

영어음성학 0905

Praat 에서 위에부분은 웨이브고 밑에거는 spectrum 이다. 아래부분의 아래거는 고주파 위에거가 저주파, 진하기는 그 정도. Pitch와 intensity

영어음성학 0917

English consonants

철자와 소리는 다르다. Ex) gap 에서 G 와 그- 라는 소리는 다름.

[J(y)]는 yearn, yaught 등을 발음할 때 나는 소리. → Year, ear 발음 달라야함

G를 발음할 때 올리는 진동을 뻗 소리가 tlong s(tʃ)임

목이 떴다 - 유성음(voiced sound) → 모음은 모두 유성, 자음은 유성 무성둘다

목을 안떴다. - 무성음(voiceless sound)

Nasal(비음) - m, n, ŋ

English vowels

Monophthongs(단모음), Diphthongs(이중모음)

여기까지는 기본적인 phonology(음운론) - 어떤 소리들이 있고 어떻게 구성되는가?에 대해서. 머릿속에서의 인지적인 과정('가'를 10번 했을 때 같다고 생각되는 것), 더 상위적이고 abstract적

Phonetics(음성학) - 깊이 들어가고, physical 함, 늘 차이가 있음

아 → 이로 소리 바뀌는 이유는 입모양 때문이지 성대 때문이 아님. 그리고 턱의 높낮이가 고정되어있는 상태에서도 소리가 바뀌기 때문에 턱의 높낮이도 소리를 바꾸는 주된 이유가 아님.

Articulatory (from mouth) - 사람이 만드는 소리

Acoustic(through air) - 공기의 흐름. 공기와 소리의 물리적인 부분

Auditory(to ear) - 소리를 듣는 것(공기가 귀로 들어감)

Articulation - THE vocal tract

이(ear), 비(nose), 인(강)(=pharynx)후(두)(=larynx)

후두는 larynx, 목젖부터 후두까지의 관을 인강이라 부름 = pharynx

Lips, teeth, alveolar ridge, hard palate, soft palate(=velum), uvula, pharynx

Alveolar은 아주 중요 외워야함 - 영어의 많은 소리들이 alveolar에서 소리가 남

soft palate = velum인 것 알아야함.

Upper structure은 고정되어있고 lower structure이 움직이면서 소리가 난다.

Lower structure은 tongue으로 이루어져있음

Epiglottis는 기도로 가는 길을 막아줌. 그래서 음식을 먹을 수 있는 것임.

5 speech organs = constrictors= articulators

- ➔ Lips, tongue tip tongue body = articulatory process
- ➔ soft velum = oro-nasal process(nasal(m,n,ng과 oral을 구분해주는 곳)
- ➔ larynx(=voice box) = phonation process

Oral cavity가 닫혀있으면 nasal cavity 는 열려있음.

Velum이 올라가면 막히고 내려가면 열림.

● 시험 문제

- velum이 raised 가 되면 nasal tract 는 열렸을까 막혔을까?
막혔음 ➔ 모든 모음은 그렇게 되고, 비음을 빼 모든 자음들이 모두 velum raised 된 상태임
- 코로 숨을 쉴 때 velum 어떤 상태?
velum lowered 상태(즉 nasal tract가 열려있는상태), 코로 공기가 나가야하니까

Larynx라는 곳에서 완전히 열려서 공기가 나오면 무성음이 되고 닫히면서 기압이 올라오면서 진동을 하면 유성음이 된다. 모든 영어의 소리는 유성음과 무성음으로 나뉜다. (일부만 무성 자음)

0919 영어음성학 → 시험은 오픈북 영어의 모음과 자음 다 구별할 줄 알아야함

Articulatory process in lips / tongue tip / tongue body (주요 조음)

Constrictor 은 lips, tongue tip, tongue body이고, 이들을 통해 미세하게 조정(control)이 가능하다.

Constriction Location – 앞뒤라고 생각, **Constriction Degree** : 상하라고 생각하면된다.

