dzsungel cikk

Rácz, Lukács

5/18/2022

Hayes et al. (2009, 2017): certain vacillating stems show a front bias when the stem-final consonant is a

- bilabial noncontinuant: [b p m]
- coronal sonorant: [n N l r]
- sibilant: [s z S Z ts tS dZ]
- consonant cluster or geminate

excluded:

- [b] comparative adjectival suffix
- [k] plural marker
- [m], [d] POSS.1/2SG
- [S] adjectivizer
- H0: no difference between the harmonic behavior of the categories
- H1: phonological and lexical categories > baseline category
- H2: phonological and baseline categories > lexical category
- H3: phonological category > baseline and lexical categories

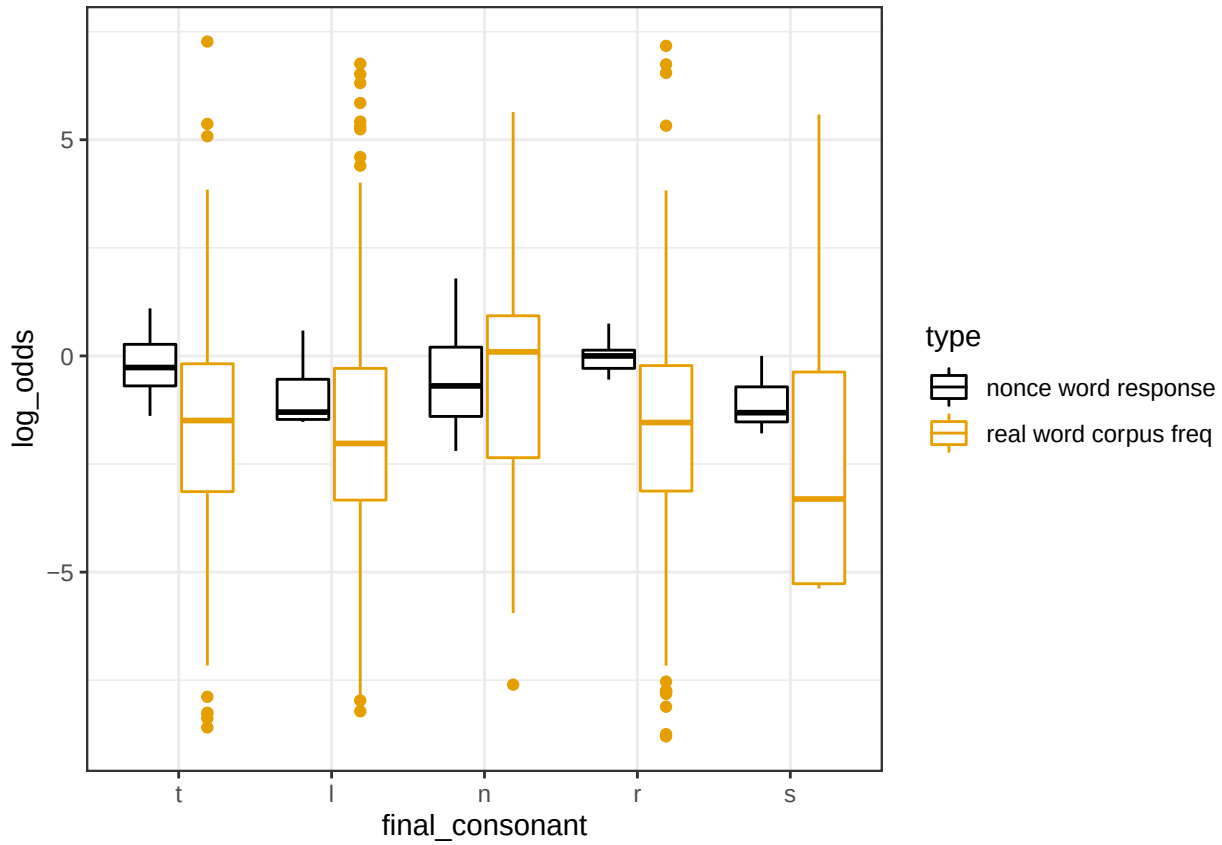
stem category	stem-final consonant	harmonic bias	motivation
baseline	[f], [t]	(front)	(height effect)
phonological	[l], [n]	front	phonological (cor son)
lexical	[r], [s]	front	phonological (cor son / sib)
		back	lexical (familiarity)

