LING48 Accelerated Computational Linguistics
Dartmouth College. Spring 2023

Exercise 2
Bribri Verbal Morphology

I. OBJECTIVE

This exercise has two larger-picture objectives: (i) Familiarizing you with finite-state transducers, and with automata in general. (ii) Attempt to make a "rule-based" model of language and understand the challenges that building such a system on a large scale would entail.

II. INSTRUCTIONS

The Bribri language is spoken by approximately 7000 people in southern Costa Rica. It is part of the Chibchan language family, a family with languages in Costa Rica, Panama and Colombia. (Have you ever heard of "El Dorado"? The legend is based on certain traditions of Chibchan communities in central Colombia).

Bribri verbs use suffixes to indicate different verbal tenses and aspects. The table below shows the details of the conjugation of the verb "to cook", *ali'*.[1]

Transitive verb *ali'* 'to cook'					
Active voice					
1. Imperfective (first)	*al-*	*è* ->	10. Imperfective habitual	*alè-*	*ke*
2. Perfective improspective	*al-*	*i'*	11. Negative imperfective habitual	*alè-*	*ku*
3. Infinitive	*al-*	*ők*	12. Imperfective potential	*alè-*	*mi*
4. Imperative	*al-*	*ő*	13. Imperfective future	*alè-*	*dâ*
5. Desiderative	*al-*	*a'ku*	14. Negative imperfective future	*alè-*	*pa*
6. Negative imperative	*al-*	*ar*			
7. Purposive (for a purpose)	*al-*	*ó*			
8. Adversative	*al-*	*a'*			
9. Perfective prospective	*al-*	*é*			
Middle voice					
15. Imperfective (first)	*alì-*	*r* ->	20. Imperfective habitual	*alìr-*	*ke*
16. Perfective improspective	*alì-*	*ne*	21. Negative imperfective habitual	*alìr-*	*ku*
17. Infinitive	*alì-*	*nuk*	22. Imperfective potential	*alìr-*	*mi*
18. Perfective prospective	*alì-*	*na*	23. Imperfective future	*alìr-*	*dâ*
19. Purposive (for a purpose)	*alì-*	*nó*	24. Negative imperfective future	*alìr-*	*pa*
			25. Anterior	*alír-*	*ule*

[1] Bribri morphology has a lot of interesting features. It is ergative, it has numeral classifiers, it has positional copulas and it uses an Iroquois kinship system!

This table shows the different morphemes for each of the verbal forms. The second column should be the input, and the fourth column should be the output (The meaning is encoded using the Leipzig Glossing Rules[2]).

Word in standard orthography	**Form that you can provide as input**	**IPA**	**Morpheme boundaries**	**Meaning (I use "I" as an example, but it could be any person)**
Active voice				
1. alè	aleH	aɹ.ˈɹeɬ	al-eH cook-IPFV	Imperfective *I cook, I used to cook*
2. ali'	ali'	aɹ.ˈɹiʔɻ	al-i' cook-THEME.PFV.IMPROSP	Perf. Improspective *I cooked [before today].*
3. alǒk	aloqFk	aɹ.ˈɹʊkꓶ	al-oqFk cook-INF	Infinitive *to cook*
4. alǒ	aloqF	aɹ.ˈɹʊꓶ	al-oqF cook-IMP	Imperative *(you) cook!*
5. ala'ku	ala'kux	aɹ.ˈɹaɬ.kũɬ	al-a'kux cook-DESIDERATIVE	Desiderative *I want to cook*
6. alar	alar	aɹ.ˈɹarɹ	al-ar cook-IMP.NEG	Imperative negative *don't cook!*
7. aló	aloF	aɹ.ˈɹoꓶ	al-oF cook-PURPOSE	Purposive *(something) for cooking*
8. ala'	ala'	aɹ.ˈɹaʔɻ	al-a' cook-ADVERSATIVE	Adversative *I cooked (because somebody didn't want me to).*
9. alé	aleF	aɹ.ˈɹeꓶ	al-eF cook-PFV.PROSP	Perf. Prospective *I cooked [today].*
10.alèke	aleHkex	aɹ.ˈɹeɬ.kẽɬ	al-eH-kex cook-IPFV-HABIT	Imperf. Habitual *I cook [often]*
11.alèku	aleHkux	aɹ.ˈɹeɬ.kũɬ	al-eH-kux cook-IPFV-HABIT.NEG	Neg. Imperf. Habitual *I don't cook [often]*
12.alèmi	aleHmix	aɹ.ˈɹeɬ.mĩɬ	al-eH-mix cook-IPFV-POT	Imperf. Potential *I would cook; I could cook*
13.alèdâ	aleHda^	aɹ.ˈɹeɬ.daɻ	al-eH-da^ cook-IPFV-FUT	Imperf. Future *I will cook.*
14.alèpa	aleHpa	aɹ.ˈɹeɬ.paɬ	al-eH-pa cook-IPFV-FUT.NEG	Neg. Imperf. Future *I won't cook.*
Middle voice				
15.alìr	aliHr	aɹˈɹirɬ	al-iHr cook-THEME.MID.IPFV	Imperfective *It is being cooked, people cook it*
16.alìne	aliHnex	aɹˈɹiɬ.nẽɬ	al-iHn-ex cook-THEME.MID-PFV.IMPROSP	Perf. Improspective *It became cooked, it was cooked [before today]*
17.alìnuk	aliHnuxk	aɹˈɹiɬ.nũkɬ	al-iHn-uxk cook-THEME.MID-INF	Infinitive *to be cooked*
18.alìna	aliHnax	aɹˈɹiɬ.nãɬ	al-iHn-ax cook-THEME.MID-PFV.PROSP	Perf. Prospective *It was cooked [today]*