Palatal과 velar → Constriction 관점에서 tongue body 를쓰는 것은 동일하지만 constriction location(CL) 관점에서는 조금 다름

th → constriction 관점에서 tongue tip 사용, CL 관점에서는 윗니를 hit 한다.

d,t,n → tongue tip을 사용하여 CL관점에서 alveolar 소리를 낸

⇒ CL 의관점에서 lips는 두개 tongue tip 은네 개 tongue body 는 두개의 발음이 있다

Lip – bilabial, labio-dental, tongue tip – dental, alveolar, palato-alveolar, retroflex,
tongue body – palatal, velar

CD 얼마나 upper part를 hit하느냐에 따라 CD가 결정된다.(stop, fricatives, approximants, vowels)

Ex) D – constriction tongue tip 사용, CL 관점에서 alveolar, CD 관점에서 stop.(upper part완전 불음)

CD 관점에서 모음은 자음보다 degree 가 더 작음. Lower part 에 더 가까움

자음은 stops (폐쇄음) fricatives(마찰음) approximants(접근음?? j/y 해당) 의 종류임

Stops - p,t,k,b,d,g,

- M,N,ŋ→ 어디에 해당될까? Oral tract 에서의 문제이므로 코가 열린것과 상관이 없어서 다 stops 임

Fricatives – s, z, f, v, theta, eth, 시(long s), 3(yogh),

Approximants – r, l, w, j(y)

- 시험문제(by specifying constrictors, CD, CL)

Velum 이 raised 되었음. Glottis(between larynx) is opened and constriction tongue tip, CL alveolar, CD stop → ‘T’

모든 모음은 constrictor 로서 tongue body 만 쓴다. 모음과 같은 constrictor을 쓰는 자음은 k 이고, K 의 velum is lowered → ng가 된다.

→ 모든 자음 모음은 이 관점에서 설명될 수 있어야함

Phoneme – individual sound that form words

개별된 소리들은 orthography들과 다름. / /안에 있는 것들은 all phonemes

.

Acoustics - Acoustic in Praat

빨간띠가 formant인데 제일 밑에거부터 첫번째임. 줄여서 f1,f2... →모음이 무엇인지를 결정함
모든 사람의 아- 를 녹을 했을 때 formant 가 보이는데 첫번째 두번째 FORMANT가 비슷하게 나타남

Formant 값이 무엇인지에 따라 무슨 모음인지 알 수 있음.

Pitch에서 남녀 목소리 구분하려면 pitch setting 에서 pitch range를 정해주어야함.

VOWEL ACOUSTICS → 어떻게 소리가 만들어지는가? 가 한학기동안 배울 내용

제일 큰 반복이 larynx의 반복과 똑같음. 1초를 wave의 duration으로 나누면 1초동안 몇 번 떨리는지 알 수 있음 Hz로 나타남.

→(praat 캡처해서 다시 정리)

0924 영어음성학

● 시험문제

1. Constrictor – lips, tongue tip, tongue body
2. Larynx
3. velum

Ex) /p/라는 소리를 specify(CL,CD) 하면 lips(constriction), bilabial(CL관점에서), stop(CD관점에서), velum → raised, larynx → opened →이렇게 specify 할 줄 알아야함

/b/ 로 바꾸고싶으면 → larynx를 close로 바꾸면된다.(즉 voiceless → voiced)

/d/로 tongue tip, CL이 alveolar, CD는 stop, velum → raised, larynx → closed

/z/ tongue tip, alveolar, fricative, velum → raised, larynx → closed

/n/ tongue tip, alveolar, stop, velum → lowered, larynx → closed

/n/에서 larynx → opened 되면 숨쉬는 상태임

Praat도 시험에 나옴.

Vowel acoustics

파고가 크고 작음에 따라 wave의 생김새 결정된다.

/a/ 녹음하고 repeating event가 일어나는데 vocal folds의 vibration 때문임

Sine wave를 만들었을 때 똑같은 음높이로 소리난다. 이렇게 음높이는 같지만 Sound quality는 다름(즉 /a/소리가 나지는 않음)

Source

성대에서 직접 녹음하는 소리는 음의 높낮이 정도만 다르고 아/a/나 이/i/는 우리의 입모양에 의해서 바뀐다.

Complex tone in spectrum

그림은 sine wave임 맨위가 가장 기본적인 형태 sine wave를 기본짓는 것은 frequency와 magnitude 에 의해서 결정된다.

