축키 -> shift + enter

Love를 변수로 사용 (love=2 / b=love / print(b) -> 2)

마지막 줄에 변수만 입력해도 그에 해당하는 값이 출력됨 (16번 참고)

(;) -> 줄을 바꾸지 않고 줄을 바꾸는 기능(19번 참고)

'Type'함수의 활용

➔ 'list' ([]) (20번)

- ➔ 'int' (정수) (2번)
- ➔ 'float' (소수) (27번)
- ➔ 'str' (문자) (29번)
- ➔ 'tuple' (list 안에 또다른 list가 있을 때) (34번) -> tuple과 list는 큰 차이 없음(tuple이 보안성이 더 높다)
- ➔ 'dict' ('a':~, 'b':~) (36번)