NLP for Endangered Languages of the Amazon

From a Uralic perspective

Jack Rueter, Niko Partanen,
Mika Hämäläinen & Khalid Alnajjar

Introduction

- Our research group works in the University of Helsinki
- **Jack Rueter**, **Niko Partanen**, Khalid Alnajjar & Mika Hämäläinen

- Research interests
  - Morphologically rich languages
  - Non-standard language varieties: dialectal and historical materials
  - Natural language processing pipelines
    - Lemmatization, tagging, dependency parsing, information extraction
  - We usually work with concrete questions that we want to solve
  - We value highly replicability and usability of our work

Some of our recent studies

J Rueter, MFP de Freitas, SDS Facundes, M Hämäläinen, N Partanen 2021: **Apurinã Universal Dependencies Treebank**

M Hämäläinen, N Partanen, J Rueter, K Alnajjar 2021: **Neural Morphology Dataset and Models for Multiple Languages, from the Large to the Endangered**

K Alnajjar, M Hämäläinen, J Rueter, N Partanen 2020: **Ve'rdd. Narrowing the Gap between Paper Dictionaries, Low-Resource NLP and Community Involvement**

J Rueter, M Hämäläinen, N Partanen 2020: **Open-Source Morphology for Endangered Mordvinic Languages**

J Rueter, N Partanen 2019: **Survey of Uralic Universal Dependencies development**

Background of this course

- We taught a course NLP for Endangered Languages in winter 2021
- This is a new edition with more focus in linguistic fieldwork & Amazon region

- This means:
  - More about lexicographic tools and other software used in this context: ELAN, FLEx
  - We use Apurinã treebank widely in examples (+ other South American treebanks)
  - Less focus in actual programming: the goal is to understand how technology can be used
  - How something is exactly done keeps changing

What is Natural Language Processing (NLP)

- Computational linguistics is a field of linguistics and computer science
- The goal is to use automated methods with natural language material
- Examples: morphological and syntactic analysis, named entity recognition etc.
- Basic task: take a string, and get some information out of it

- We normally work with language represented as *text*
- Transcriptions, written documents etc.
- Methods that produce text from other media also important
    - Text recognition, speech recognition

Contemporary Natural Language Processing

In recent years most of the new work is conducted with neural networks
Traditionally rule based and various statistical methods also widely used
Contemporary work often focuses into the largest languages: English, Chinese

Ywa	kaiãa-puku	maky.
3SG.M	ter.muito-DISTR	castanha

‘Ele tinha muita castanha.’

Ywa	PRON	Case=Nom|Gender=Masc|Number=Sing|Person=3|PronType=Prs
kaiãapuku	VERB	?
maky	NOUN	Gender=Fem|Possessed=No
.	PUNCT	_

Three different approaches

1) We write explicit rules about each lexeme and their possible affixes
2) We have a **large** corpus, and we use statistical distributions
3) We have a **large** corpus, and we train a neural network to map the relationship

What do we do with a small corpus?
How large is a large corpus?
In principle the end result is similar.
The analysis is normally presented as attributes of a word form.

Uralic language Morphology

Uralic includes Finno-Ugric and Samoyedic languages of northern Eurasia.

Nouns and verbs have regular morphology

Nouns, and other nominals

Semantic categories of case, number, person, possession, subject, tense, etc.

Verbs, both finite and non-finite

Semantic categories of case, mood, number, object, person, possession, subject, tense, etc.

Much of the morphology and related semantic categories can be aligned

This is what we call **regular morphology**

Regular morphology

Regular morphology ()

The junction of lexical work, affixes and semantic values together.

Lexical work (https://www.akusanat.com/verdd/)

Determining a base form for identification and inflection

Lemma (dictionary form), Stem (base for inflection)

Affixes = morphology

Junction: affix + stem, stem + affix or affix + affix

Semantic values (multilingual dictionaries)

Universal Dependencies

- Traditionally datasets in different languages have been very different!
- No matter what we do, some linguistic data is necessary
- In an ideal situation we have manually corrected high quality resources

- Universal Dependencies is a project that aims to create comparable annotated materials in different languages (mostly succeeding!)

