Carnegie Mellon University
Language
Technologies
Institute

# 11-324/11-624/11-724 Human Language for AI

## Phonological Features and Optimality Theory

David R. Mortensen

September 29, 2021

Language Technologies Institute
Carnegie Mellon University

# Learning Objectives

At the end of this lectures, students will know:

- The motivations for phonological feature theory
- How segments are defined in terms of features
- Where to find complete feature vectors (PanPhon, PHOIBLE)
- The advantages and challenges of constraint-based phonology
- What Optimality Theory and Correspondence Theory are and how they work
- What the relationship between Markedness and Faithfulness is

Students will know how to do the following things:

- Identify a natural class using phonological features
- Evaluate an OT tableau

# Feature Theory

# Vowel Place Features

| | i | y | ɨ | u | e | ø | ʌ | o | æ | œ | ɑ | ɒ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| high | + | + | + | + | − | − | − | − | − | − | − | − |
| low | − | − | − | − | − | − | − | − | + | + | + | + |
| back | − | − | + | + | − | − | + | + | − | − | + | + |
| round | − | + | − | + | − | + | − | + | − | + | − | + |

# Some Consonant Place Features

| | p | t̪ | t | t͡ʃ | ʈ | c | k | q | ʕ | ʔ |
|---|---|---|---|---|---|---|---|---|---|---|
| anterior | + | + | + | − | − | − | − | − | − | − |
| coronal | − | + | + | + | + | − | − | − | − | − |
| distributed | | + | − | + | − | | | | | |
| high | − | − | − | − | − | + | + | − | − | − |
| back | − | − | − | − | − | − | + | + | + | − |
| low | − | − | − | − | − | − | − | − | + | − |

# Some Manner Features

| | u | w | l | m | β | b |
|---|---|---|---|---|---|---|
| syllabic | + | − | − | − | − | − |
| consonantal | − | − | + | + | + | + |
| sonorant | + | + | + | + | − | − |
| continuant | + | + | + | − | + | − |
| nasal | − | − | − | + | − | − |

# Catalan Example I

| MASC SG | FEM SG | | MASC SG | FEM SG | |
|---|---|---|---|---|---|
| əkelʲ | əkelʲə | ‘that’ | mal | malə | ‘bad’ |
| siβil | siβilə | ‘civil’ | əskerp | əskerpə | ‘shy’ |
| ʃop | ʃopə | ‘drenched’ | sɛk | sɛkə | ‘dry’ |
| əspɛs | əspɛsə | ‘thick’ | gros | grosə | ‘large’ |
| baʃ | baʃə | ‘short’ | koʃ | koʃə | ‘lame’ |
| tot | totə | ‘all’ | brut | brutə | ‘dirty’ |
| pɔk | pɔkə | ‘little’ | prəsis | prəsizə | ‘precise’ |
| frənses | frənsezə | ‘French’ | gris | grizə | ‘grey’ |
| kəzat | kəzaðə | ‘married’ | bwit | bwiðə | ‘empty’ |
| rɔt͡ʃ | rɔʒə | ‘red’ | bot͡ʃ | boʒə | ‘crazy’ |
| orp | orβə | ‘blind’ | lʲark | lʲaryə | ‘long’ |
| sek | seɣə | ‘blind’ | fəʃuk | fəʃuɣə | ‘heavy’ |
| grok | groɣə | ‘yellow’ | puruk | puruɣə | ‘fearful’ |
| kandit | kandiðə | ‘candid’ | frɛt | frɛðə | ‘cold’ |

# Catalan Example II

| MASC SG | FEM SG | | MASC SG | FEM SG | |
|---|---|---|---|---|---|
| səɣu | səɣurə | ‘sure’ | du | durə | ‘hard’ |
| səɣəðo | səɣəðorə | ‘reaper’ | kla | klarə | ‘clear’ |
| nu | nuə | ‘nude’ | kru | kruə | ‘raw’ |
| flɔɲd͡ʒu | flɔɲd͡ʒə | ‘soft’ | dropu | dropə | ‘lazy’ |
| əgzaktə | əgzaktə | ‘exact’ | əlβi | əlβinə | ‘albino’ |
| sa | sanə | ‘healthy’ | pla | planə | ‘level’ |
| bo | bonə | ‘good’ | sərɛ | sərɛnə | ‘calm’ |
| suβlim | suβlimə | ‘sublime’ | al | altə | ‘tall’ |
| fɔr | fɔrtə | ‘strong’ | kur | kurtə | ‘short’ |
| sor | sorðə | ‘deaf’ | bɛr | bɛrðə | ‘green’ |
| san | santə | ‘saint’ | kəlɛn | kəlɛntə | ‘hot’ |
| prufun | prufundə | ‘deep’ | fəkun | fəkundə | ‘fertile’ |
| dəsen | dəsentə | ‘decent’ | dulen | dulentə | ‘bad’ |
| əstuðian | əstuðiantə | ‘student’ | blaŋ | blaŋkə | ‘white’ |