Final consonants in the data and the corpus

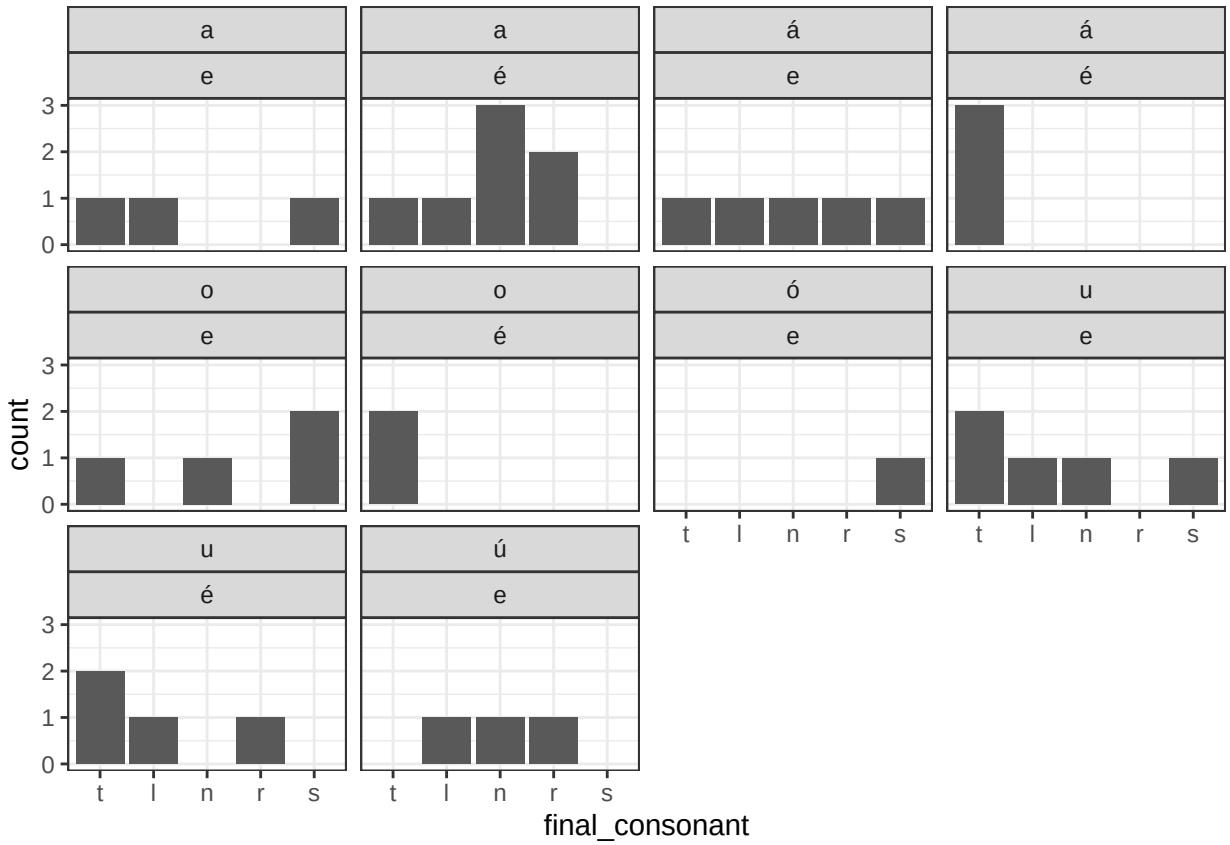
Preferences across final consonants in the data are inconclusive, don't look much like the corpus at all.