Current UD Languages

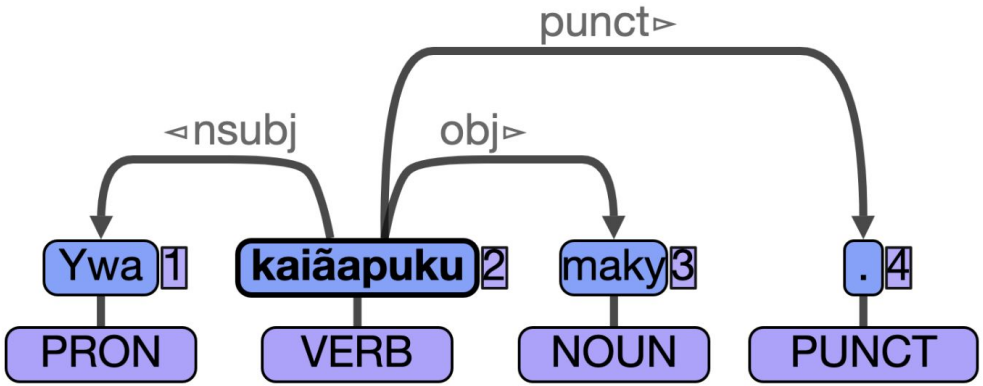
Information about language families (and genera for families with multiple branches) is mostly taken from WALS Online (IE = Indo-European).

	Abaza	1	<1K		Northwest Caucasian
	Afrikaans	1	49K		IE, Germanic
	Akkadian	2	25K		Afro-Asiatic, Semitic
	Akuntsu	1	<1K		Tupian, Tupari
	Albanian	1	<1K		IE, Albanian
	Amharic	2	10K		Afro-Asiatic, Semitic
	Ancient Greek	2	416K		IE, Greek
	Apurina	1	<1K		Arawakan
	Arabic	3	1,042K		Afro-Asiatic, Semitic
	Armenian	1	52K		IE, Armenian
	Assyrian	1	<1K		Afro-Asiatic, Semitic
	Bambara	1	13K		Mande
	Basque	1	121K		Basque
	Beja	1	1K		Afro-Asiatic, Cushitic
	Belarusian	1	305K		IE, Slavic
	Bhojpuri	2	6K		IE, Indic
	Breton	1	10K		IE, Celtic
	Bulgarian	1	156K		IE, Slavic
	Buryat	1	10K		Mongolic
	Cantonese	1	13K		Sino-Tibetan
	Catalan	1	546K		IE, Romance
	Chinese	5	285K		Sino-Tibetan
	Chukchi	1	6K		Chukotko-Kamchatkan
	Classical Chinese	1	269K		Sino-Tibetan
	Coptic	1	48K		Afro-Asiatic, Egyptian
	Croatian	1	199K		IE, Slavic
	Czech	6	3,428K		IE, Slavic
	Danish	2	100K		IE, Germanic
	Dutch	2	306K		IE, Germanic
	English	10	1,880K		IE, Germanic
	Erzya	1	17K		Uralic, Mordvin
	Estonian	2	507K		Uralic, Finnic
	Faroese	2	50K		IE, Germanic
	Finnish	4	397K		Uralic, Finnic

- Apurinã (Arawak)
- Cusco Quechua (Quechuan)

- Portuguese (also Brazilian)

- Akuntsu (Tupian)
- Guajajara (Tupian)
- Kaapor (Tupian)
- Makurap (Tupian)
- Mbya Guarani (Tupian)
- Munduruku (Tupian)
- Tupinamba (Tupian)
- Karo (Tupian)



Ywa kaiãa-puku maky.
3SG.M ter.muito-DISTR castanha
‘Ele tinha muita castanha.’