모든 sound를 포함한 signal은 여러 다르게 생긴 sine wave의 결합으로 형성된다.(=모든 신호들은

sine wave의 합으로 표현할 수 있음) → 단순한 것들로 복잡한 것을 표현할 수 있다.

ppt 그래프에서 첫번째 그래프는 magnitude는 크지만 frequency(빠르기 즉 저음)은 작다

세번째거는 첫번째거에 비해서 세배 빠르다. Magnitude는 첫번째가 제일크고 세번째가 제일 작다.

Sine wave를 각각 표현할 수 있음. 세개를 합쳐보면, 마지막 그래프의 형태가 나타나고 sine wave가 아닌 복잡한 소리로 만들수 있음 . 반대로 복잡한 신호들은 sine wave들의 합으로 표현될 수 있다.

4번 sine wave에서 반복되는 주기는 1번 그래프와 같다. 1초에 100번 반복 = 100HZ

simplex tone = sine wave

Sine wave를 더 단순하게 표현한 것은 오른쪽 그래프

- 시험문제

- 왼쪽그래프에서 X축은 시간이고 Y축은 단순한 숫자값 value라고 생각하면된다. 볼티지 값이라고 생각하면됨. 오른쪽 그래프는 X축이 frequency, Y축이 amplitude
- Time value그래프를 frequency value 그래프로 변환시킬 줄 알아야함

오른쪽 그래프를 뭐라고 하나면 spectrum이라고 함. spectrum은 일반적인 이퀄라이즈의 형태임

Specctrum이 일반적으로 어떻게 형태인가를 보면 simple tone(wave)인 단순한 소리들의 합으로 4번 그래프임. 그것을 만들어내는 것을 synthesize(합성)이라 하고, 복잡한 소리를 어떤것들의 합으로 이루어져있는지 보는 것을 analysis(분석)이라고 한다

Pure tone=simplex tone

praat에서 아랫 부분은 spectro gram 은 spectrum을 시간축으로 구성한 것

view spectrum slice에서 보이는 것들이 오른쪽 그래프랑 똑같은 것임

아- 라는 소리를 view spectrum slice에서 보면 소리들이 반복되는데 **등간격으로** 있음. 첫번째가 100이라면 200 300 400 이렇게 갈 것임. 첫번째가 100이었다면 나의 pitch도 100이었다는 것임 즉 제일 작은 simplex tone 하고 일치함.

단순한 sine wave이 이루어져 '아'라는 소리를 이루는데

모음을 어떻게 만드는데가 중요 pitch 에 해당하는 frequency 알게 된다면 배수로 넘어가면서 소리를 만들수 있는 것임.

frequency(진동수)가 어떻게 되는건지는 1초에 몇번 성대가떨리는건가와 같은 것임

요약하자면 '아' 라고 이야기하면 simplex tone 들로 이루어져있고 제일 느린 frequency 우리말의

pitch (음높이)와 동일하다. 그리고 그것은 우리 성대에서 1초에 몇번 떨리는가와도 일치한다.

Practice with pure tone and spectrum

Human voice source 에서는 성대가 떨려 나는 소리를 말하고, 성대가 떨려나는 소리를 캡처한 것은 source 라고함

larynx에서 나는 소리를 캡처했을때를 source라고 하고 filter에서 어떻게 바뀌느냐에 따라서 소리가 달라짐. nasal과 oral 이 된다.

첫번째 주파수가 우리의 pitch와 일치함. 제일 첫번째 나온 것을 F_0 =fundamental frequency
amplitude가 Gradually decreasing하는 것이 일반적으로 나타나는 형태임
 F_0 에서 계속 곱해져 가는 것을 harmonics라고 함

- 시험문제

여성의 아—는 첫 시작이 남자보다 높고 더 듬성듬성한 형태의 그래프가 나타날 것임
따라서 10000HZ로 잘랐을 때 남자의 배음 숫자가 더 많음

배음의 구조는 깨지지 않음(배수로 가는것) BUT amplitude구조는 깨질 수 있음. Smoothly decreasing 되던 것이 깨진다는 것임

Spectro gram을 읽는 방법 = spectro gram 도 x축은 시간인데 y축은 frequency 임.

까맣게 생긴 것이 크기가 센 것이라고 보면된다. Low frequency 쪽으로 갈수록 에너지가 큼. High 로 갈수록 에너지가 약해짐

➔ Filter를 거지면 **filtered by vocal tract**에서 처럼 amplitude가 깨지는 형태의 그래프가 나타남

926 영어음성학

Sine wave를 합하는 것을 synthesize, 합성된 complex wave를 분석하는 것이 analysis

- 시험문제

Spectro gram과 wave form은 x축은 시간이고 y축은 frequency이다.