Final consonants are not balanced across stem vowels particularly well.

```
d %>%
  filter(
    !is.na(harmony_effect),
    !stem_vowel %in% c('i','í')
    ) %>%
  mutate(final_consonant = fct_relevel(final_consonant, 't','l','n','r','s')) %>%
  ggplot(aes(final_consonant,log_odds,colour = type)) +
  geom_boxplot() +
  theme_bw() +
  scale_colour_colorblind()
```




```
d %>%
  filter(
    !stem_vowel %in% c('i','í'),
    type == 'nonce word response',
    !is.na(harmony_effect)
    ) %>%
  mutate(final_consonant = fct_relevel(final_consonant, 't','l','n','r','s')) %>%
  ggplot(aes(final_consonant)) +
  geom_bar() +
  theme_bw() +
  facet_wrap( ~ stem_vowel + vowel)
```



Raw string data

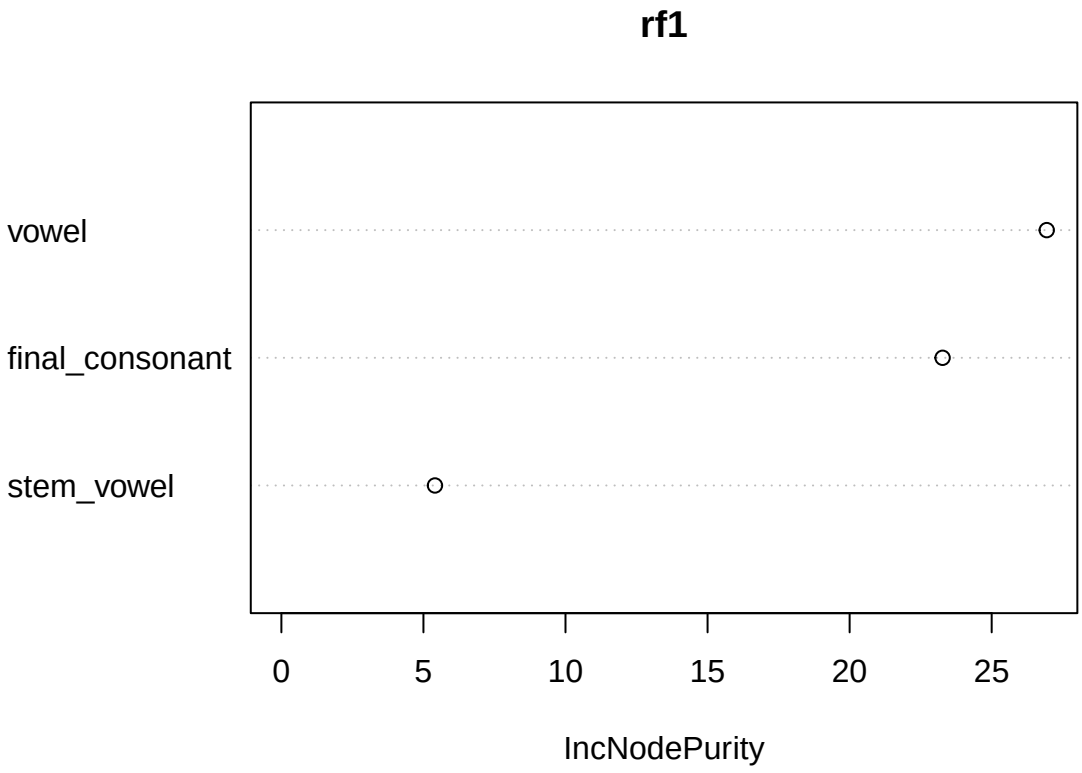
We take exp data and exclude front stem vowels. We rerank final consonant so that [t] (the most frequent final consonant) is baseline level.

We fit a rf and a lm.

```
d2 = d %>%
  filter(type == 'nonce word response', !stem_vowel %in% c('i','í')) %>%
  mutate(final_consonant = fct_relevel(final_consonant, 't'))

rf1 = randomForest(log_odds ~ stem_vowel + vowel + final_consonant, data = d2)

lm1 = lm(log_odds ~ stem_vowel + vowel + final_consonant, data = d2)

print(formula(lm1))
```

```
## log_odds ~ stem_vowel + vowel + final_consonant
```

```
varImpPlot(rf1)
```




```
tidy(lm1, conf.int = T) %>%
  select(-p.value) %>%
  kable(digits = 2)
```