[2] https://www.eva.mpg.de/lingua/resources/glossing-rules.php

Word in standard orthography	Form that you can provide as input	IPA	Morpheme boundaries	Meaning (I use "I" as an example, but it could be any person)
19. alìnó	aliHnoFx	aɭˈɹiɽˌnõꓶ	al-iHn-oFx cook-THEME.MID-PURPOSE	Purposive *(something) so that X is cooked.*
20.alìrkᵉ	aliHrkex	aɭˈɹirɽ.kẽꟻ	al-iHr-kex cook-THEME.MID.IPFV-HABIT	Imperf. Habitual *It is cooked [often]*
21.alìrkᵘ	aliHrkux	aɭˈɹirɽ.kũꟻ	al-iHr-kux cook-THEME.MID.IPFV-HABIT.NEG	Neg. Imperf. Habitual *It isn't cooked [often]*
22.alìrmᵢ	aliHrmix	aɭˈɹirɽ.mĩꟻ	al-iHr-mix cook-THEME.MID.IPFV-POT	Imperf. Potential *It would be cooked, it could be cooked*
23.alìrdâ	aliHrda^	aɭ.ˈɹirɽ.daʌ	al-iHr-da^ cook-THEME.MID.IPFV-FUT	Imperf. Future *It shall be cooked.*
24.alìrpa	aliHrpa	aɭ.ˈɹirɽ.paꟻ	al-iHr-pa cook-THEME.MID.IPFV-FUT.NEG	Neg. Imperf. Future *It shall not be cooked.*
25.alírule	aliFrule	aɭˈɹiɽ.ɾuꟻ.ɹeꟻ	al-iFrule cook-ANTERIOR	Anterior. *I have cooked.*

Your homework will be to modify the template on `hw2_openfst_template.ipynb` and create two finite state transducers:

(1) A transducer that takes Bribri words as input and gives you the separate morphemes as output. **You only need to do words #1-5 and #15-17. These are eight words. You do not need to do all 25.**

Input: aleH
Output: al-eH

(2) A transducer that takes Bribri words as input and gives you the meaning of the morphemes as output. **You only need to do words #1-5 and #15-17. These are eight words. You do not need to do all 25.**

Input: aleH
Output: cook-IPFV

(3) Expand the same FSTs so that they can conjugate two more verbs:

Perf. Improspective (Form #2)	Infinitive (Form #3)	Meaning
tsaki'	tsakő́k	'to pop (a bubble)'
bi'	biő́k	'to dig'

I suggest that you use the second column as the input of the system. This column has certain special symbols:

H	High tone (5):	aH = [aɽ]
F	Falling tone (53):	aF = [aꓶ]
^	Rising tone (13):	a^ = [aʌ]
x	Nasal vowel	ax = [ã]
q	Lax vowel	oq = [ʊ]
'	Glottal stop	' = [ʔ]

(If you want to use Unicode, then use the first column as input. However, I do not recommend this, because it is not easy to provide this as input for the system. You can, for example, format the output so that a sequence like *eH* is converted into *è*. However, you do not need to do this).

The output is the two parts of the fourth column (morpheme boundaries). The first transducer should output the first line of the fourth column (e.g. *al-eH*). The second transducer should output the second line of the fourth column (e.g. *cook-IPFV*). **You need to make Finite State Transducers for 8 forms in the table above (forms 1-5, 15-17). You do not need to do this for all 25 verbal forms**.

The whole output of the program should look like this. It should have the 8 forms for the verb "to cook", and it should be display the infinitives of "to pop" and "to dig".

```
01. aleH
    al-eH
    cook-IPFV

02. ali'
    al-i'
    cook-THEME.PFV.IMPROSP

03. aloqFk
    al-oqFk
    cook-INF

04. aloqF
    al-oqF
    cook-IMP

05. ala'kux
    al-a'kux
    cook-DESIDERATIVE

15. aliHr
    al-iHr
    cook-THEME.MID.IPFV

16. aliHnex
    al-iHn-ex
    cook-THEME.MID-PFV.IMPROSP

17. aliHnuxk
    al-iHn-uxk
    cook-THEME.MID-INF

03. tsakoqFk
    tsak-oqFk
    pop-INF

03. bioqFk
    bi-oqFk
    dig-INF
```

III. Environment Instructions

Please run this in Google Colab. Go to https://colab.research.google.com/ and upload the file `hw2_openfst_template_2023.ipynb`. Modify this template so that it can process the Bribri verbs.

You can try to do this in your computer but only if you have an Ubuntu system and a Python 3.7 or older. The `openfst` package is very difficult to install on MacOS, and it is not possible to install it on Windows. Please do not try to run it in these two platforms. **We will not be able to offer any help if you choose to try this**.

IV. EVALUATION

You need to deliver four files: (1) Your code as a Notebook from Google Colab (`.ipynb`) file. (2) The PDF with the structure of the first transducer. (3) The PDF with the structure of the second transducer. (4) A screenshot(s) of your program providing the correct outputs for the ten forms request (eight forms of "to eat" and the infinitives of "pop" and "dig"). You should use the example file `hw2_openfst_template.py` as the basis for your work.

Please be sure to name your symbol tables `morphSymbols` and `translateSymbols`. Also, make sure you name your compiled FSTs `morphFST` and `translateFST`. This is so that we can call them and grade the assignment. Also, make sure you write your name at the beginning of the code, as well as a brief description of what the program does.

Please turn these into Canvas before 11:59 EDT of Sunday, April 16th.

10%	Instructions (having your name in the code, making a short description of the code, using the requested variable names)
10%	General structure (one FST for morphological glossing and one for translation; correct generalization of conjugations)
40%	Correct morpheme divisions (4% per word)
40%	Correct translations (4% per word)