Ywa	PRON	Case=Nom|Gender=Masc|Number=Sing|Person=3|PronType=Prs
kaiãapuku	VERB	?
maky	NOUN	Gender=Fem|Possessed=No
.	PUNCT	_

Basic annotation types

Lemma: Dictionary headword (usually underived)

Part of speech: ? classes used in UD

Morphological features: Shared tags, language specific set in use

Syntactic label: Subject, object, auxiliary…

Syntactic relationship: Link to parent word (as defined in UD conventions)

MORE ABOUT ANNOTATIONS

Parts of speech = ADJ, ADP, ADV, AUX, CCONJ, DET, NOUN, NUM, PART, PRON, PROPN, PUNCT, SCONJ, VERB, X

Morphological features =

Case=Com|Dat|Loc|Nom|Tem,

Gender=Fem|Masc, Gender[obj]=Masc, Gender[psor]=Fem|Masc, Gender[subj]=Masc,

Number=Plur|Sing, Number[obj]=Plur|Sing, Number[psor]=Plur|Sing, Number[subj]=Plur|Sing,

Person=3, Person[obj]=3,Person[psor]=1|3, Person[subj]=3,

AdvType=Tim, Aspect=Prog, Derivation=Proprietive, Possessed=No|Yes, PronType=Prs, VerbForm=Conv|Vnoun, VerbType=Vido

MORE ABOUT ANNOTATIONS

Syntactic labels =

root, punct,

acl, acl:relcl, advcl, advcl:tcl, advmod, advmod:lmod, advmod:neg, advmod:tmod, aux, case, cc, conj, cop, csubj, det, dislocated, mark, nmod, nmod:poss, nsubj, nsubj:cop, nummod, obj, obj:agent, obl, obl:lmod, obl:tmod, xcomp

dep,

Practical examples

We use Python library UralicNLP to demonstrate our approach on Apurinã

https://github.com/mikahama/uralicNLP

pip install uralicNLP

```
> from uralicNLP import uralicApi
> uralicApi.download("apu") # model's are updated daily!
```

Same analysers can also be accessed through other tools. See also a related package, murre, that uses neural networks to normalize non-standard text:

https://github.com/mikahama/murre

Analyse one form

```
python3
>>> from uralicNLP import uralicApi
>>> uralicApi.download("apu")
>>> uralicApi.analyze("awary", "apu")
[('awa+V+ScSg3M+Oc3M', 0.0), ('awa+V+Oc3M', 0.0)]
>>> uralicApi.analyze("awaru", "apu")
[('awa+V+ScSg3M+Oc3F', 0.0), ('awa+V+Oc3F', 0.0)]
```

Analyse a sentence

```
python3
>>> from uralicNLP import uralicApi
>>> for word in "Ywa kaiãapuku maky .".split(" "):
...     print(uralicApi.analyze(word,"apu"))
...
[('ywa+Pron+Pers+Sg3+Msc+Nom', 0.0)]
[]
[('maky+N+Msc+Sg', 0.0)]
[('.+CLB', 0.0)]
```

Generate all word forms (that the computer knows)

```
>>> uralicApi.get_all_forms("ximaky", "N", "apu")
```

```
[('Aximaky:ximaky+N+Msc+Sg+PxPl1', 0.0), ..., ('aximaky:ximaky+N+Msc+Sg+PxPl1', 0.0),
('aximakywaku:ximaky+N+Msc+Pl+PxPl1', 0.0), ('hĩximaky:ximaky+N+Msc+Sg+PxPl2', 0.0),
('hĩximakywaku:ximaky+N+Msc+Pl+PxPl2', 0.0), ('hĩximaky:ximaky+N+Msc+Sg+PxPl2', 0.0),
('hĩximakywaku:ximaky+N+Msc+Pl+PxPl2', 0.0), ('iximaky:ximaky+N+Msc+Sg+PxSg3+PxMsc',
0.0), ('iximakyna:ximaky+N+Msc+Sg+PxPl3+PxMsc', 0.0),
('iximakywaku:ximaky+N+Msc+Pl+PxSg3+PxMsc', 0.0),
('iximakywakuna:ximaky+N+Msc+Pl+PxPl3+PxMsc', 0.0),
('nhiximaky:ximaky+N+Msc+Sg+PxSg1', 0.0), ('nhiximakywaku:ximaky+N+Msc+Pl+PxSg1', 0.0),
('piximaky:ximaky+N+Msc+Sg+PxSg2', 0.0), ('piximakywaku:ximaky+N+Msc+Pl+PxSg2', 0.0),
('uximaky:ximaky+N+Msc+Sg+PxSg3+PxFem', 0.0),
('uximakyna:ximaky+N+Msc+Sg+PxPl3+PxFem', 0.0),
('uximakywaku:ximaky+N+Msc+Pl+PxSg3+PxFem', 0.0),
('uximakywakuna:ximaky+N+Msc+Pl+PxPl3+PxFem', 0.0), ..., ('ximaky:ximaky+N+Msc+Sg', 0.0),
('ximakywaku:ximaky+N+Msc+Pl', 0.0)]
```