term	estimate	std.error	statistic	conf.low	conf.high
(Intercept)	-0.85	0.21	-4.04	-1.27	-0.43
stem_vowelá	0.11	0.16	0.67	-0.21	0.43
stem_vowelo	-0.16	0.16	-1.02	-0.48	0.16
stem_voweló	-0.27	0.30	-0.92	-0.86	0.32
stem_vowelu	-0.16	0.15	-1.01	-0.46	0.15
stem_vowelú	-0.44	0.36	-1.21	-1.16	0.28
vowelé	1.21	0.15	7.81	0.90	1.52
final_consonant©	-0.67	0.33	-2.06	-1.32	-0.02
final_consonant¬	0.33	0.25	1.35	-0.16	0.82
final_consonant¥	-0.20	0.26	-0.75	-0.72	0.32
final_consonantc	0.14	0.29	0.49	-0.44	0.73
final_consonantd	0.14	0.27	0.52	-0.40	0.68
final_consonantg	-0.53	0.29	-1.81	-1.12	0.05
final_consonantj	-0.19	0.30	-0.64	-0.78	0.40
final_consonantk	-0.30	0.29	-1.04	-0.88	0.27
final_consonantl	-0.34	0.28	-1.22	-0.89	0.21
final_consonantm	-0.24	0.33	-0.73	-0.88	0.41
final_consonantn	-0.08	0.26	-0.29	-0.60	0.45
final_consonantp	-0.66	0.28	-2.36	-1.21	-0.10

term	estimate	std.error	statistic	conf.low	conf.high
final_consonantr	0.23	0.30	0.77	-0.36	0.82
final_consonants	-0.13	0.29	-0.46	-0.71	0.44
final_consonantß	-0.06	0.29	-0.20	-0.64	0.52
final_consonantv	-0.34	0.42	-0.82	-1.18	0.49
final_consonantz	0.13	0.26	0.49	-0.39	0.65

Phonetic features

We code variables.