막대 그래프인 spectrum으로 나타낸다면 x축 frequency, y축이 amplitude

Human voice source

Sine wave의 F_0 가 우리의 목소리 높이 (pitch)와 일치한다. 단위는 Hz ,등간격으로 frequency는 점점 높아지는 그래프들이 있음

Filtered by vocal tract

입을 가지고 filter 역할을 한다. Vocal tract를 가지고 filter가 씌인 상태에서는 무슨 소리인지 식별 가능해진다.

Source and filter

저주파에서만 강하고 고주파로 갈수록 에너지는 약해짐

ppt에서 위에부분이 source 밑에 부분이 filter 임

Source

harmonics들은 sine wave의 배음으로 이루어져 있다. F0에 곱하기 1,2,3, 이런것들로 이루어져 있다는 것

pure tone들이 점점 high frequency로 갈수록 amplitude는 decrease 된다.

Filter

Vocal tract는 우리의 입모양, tract는 tube를 뜻하는데 여기서 filtered 되어서 다른소리로 바뀌는 것임.

Peaks/mountain부분이 있고, 그렇지 않은 valleys부분들이 있는데 어디에 산맥이 나타나느냐는 다르다. 아와 이의 산맥의 패턴은 다르겠지만 아 할때의 산맥과 할때의 산맥은 누구든지 똑같이 나타남.

첫 번째 산맥에 해당되는 주파수를 첫번째 formant라고 함. 그래서 산맥들을 formants 라고 한다..

F0는 제일 첫번째 harmonics이고, source에서 산맥이 만들어질 때 첫번째 산맥이 F1 formant 임

Guitar plucking

Guitar 소리는 complex tone임 하지만 perceive하는 음높이는 똑같음

Synthesizing source

praat에서 10개의 소리를 combine stereo로 열었을 때 10개의 소리가 동시에 존재하는 것임 (stereo로 존재하는상태)

스테레오를 convert to mono로 했을 때 complex wave가 나타난다. 반복주기는 우리가 만들었던 첫번째 sine wave와 일치한다.

음높이는 인지학적으로 100Hz 와 똑같다고 인식하게 됨
이를 F0라 부르고 fundamental frequency라고 부르고 pitch라고 부름

무한대로 갈수록 peak 하나 0 계속 peak 하나 0 계속 인 형태로 나올 것임
이를 우리는 pulse? Perse? train 이라고 부른다.

Synthesizing source

Source-filter theory

Output spectrum 에서 첫번째 산맥에서 튀어나온 부분을 F1이라고하고 그뒤에 F2 F3 ...
Source spectrum에서 처음 시작하는 부분이 F0 임

Formants

그림의 F1과 F2는 서로 명백하게 다른 것임. 그래서 F1과 F2만 있으면 된다

F1 과 F2를 그래프로 나타내보면 F2를 x축 F1을 y축으로 해서 점을 찍어보면 F1은 모음의 높낮이를 결정하고, F2는 front/back vowel 을 결정한다.

10월 1일

Praat은 만들어놓은 프로그램이고

코딩은 그걸 만드는 것임. Praat의 일부를 만들어보는 것이 이번학기 수업

코딩= 자동화라고 생각하면된다.

모든language는 단어가 있음. 어떻게 combine을 하느냐에 따라 communication 이 된다.

단어의 역할은 정보를 담는 그릇이고 단어는 변수(variable)이다.

1. 변수라는 그릇에다가 정보를 넣는 것. Variable assigning
2. If 문법의 사용 if conditioning

3. 자동화의 가장중요한 것은 여러 번 반복하는 것인데 for문법을 사용
4. 함수. 입력과 출력으로 packaging 을 하는 것

오른쪽에 있는 정보를 왼쪽에 있는 정보로 assign한다. 1이라는 정보를 a라는 variable에 넣는다.

순서가 바뀌면안된다

Print는 함수역할 모든 함수는 누가 만들어 놔야한다.

Print는 어떤 함수인지 알고 있어야함. 입력은 괄호안에 넣으면 된다.

문자정보는 " 해야한다.