Find all unknown word forms

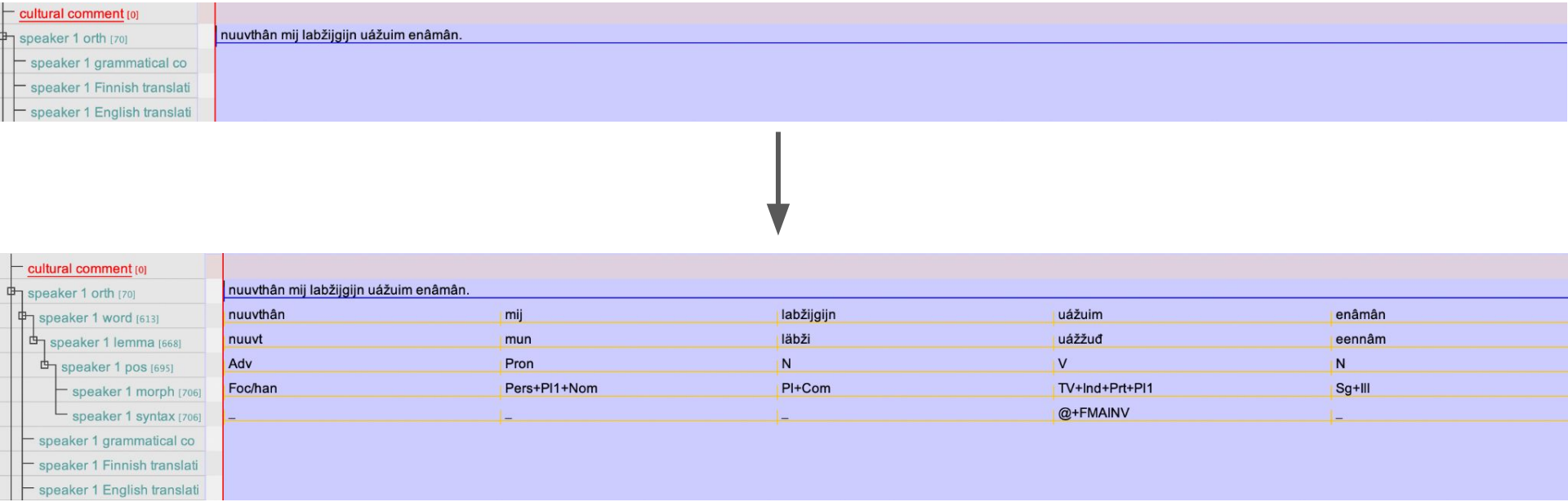
```
>>> from uralicNLP import uralicApi
>>>
>>> for word in "Ywa kaiãapuku maky .".split(" "):
…         if not uralicApi.analyze(word, "apu"):
…             print(f"Cannot analyze: {word}")
Cannot analyze: kaiãapuku
```

Possible applications

- Testing if all transcribed forms can be analysed
- Testing that all morphology that occurs in the language is described
- Testing that all lexemes are in the dictionary (analyser uses a dictionary)
- Testing how well we understand complex processes:
  - If we can describe it computationally, we have understood it (some way!)
  - Annotating a treebank manually is also one test for syntactic structures

- Language technology is also useful for the community! Spell-checkers, keyboards, online dictionaries, speech technology, access to materials etc…

Further example: Writing annotations directly to ELAN



“So in this very way we got them ashore with belts”

Source: Giellagas Corpus of Spoken Saami Languages (Inari Saami portion)

Our repository for related work: https://github.com/langdoc/elan-fst

Discussion

- NLP approaches can be adapted to data in ELAN and FLEx
- It is possible to extract the materials from these formats
- It is important to get familiar with different annotation practices

- By using computational methods we can process our materials effectively
- We can search and analyse materials that are in formats useful for NLP too
- Computational description serves also as one type of documentation

Some links

https://www.akusanat.com/verdd/ (Lexical work)

https://github.com/giellalt/lang-apu/ (Morphological work, etc.)

Thank you!