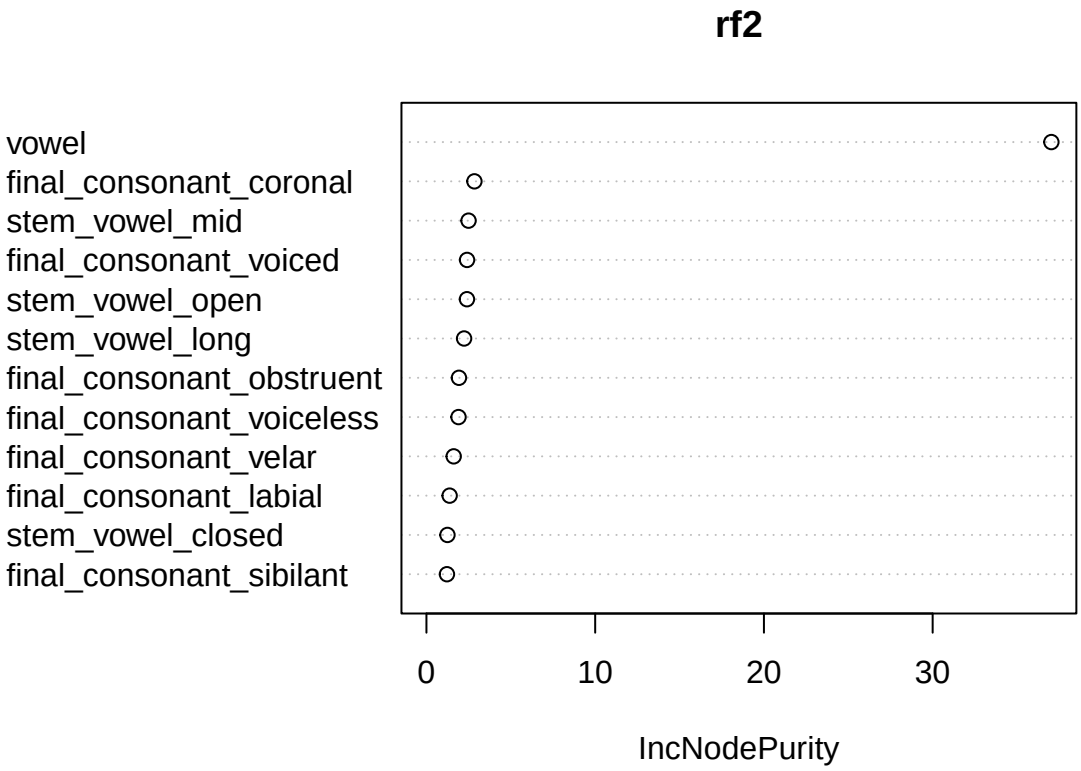
```
# 'cs' = 'ç', 'sz' = 'ß', 'ty' = '†', 'gy' = '©', 'ny' = '¥', 'ly' = '¬'
d2 %<>%
  mutate(
    stem_vowel_long = str_detect(stem_vowel, '[áóú]'),
    stem_vowel_open = str_detect(stem_vowel, '[aá]'),
    stem_vowel_mid = str_detect(stem_vowel, '[oó]'),
    stem_vowel_closed = str_detect(stem_vowel, '[uú]'),
    final_consonant_labial = str_detect(final_consonant, '[pmv]'),
    final_consonant_coronal = str_detect(final_consonant, '[dcsztnrlj©¥¬]'),
    final_consonant_velar = str_detect(final_consonant, '[kg]'),
    final_consonant_voiceless = str_detect(final_consonant, '[pcßt]'),
    final_consonant_voiced = str_detect(final_consonant, '[dzgv]'),
    final_consonant_obstruent = str_detect(final_consonant, '[tdpbcßszç©kg]'),
    final_consonant_sibilant = str_detect(final_consonant, '[sßz]')
  )
```

```
rf2 = randomForest(log_odds ~
                       vowel +
                       stem_vowel_long +
                       stem_vowel_open +
                       stem_vowel_mid +
                       stem_vowel_closed +
                       final_consonant_labial +
                       final_consonant_coronal +
                       final_consonant_velar +
                       final_consonant_voiceless +
                       final_consonant_voiced +
                       final_consonant_obstruent +
                       final_consonant_sibilant,
                       data = d2
)

lm2 = lm(log_odds ~
                       vowel +
                       stem_vowel_long +
                       stem_vowel_open +
                       stem_vowel_mid +
                       # stem_vowel_closed +
                       final_consonant_labial +
                       final_consonant_coronal +
                       # final_consonant_velar +
```

```
                      final_consonant_voiceless +
                      # final_consonant_voiced +
                      final_consonant_obstruent +
                      final_consonant_sibilant,
                      data = d2
)

print(formula(lm2))
```

```
## log_odds ~ vowel + stem_vowel_long + stem_vowel_open + stem_vowel_mid +
##     final_consonant_labial + final_consonant_coronal + final_consonant_voiceless +
##     final_consonant_obstruent + final_consonant_sibilant
```

```
varImpPlot(rf2)
```




```
tidy(lm2, conf.int = T) %>%
  select(-p.value) %>%
  kable(digits = 2)
```

term	estimate	std.error	statistic	conf.low	conf.high
(Intercept)	-1.38	0.22	-6.31	-1.81	-0.95
vowelé	1.30	0.11	11.61	1.08	1.52

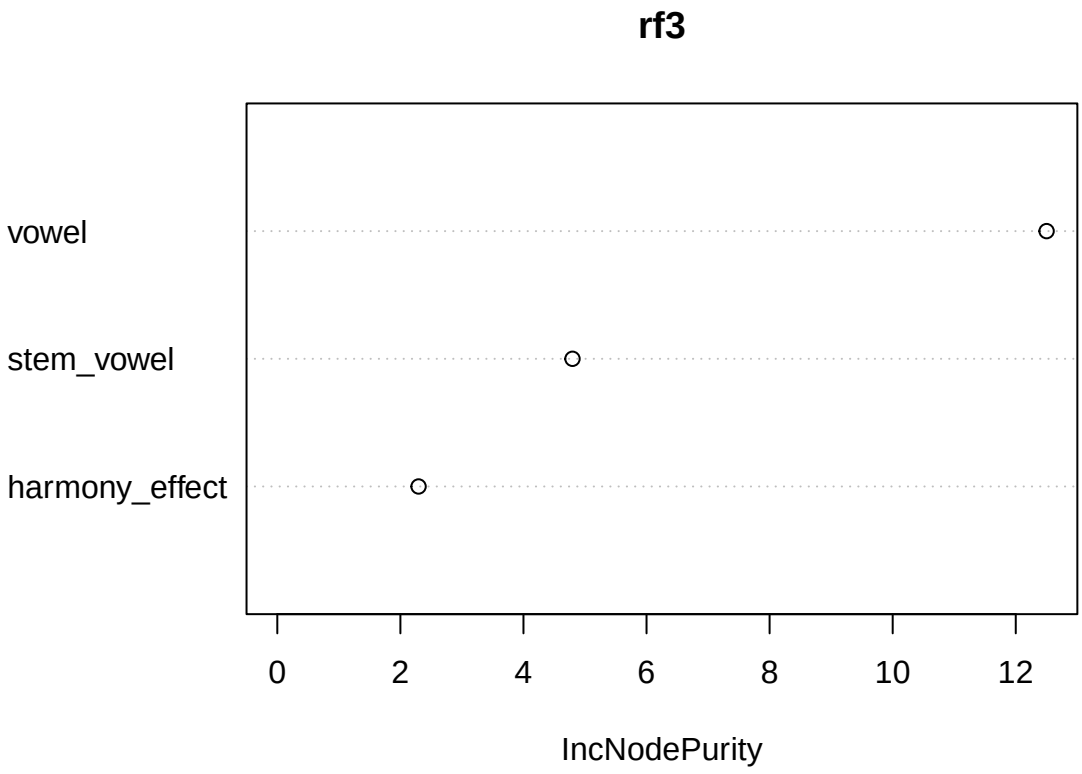
term	estimate	std.error	statistic	conf.low	conf.high
stem_vowel_longTRUE	0.06	0.13	0.44	-0.20	0.31
stem_vowel_openTRUE	0.22	0.14	1.59	-0.06	0.50
stem_vowel_midTRUE	0.08	0.16	0.50	-0.23	0.39
final_consonant_labialTRUE	-0.22	0.23	-0.95	-0.68	0.24
final_consonant_coronalTRUE	0.25	0.17	1.51	-0.08	0.59
final_consonant_voicelessTRUE	0.16	0.15	1.07	-0.14	0.46
final_consonant_obstruentTRUE	-0.15	0.15	-0.97	-0.45	0.16
final_consonant_sibilantTRUE	0.18	0.16	1.10	-0.14	0.50

Category label (baseline / lexical / phonetic, from Patay and Rácz)

```
d3 = filter(d2, !is.na(harmony_effect))

rf3 = randomForest(log_odds ~ vowel + stem_vowel + harmony_effect, data = d3)
lm3 = lm(log_odds ~ vowel + stem_vowel + harmony_effect, data = d3)

print(formula(lm3))
```

```
## log_odds ~ vowel + stem_vowel + harmony_effect
```

```
varImpPlot(rf3)
```



```
tidy(lm3, conf.int = T) %>%
  select(-p.value) %>%
  kable(digits = 2)
```

term	estimate	std.error	statistic	conf.low	conf.high
(Intercept)	-0.73	0.27	-2.68	-1.28	-0.17
vowelé	1.37	0.22	6.28	0.93	1.82
stem_vowelá	-0.04	0.29	-0.14	-0.64	0.56
stem_vowelo	-0.25	0.32	-0.78	-0.90	0.41
stem_voweló	-0.80	0.63	-1.27	-2.09	0.49
stem_vowelu	-0.41	0.28	-1.48	-0.98	0.16
stem_vowelú	-0.42	0.40	-1.05	-1.25	0.40
harmony_effectphonetic_front	-0.08	0.26	-0.31	-0.61	0.45
harmony_effectphonological_front	-0.30	0.24	-1.22	-0.80	0.20