변수에다가 정보를 하나만 넣었음

여러 개를 한꺼번에 넣을때는? []에 넣어 표현하면되고 이를 list라 부름 a에는 list로서 1,2,3,5가 들어가있는 것임

Data type - list, int, float, str, tuple, dict(중괄호써야함 표제어와 설명의 쌍으로 구성되어있음, column, comma로 이어줌)

10월10일

하나의 셀을 markdown 으로 하면 실행이 안된다. (주석처럼 쓰는 것 가능)

In [2]:

```
#오늘은 syntax 파일 ==> for 문법과 if conditioning
#function 이 뭔지 print,len,type 의 function 배웠음
#comment 다는 방법은 # ~~라 쓰면 된다. (가장 원시적인 방법)
```

In [3]:

```
#여러개를 여러 번 해야할 때 for 을 쓴다.
```

In [4]:

```
#for i in ~ : ==> in 뒤에있는 것을 하나씩 돌려서 i 로 받아서 무엇인가를 하라.
```

```
a=[1,2,3,4]
```

```
for i in a:
```

```
    #밑에 아무것도 없으니까 error 뜨는 것임
```

```
File "<ipython-input-4-c60b3401d523>", line 3
```

```
    for i in a:
```

```
        ^
```

```
SyntaxError: unexpected EOF while parsing
```

In [5]:

```
a=[1,2,3,4]
```

```
for i in a:
```

```
    print(i)
```

1
2
3
4

In [6]:

```
#a 속에 있는 것을 하나하나씩 불러서 i 에다가 넣어라 그리고 계속 돌아라.  
#첫번째가 1 이니까 i 가 1 을 받게되고 1 이 print out 이된다.  
#두번째거는 2 가 i 로 들어가고 print out 은 2 로 3,4 도 마찬가지
```

In [9]:

```
a = [1, 2, 3, 4]  
for i in range(4):  
    print(a[i])  
#in 뒤에 range 함수 쓸 수 있음 range 에 숫자가 나오면 list 를 만들어줌  
#4 가 나온건 4 개의 index 를 만들어주어라임  
#(index 는 0 부터 시작 즉 0 부터 3 까지의 list 만듦) 그래서 range 가  
0,1,2,3 을 만들어줌  
#i 는 0 부터 받음. a 의 0 번 째 거를 print out 하라부터 시작해서 3 번째까  
지
```

1
2
3
4

In [10]:

```
a = [1, 2, 3, 4]  
for i in range(4):  
    print(i)  
  
#i 만 적으면 0,1,2,3 을 도출하는데 왜냐면 range(4) 는 0 부터 4 개의 숫자를  
받음
```

0
1
2
3

In [14]:

```
a = [1, 2, 3, 4]  
for i in range(len(a)):  
    print(a[i])  
  
#len(a) 가 4 기 때문에 0,1,2,3 으로 돌고 print(a[i]) 에 들어감
```

1
2
3
4

In [18]:

```
a = ['red', 'green', 'blue', 'purple'] #str 의 list
print(a[0])
print(a[1])
print(a[2])
print(a[3])

red
green
blue
purple
```

In [19]:

```
a = ['red', 'green', 'blue', 'purple']
for s in a:
    print(s)
#for 루프 안에 indent?(앞에 공간)가 있어야함
#a 에 있는 list 를 갯수만큼 for 루프를 돌려라
#for 루프가 4 번 돌면서 s(variable)가 매번 바뀌면서 print out 을 해주는
것임

red
green
blue
purple
```

In [20]:

```
a = ['red', 'green', 'blue', 'purple']
for s in range(len(a)):
    print(a[s])

red
green
blue
purple
```

In [21]:

```
a = ['red', 'green', 'blue', 'purple']
for s in range(4):
    print(a[s])

red
green
blue
purple
```

In [23]:

```
a = ['red', 'green', 'blue', 'purple']
for s in range(len(a)):
    print(s)
#range(len(a))의 index 값이 0,1,2,3 이기 때문에 print(s)만 하면 아래
값 도출
#range(4), range(len(a)) 둘다 똑같은 결과 나오는데 len 쓰는게 더 유용
```