# PanPhon

**PanPhon** is an ontology and associated Python library for dealing with phonological feature representations.

- `https://github.com/dmort27/panphon`
- Feature tables at `https://github.com/dmort27/panphon/blob/master/panphon/data/ipa_bases.csv` and `https://github.com/dmort27/panphon/blob/master/panphon/data/ipa_all.csv`
- `pip install panphon`
- Python 2 and 3
- Compute subsumption relations among feature matrices
- Compute edit distance between feature vectors
- Compatible with **Epitran** G2P

# Panphon example

```
>>> import panphon
>>> ft = panphon.FeatureTable()
>>> ft.word_fts(u'swit')
[<Segment [-syl, -son, +cons, +cont, -delrel, -lat, -nas, 0strid, -voi, -sg,
>>> ft.word_fts(u'swit')[0].match({'cor': 1})
True
>>> ft.word_fts(u'swit')[0] >= {'cor': 1}
True
>>> ft.word_fts(u'swit')[1] >= {'cor': 1}
False
>>> ft.word_to_vector_list(u'sauɹ', numeric=False)
[[u'-', u'-', u'+', u'+', u'-', u'-', u'-', u'0', u'-', u'-', u'-', u'+', u'
```

# PHOIBLE

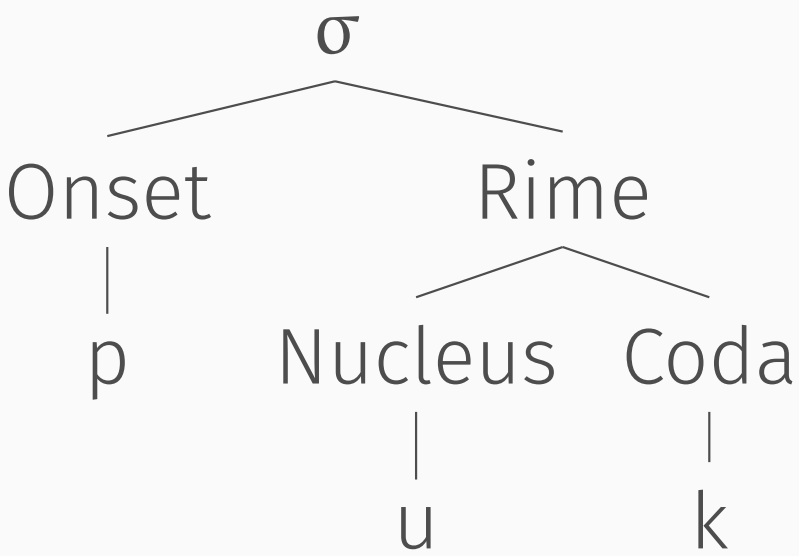
PHOIBLE is a database of segment inventories for a large number (2186) of languages.

- `https://phoible.org/`
- `https://github.com/phoible/dev`

- Phonological feature for each segment in each inventory
- Somewhat different system that PanPhon

# Constraint-based Phonology

# Background: syllable structure



# Optimality Theory Tableau: Maori

| /hopuk/ | NoCoda | Max |
|---|---|---|
| ☞ a. ho.pu | | * |
| b. ho.puk | * | |

# Optimality Theory Tableau: English

| /bik/ | MAX | NOCODA |
|---|---|---|
| a. bi | * | |
| ☞ b. bik | | * |

# More Maori

| /hopuk/ | Ident | Dep | NoCoda | Max |
|---|---|---|---|---|
| ☞ a. ho.pu | | | | * |
| b. ho | | | | *** |
| c. ho.puk | | | * | |
| d. ho.pu.kə | | * | | |
| e. ho.pu.i | * | | | |

# Markedness and Faithfulness

- **Markedness constraints** penalize dispreferred structures in the output (surface form). They tell you which sequences are malformed ("low probability")
  - Onset
  - NoCode
  - *Complex
  - Agree
  - CodaCond
- **Faithfulness constraints** penalization differences between the input and the output (the underlying representation and the surface representation). They tell you which mappings are dispreferred ("low probability")
  - Ident
  - Max
  - Dep
  - Linearity
  - Uniformity
  - Integrity

# Correspondence Theory and the Noisy Channel Model

$$\begin{aligned}\hat{s} &= \arg\max_s p(s|u) \\ &= \arg\max_s \frac{p(u|s)p(s)}{p(u)} \\ &= \arg\max_s p(u|s)p(s)\end{aligned}$$

---

Markedness — Faithfulness

Source Model — Channel Model

Source $\longrightarrow$ Surface $\longrightarrow$ Channel $\rightarrow$ Underlying

decode

# Advertising