0
1
2
3

In [24]:

```
a = ['red', 'green', 'blue', 'purple']
b=[0.2,0.3,0.1,0.4]
for i, s in enumerate(a):
    print(a[i])
#enumerate: 번호를 매기다.
#i 에는 0 이 들어가고 s 에는 red 가 들어가게 됨
#output 값이 자기자신도 되지만 번호도 매겨준다. (앞에있는게 번호, 뒤가 자
기자신)
#i 에는 번호가 들어온 거니까 a 의 몇번째가 나오게 되는 것임
#s 는 사실상 안쓴것임.

red
green
blue
purple
```

In [26]:

```
a = ['red', 'green', 'blue', 'purple']
b=[0.2,0.3,0.1,0.4]
for i, s in enumerate(a):
    print("{}:{}".format(s,b[i]*100))
#""형태로 적고 싶을때 중괄호로 해주면 된다.
#i,s 는 0 과 red 를 묻고 밑으로 들어간다. 중괄호와 format() 괄호안의 갯수
는 같음

red:20.0%
green:30.0%
blue:10.0%
purple:40.0%
```

In [28]:

```
a = ['red', 'green', 'blue', 'purple']
b=[0.2,0.3,0.1,0.4]
for s,i in zip(a,b):
    print("{}:{}".format(s,i*100))

#a,b 는 반드시 갯수가 같은 list 여야함 zip 은 독립적인 list 를 합치는 것
임.

red:20.0%
green:30.0%
blue:10.0%
purple:40.0%
```

In [29]:

```
a=0
```

```
if a==0:
    print("yay!")
```

*#두개의 '=='은 우리가 아는 진짜의 equal 의미
#if 에서는 == ,<,> 많이 쓰임
#if 하면 일해라를 i 그 해당되는 부분을 반드시 indent 로 시작해야한다.*

yay!

In [34]:

```
a=0
if a==0:
    print("yay!")
    print("let's go")
```

yay!

let's go

In [35]:

```
a=0
if a>=0:
    print("yay!")
    print("let's go")
#부등호가 먼저나오고 = 나와야함
```

yay!

let's go

In [37]:

```
a=0
if a!=0:
    print("yay!")
    print("let's go")
else:
    print("no")
```

no

In [38]:

```
a=1
if a!=0:
    print("yay!")
    print("let's go")
else:
    print("no")
```

yay!

let's go

In [39]:

#100 번의 루프가 도는데 50 번의 루프가 또 돈다. --> 총 5000 번이 돈다.

In [41]:

```
for i in range(1, 3):
    for j in range(3, 5):
```

```

    print(i*j)
#무조건 시험문제, 두번 for loop 가 되는것임
#range 함수에 숫자를 두개를 넣으면 첫번째 숫자부터 두번째 숫자 직전까지 감
#즉 range(1,3)에서는 1,2 가 나옴
#제일 큰 forloop는 1,2 두 번 돈다.
#이 두개가 두번째 3,4 에서 각각 돈다. 그래서 총 4 개의 숫자가 print out
되는것임

```

```

3
4
6
8

```

In [45]:

```

for i in range(1,3):
    print(i)
    for j in range(3,5):
        print(i*j)

```

```

#print(i) 위에 for 만 실행 그래서 print(i)는 두번 실행되고
#print(i*j)는 아래 for 은 두번이지만 위에 for 에서도 두번이 실행되니까
내려오면서
#4 번이 실행된다
#그래서 print out 되는 갯수는 총 6 개 즉 독립적으로 실행된다

```

```

1
3
4
2
6
8

```

In [46]:

```

for i in range(1,3):
    for j in range(3,5):
        if j>=4:
            print(i*j)
#j 가 3 일 때는 실행이 안된다.

```

```

4
8

```

In [48]:

```

for i in range(1,3):
    for j in range(3,5):
        if i>=3:
            print(i*j)
#여기서 아무것도 print out 안되는 이유는 i 가 3 보다 큰 index 가 없으니까

```

In [50]:

```

for i in range(1,3):

```

```
if i>=3:
    for j in range(3,5):
        print(i*j)
```

#if 다음에 indent 가 없기때문에 error 가 뜨는 것. indent 가 있고 없고의 차이가 있음

#48 번째 cell 은 indent 있으니 error 가 뜨지는 않음

```
File "<ipython-input-50-c2973b39e11f>", line 3
    for j in range(3,5):
    ^
```

IndentationError: expected an indented block

In []: