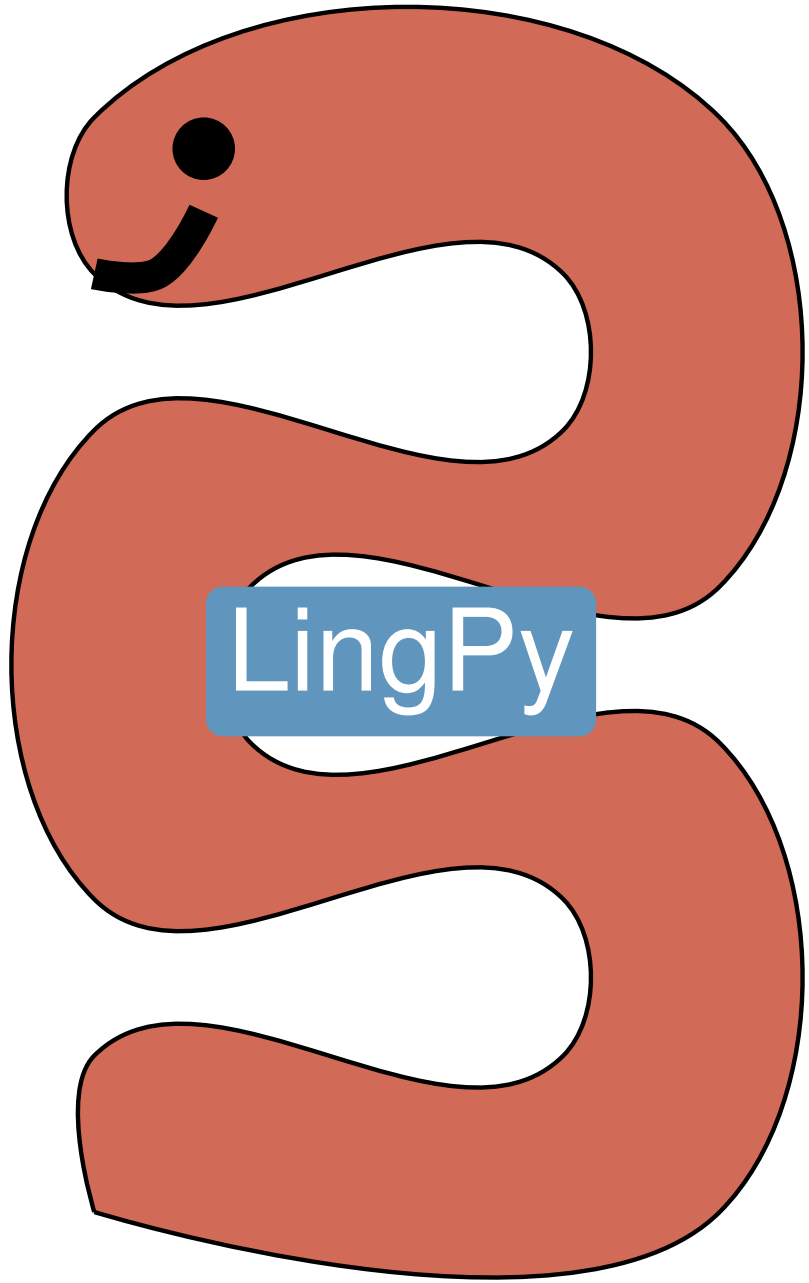
The LingPy library for quantitative historical linguistics

Background, theory, and application

Johann-Mattis List

Forschungszentrum Deutscher Sprachatlas
Philipps-Universität Marburg

15.02.2014


LingPy

What is LingPy?

What is LingPy?

- Python library for automatic tasks in historical linguistics

What is LingPy?

- Python library for automatic tasks in historical linguistics
- project homepage at `http://lingpy.org`

What is LingPy?

- Python library for automatic tasks in historical linguistics
- project homepage at `http://lingpy.org`
- code base for developers at `https://github.com/lingpy/lingpy`

What is LingPy?

- Python library for automatic tasks in historical linguistics
- project homepage at `http://lingpy.org`
- code base for developers at `https://github.com/lingpy/lingpy`
- supports Python2 and Python3

What is LingPy?

- Python library for automatic tasks in historical linguistics
- project homepage at `http://lingpy.org`
- code base for developers at `https://github.com/lingpy/lingpy`
- supports Python2 and Python3
- works on Mac, Linux, and (basically also) Windows

What is LingPy?

- Python library for automatic tasks in historical linguistics
- project homepage at `http://lingpy.org`
- code base for developers at `https://github.com/lingpy/lingpy`
- supports Python2 and Python3
- works on Mac, Linux, and (basically also) Windows
- current release: 2.2

What is LingPy?

- Python library for automatic tasks in historical linguistics
- project homepage at `http://lingpy.org`
- code base for developers at `https://github.com/lingpy/lingpy`
- supports Python2 and Python3
- works on Mac, Linux, and (basically also) Windows
- current release: 2.2
- offers methods for sequence modeling, phonetic alignment, cognate and borrowing detection, and tools for data manipulation and visualization

What can be done with LingPy?

- tokenize phonetic sequences

tʰɔxtɐ	tʰ ɔ x t ɐ
dɔːtɚ	d ɔː t ɚ
dɔtːa	d ɔ tː a

What can be done with LingPy?

- align phonetic sequences

tʰɔxtɐ	tʰ	ɔ	x	t	ɐ
dɔːtɐ̯	d	ɔː	-	t	ɐ̯
dɔtːa	d	ɔ	-	tː	a

What can be done with LingPy?

- search for cognates

Concept: *belly* (ID: 4)

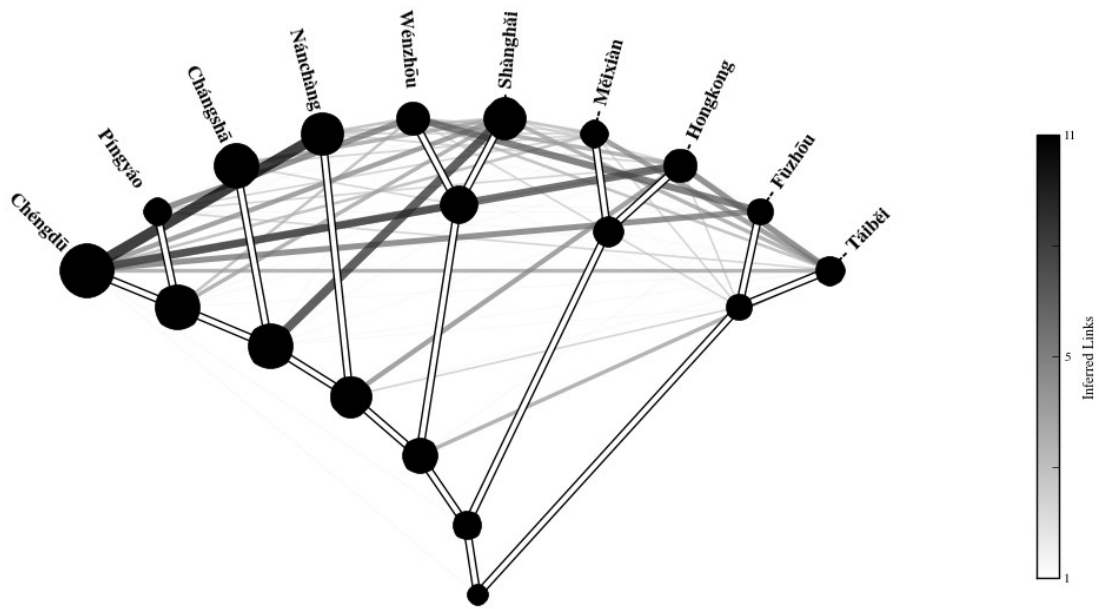
CogID	Language	Entry	Aligned Entry
6	Danish	ɔnʌliwˀ	--
7	Dutch	bœyk	b œy k
7	German	baux	b au x
7	Norwegian	bʉ:k	b ʉ: k
7	Swedish	buk	b u k
8	English	bɛlɪ	--
9	Danish	mæ:və	m æ: v ə
9	Norwegian	mɑ:gə	m ɑ: g ə
9	Swedish	ma:ge	m a: g e
10	Icelandic	kʰvɪ:ðʏr	--

Concept: *big* (ID: 5)

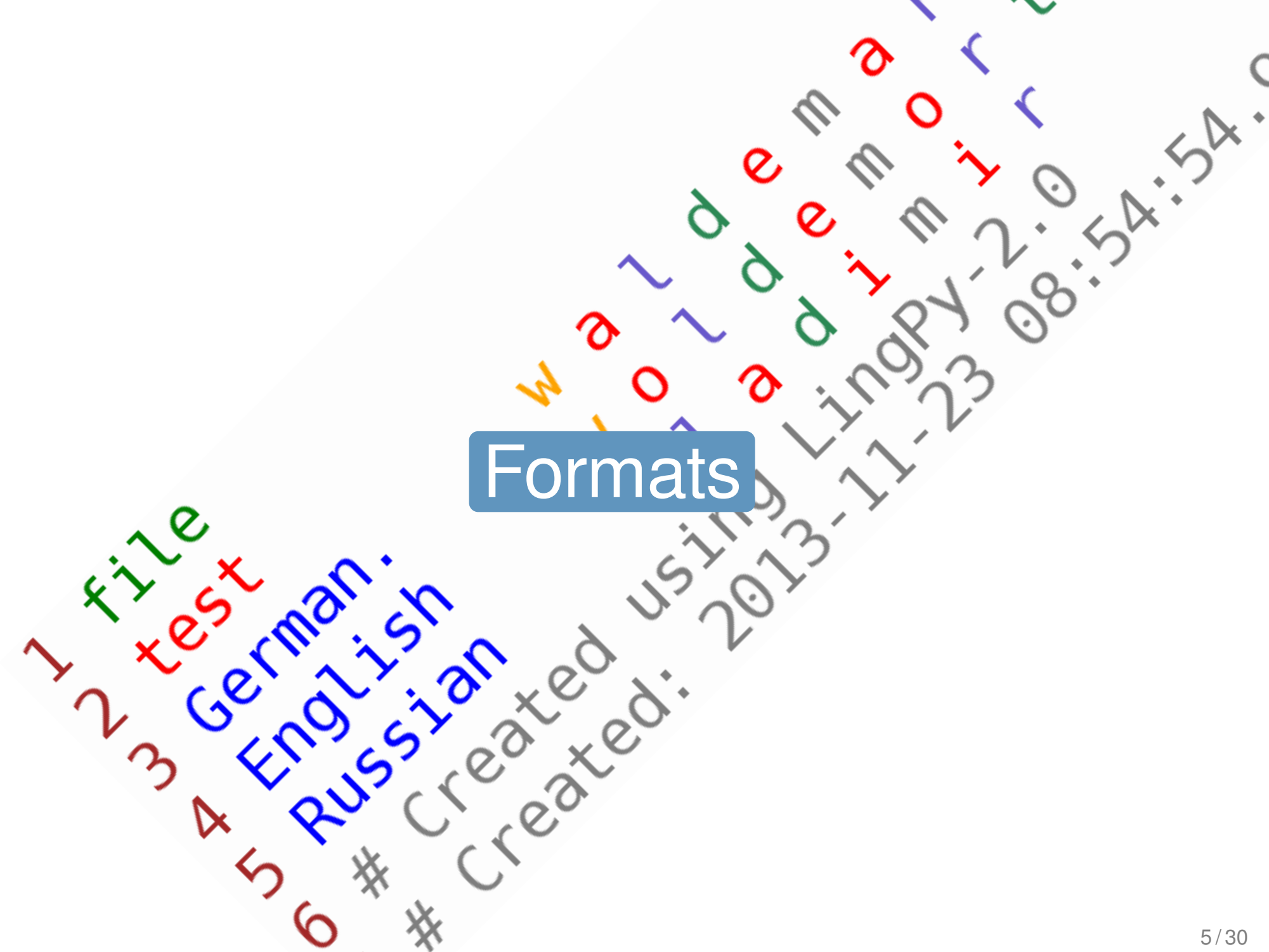
CogID	Language	Entry	Aligned Entry
11	Danish	sdoˀʁ	s d oˀ ʁ

What can be done with LingPy?

- search for borrowings



Formats


1 file
2 test
3 German.
4 English
5 Russian
6 # Created using LingPy-2.0
Created: 2013-11-23 08:54:54.
w a l d e m a r
w o l d e m o r t
a d i m i r

Formats: Basics

ID	CONCEPT	COUNTERPART	IPA	DOCULECT	COGID
1	hand	Hand	hant	German	1
2	hand	hand	hænd	English	1
3	hand	рука	ruka	Russian	2
4	hand	рука	ruka	Ukrainian	2
5	leg	Bein	bain	German	3
6	leg	leg	lɛg	English	4
7	leg	нога	noga	Russian	5
8	leg	нога	noha	Ukrainian	5
9	Woldemort	Waldemar	valdemar	German	6
10	Woldemort	Woldemort	wɔldemɔrt	English	6
11	Woldemort	Владимир	vladimir	Russian	6
12	Woldemort	Володимир	volodimir	Ukrainian	6
13	Harry	Harald	haralt	German	7
14	Harry	Harry	hæri	English	7
15	Harry	Гарри	gari	Russian	7
16	Harry	Гаррі	hari	Ukrainian	7

Formats: Basics

CONCEPT	GERMAN	ENGLISH	RUSSIAN	UKRAINIAN
hand	Hand	hand	рука	рука
leg	Bein	leg	нога	нога
Woldemort	Waldemar	Woldemort	Владимир	Володимир
Harry	Harald	Harry	Гарри	Гаррі

+ Orthography +

Formats: Basics

CONCEPT	GERMAN	ENGLISH	RUSSIAN	UKRAINIAN
hand	hant	hænd	ruka	ruka
leg	bain	lɛg	noga	noha
Woldemort	valdəmar	wɔldəmɔrt	vladimir	volodimir
Harry	haralt	hæri	gari	hari

+ Entries in IPA +

Formats: Basics

CONCEPT	GERMAN	ENGLISH	RUSSIAN	UKRAINIAN
hand	1	1	2	2
leg	3	4	5	5
Woldemort	6	6	6	6
Harry	7	7	7	7

+ Cognate-IDs +

Formats: Key-Value Extension

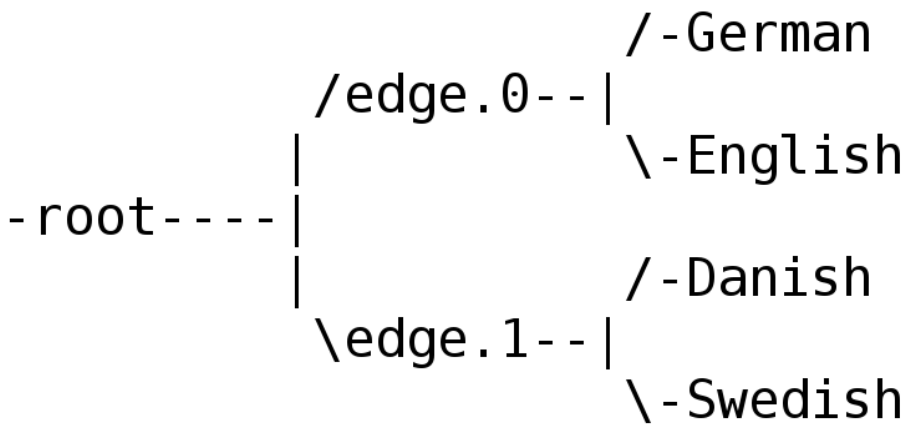
```
# Wordlist

# META
@author: Potter, Harry
@date: 2013-04-02
@tree: ((German,English),(Russian,Ukrainian));
@note: Use the data with care, it might have been charmed...

# DATA
ID    CONCEPT      COUNTERPART    IPA          DOCULECT      COGID
1     hand         Hand           hant         German        1
2     hand         hand           hænd         English       1
3     hand         рука           ruka         Russian       2
4     hand         рука           ruka         Ukrainian     2
5     leg          Bein           bain         German        3
...   ...          ...            ...          ...           ...
```

Formats: Further Extensions

```
# Wordlist

# META
@author:Potter, Harry
@date:2012-11-07

# JSON
<json>
{
    "taxa": [
        "English",
        "German",
        "Russian",
        "Ukrainian"
    ]
}
</json>
```

Formats: Further Extensions

```
# DISTANCES
<dst>
 4
English   0.000000 0.333333 0.666667 0.666667
German    0.333333 0.000000 0.666667 0.666667
Russian   0.666667 0.666667 0.000000 0.000000
Ukrainian 0.666667 0.666667 0.000000 0.000000
</dst>

# DATA
ID   CONCEPT COUNTERPART     IPA     DOCULECT COGID
#
1    hand    Hand            hant    German   1
2    hand    hand            hænd    English  1
...  ...     ...             ...     ...      ...
```



Representation

```
((German,English)(Danish,Swedish));
```

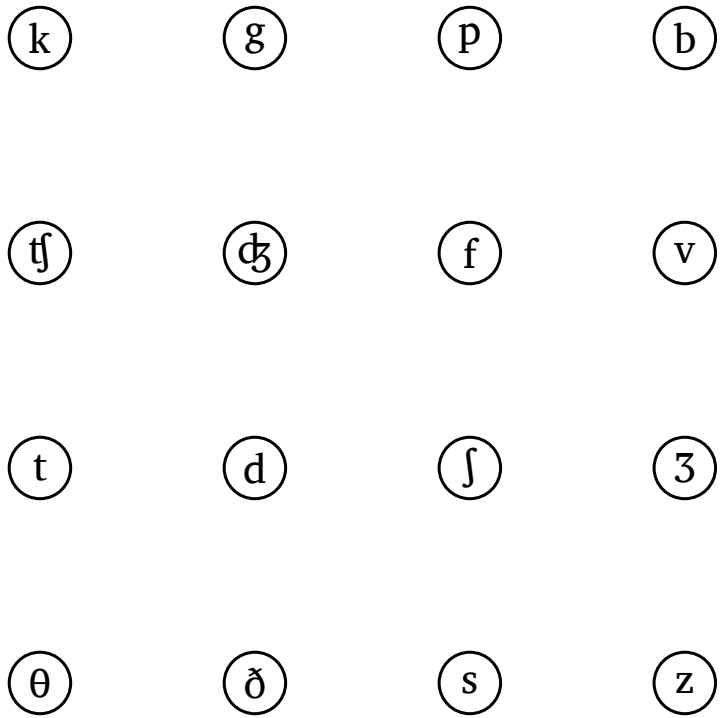
Sound Classes: General Idea

Sound Classes

Sound Classes: General Idea

Sound Classes

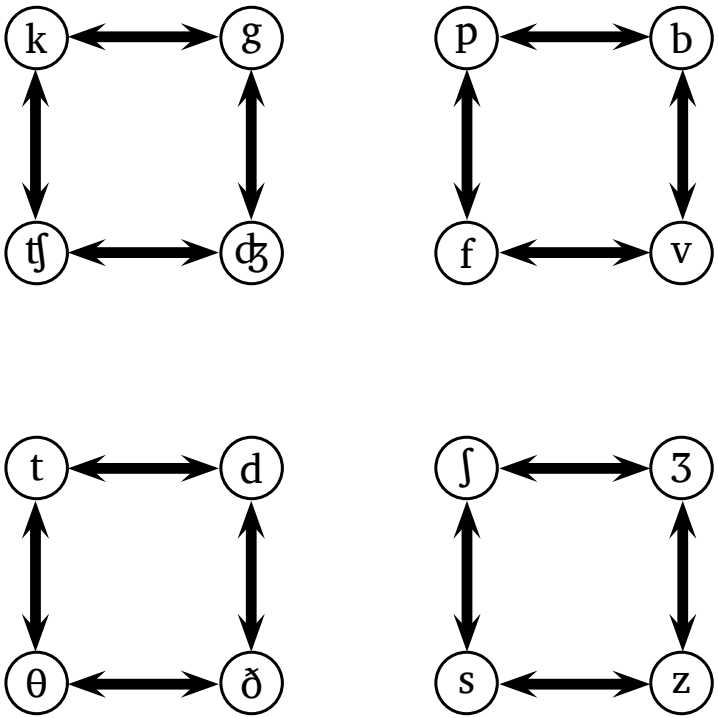
Sounds which often occur in correspondence relations in genetically related languages can be clustered into classes (types). It is assumed “that phonetic correspondences inside a ‘type’ are more regular than those between different ‘types’ ” (Dolgopolsky 1986: 35).



Sound Classes: General Idea

Sound Classes

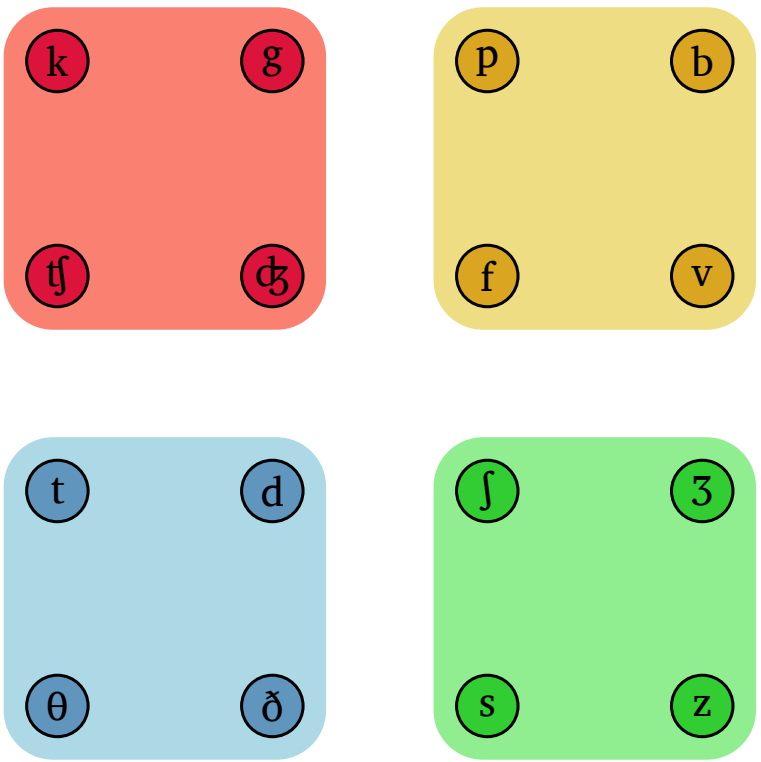
Sounds which often occur in correspondence relations in genetically related languages can be clustered into classes (types). It is assumed "that phonetic correspondences inside a 'type' are more regular than those between different 'types' " (Dolgopolsky 1986: 35).



Sound Classes: General Idea

Sound Classes

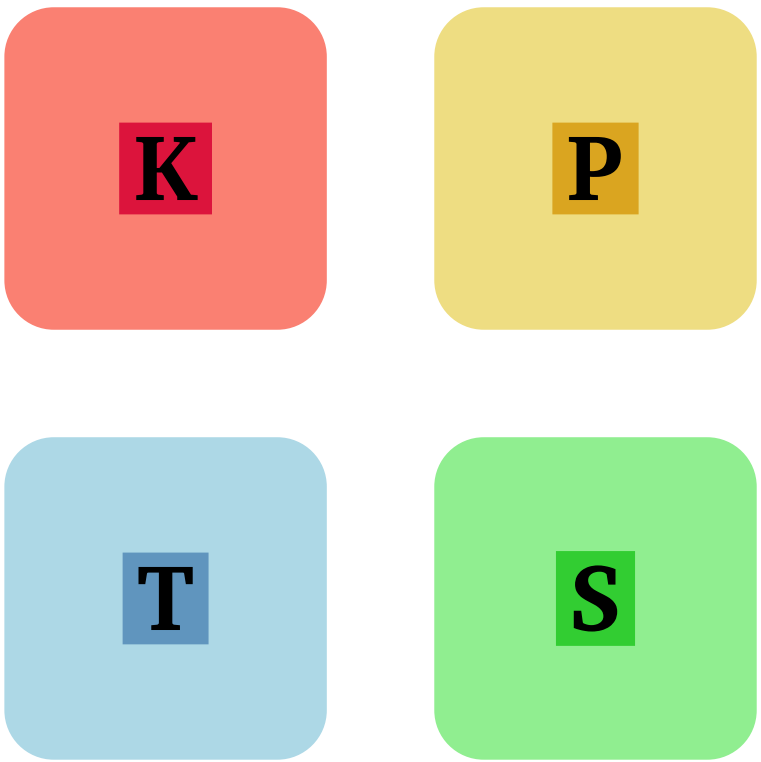
Sounds which often occur in correspondence relations in genetically related languages can be clustered into classes (types). It is assumed "that phonetic correspondences inside a 'type' are more regular than those between different 'types' " (Dolgopolsky 1986: 35).



Sound Classes: General Idea

Sound Classes

Sounds which often occur in correspondence relations in genetically related languages can be clustered into classes (types). It is assumed “that phonetic correspondences inside a ‘type’ are more regular than those between different ‘types’ ” (Dolgopolsky 1986: 35).



Sound Classes: Scoring Functions

- LingPy offers default scoring functions for three standard sound-class models (ASJP, SCA, DOLGO).
- The standard models vary regarding the roughness by which the continuum of sounds is split into discrete classes.
- The scoring functions are based on empirical data on sound correspondence frequencies (ASJP model, Brown et al. 2013), and on general theoretical models of the directionality and probability of sound change processes (SCA, DOLGO, see List 2012b for details).
- Scoring functions can be easily expanded by the user.

Prosodic Strings

Prosodic Strings

- Sound change occurs more frequently in prosodically **weak** positions (Geisler 1992).

Prosodic Strings

- Sound change occurs more frequently in prosodically **weak** positions (Geisler 1992).
- Given a **sonority profile**, one can distinguish positions that differ regarding their **prosodic context**.

Prosodic Strings

- Sound change occurs more frequently in prosodically **weak** positions (Geisler 1992).
- Given a **sonority profile**, one can distinguish positions that differ regarding their **prosodic context**.
- **Prosodic strings** indicate different prosodic contexts for each segment.

Prosodic Strings

- Sound change occurs more frequently in prosodically **weak** positions (Geisler 1992).
- Given a **sonority profile**, one can distinguish positions that differ regarding their **prosodic context**.
- **Prosodic strings** indicate different prosodic contexts for each segment.
- Substitution scores and gap penalties can be modified depending on the underlying prosodic string.

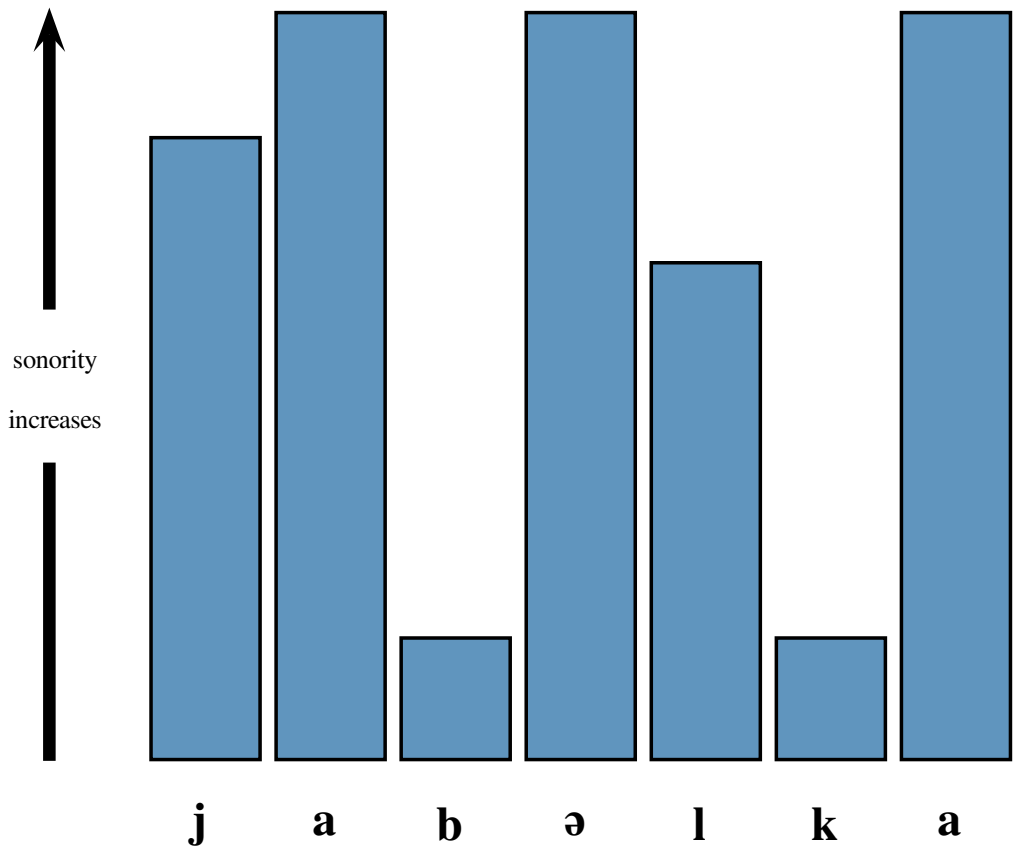
Prosodic Strings

- Sound change occurs more frequently in prosodically **weak** positions (Geisler 1992).
- Given a **sonority profile**, one can distinguish positions that differ regarding their **prosodic context**.
- **Prosodic strings** indicate different prosodic contexts for each segment.
- Substitution scores and gap penalties can be modified depending on the underlying prosodic string.
- Prosodic strings are an alternative to *n*-gram approaches: they also handle context, but their advantage is that they are more abstract and less data-dependent than *n*-grams.

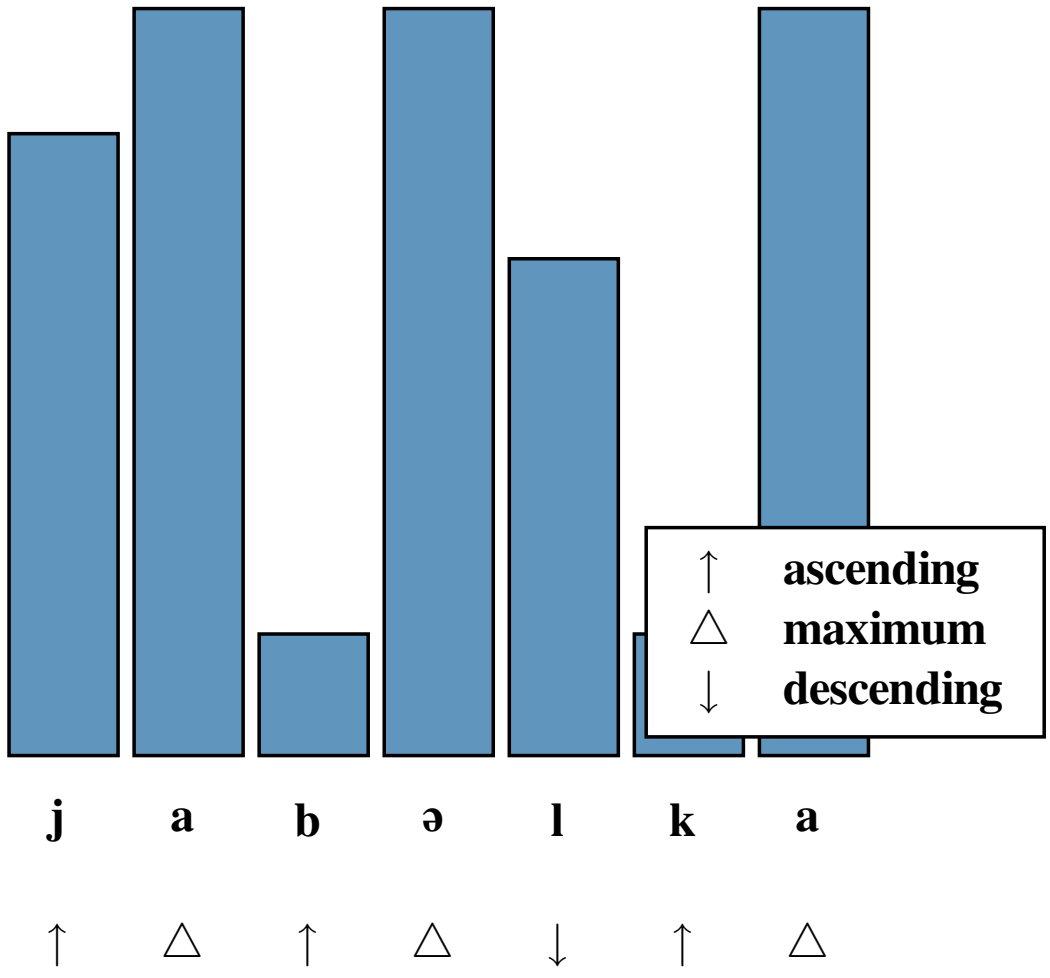
Prosodic Strings

j a b ə l k a

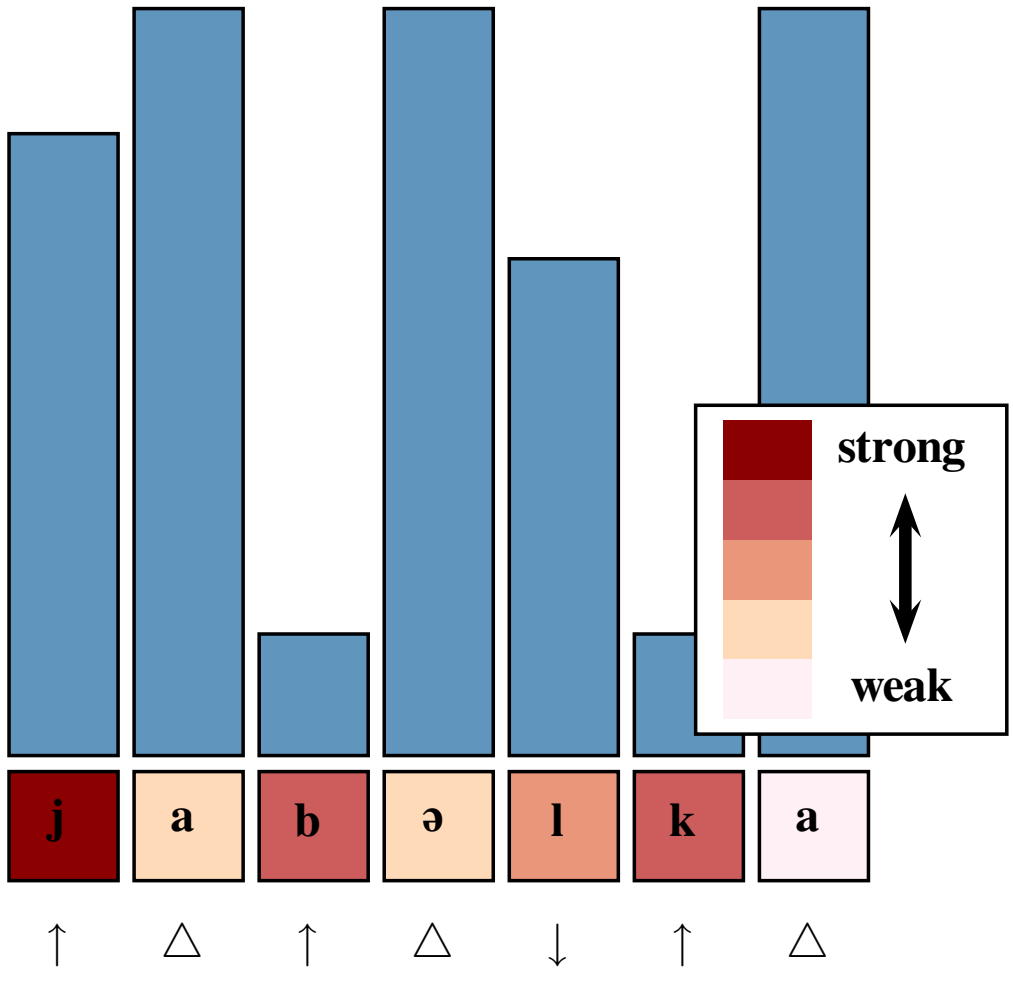
Prosodic Strings



Prosodic Strings



Prosodic Strings



Prosodic Strings

phonetic sequence	j	a	b	ə	l	k	a
SCA model	J	A	P	E	L	K	A
ASJP model	y	a	b	I	l	k	a
DOLGO model	J	V	P	V	R	K	V
sonority profile	6	7	1	7	5	1	7
prosodic string	#	v	C	v	c	C	>
Relative Weight	2.0	1.5	1.5	1.3	1.1	1.5	0.7

v o l - d e m o r t

v - l a d i m i r -

v a l - d e m a r -

Analysis

Sound-Class-Based Phonetic Alignment (SCA)

List, JM (2012). “SCA. Phonetic alignment based on sound classes”. In: *New directions in logic, lan- guage, and computation*. Ed. by M Slavkovik and D Lassiter. Berlin and Heidelberg: Springer, 32–51.

Sound-Class-Based Phonetic Alignment (SCA)

List, JM (2012). “SCA. Phonetic alignment based on sound classes”. In: *New directions in logic, lan- guage, and computation*. Ed. by M Slavkovik and D Lassiter. Berlin and Heidelberg: Springer, 32–51.

- method for pairwise and multiple phonetic alignment

Sound-Class-Based Phonetic Alignment (SCA)

List, JM (2012). “SCA. Phonetic alignment based on sound classes”. In: *New directions in logic, lan- guage, and computation*. Ed. by M Slavkovik and D Lassiter. Berlin and Heidelberg: Springer, 32–51.

- method for pairwise and multiple phonetic alignment
- internal sequence representation as *sound classes* and *prosodic strings*

Sound-Class-Based Phonetic Alignment (SCA)

List, JM (2012). “SCA. Phonetic alignment based on sound classes”. In: *New directions in logic, lan- guage, and computation*. Ed. by M Slavkovik and D Lassiter. Berlin and Heidelberg: Springer, 32–51.

- method for pairwise and multiple phonetic alignment
- internal sequence representation as *sound classes* and *prosodic strings*
- supports *global*, *local*, *semi-global*, and *diagonal* alignment analyses

Sound-Class-Based Phonetic Alignment (SCA)

List, JM (2012). “SCA. Phonetic alignment based on sound classes”. In: *New directions in logic, lan- guage, and computation*. Ed. by M Slavkovik and D Lassiter. Berlin and Heidelberg: Springer, 32–51.

- method for pairwise and multiple phonetic alignment
- internal sequence representation as *sound classes* and *prosodic strings*
- supports *global*, *local*, *semi-global*, and *diagonal* alignment analyses
- handles *secondary sequence structures* (morpheme, syllable boundaries)

Sound-Class-Based Phonetic Alignment (SCA)

List, JM (2012). “SCA. Phonetic alignment based on sound classes”. In: *New directions in logic, lan- guage, and computation*. Ed. by M Slavkovik and D Lassiter. Berlin and Heidelberg: Springer, 32–51.

- method for pairwise and multiple phonetic alignment
- internal sequence representation as *sound classes* and *prosodic strings*
- supports *global*, *local*, *semi-global*, and *diagonal* alignment analyses
- handles *secondary sequence structures* (morpheme, syllable boundaries)
- can identify swapped sites in multiple phonetic alignments

Sound-Class-Based phonetic Alignment (SCA)

INPUT
jabl̩ko
jabəlka
jabləkə
japkɔ

Sound-Class-Based phonetic Alignment (SCA)

CONVERSION (1)		
jabl̩ko	→	JAPLKU
jabəlka	→	JAPELKA
jablәkə	→	JAPLEKE
japkɔ	→	JAPKU

Sound-Class-Based phonetic Alignment (SCA)

CONVERSION (2)		
jabḷko	→	#VCVC>
jabəlka	→	#VCVcC>
jablәkə	→	#VCCVC>
japkɔ	→	#VcC>

Sound-Class-Based phonetic Alignment (SCA)

ALIGNMENT							
J	A	P	–	L	–	K	U
J	A	P	E	L	–	K	A
J	A	P	–	L	E	K	E
J	A	P	–	–	–	K	U

Sound-Class-Based phonetic Alignment (SCA)

OUTPUT							
j	a	b	-	l̩	-	k	o
j	a	b	ə	l	-	k	a
j	a	b	-	l	ə	k	ə
j	a	p	-	-	-	k	ɔ

LexStat

List, JM (2012): “LexStat. Automatic detection of cognates in multilingual word-lists”. In: *Proceedings of the EACL 2012 Joint Workshop of Visualization of Linguistic Patterns and Uncovering Language History from Multilingual Resources*.“LINGVIS & UNCLH 2012” (Avignon, 04/23–04/24/2012).

LexStat

List, JM (2012): “LexStat. Automatic detection of cognates in multilingual word-lists”. In: *Proceedings of the EACL 2012 Joint Workshop of Visualization of Linguistic Patterns and Uncovering Language History from Multilingual Resources*.“LINGVIS & UNCLH 2012” (Avignon, 04/23–04/24/2012).

- multilingual and language-specific method for cognate detection

LexStat

List, JM (2012): “LexStat. Automatic detection of cognates in multilingual word-lists”. In: *Proceedings of the EACL 2012 Joint Workshop of Visualization of Linguistic Patterns and Uncovering Language History from Multilingual Resources*.“LINGVIS & UNCLH 2012” (Avignon, 04/23–04/24/2012).

- multilingual and language-specific method for cognate detection
- alignment-based detection of regular sound correspondences

LexStat

List, JM (2012): “LexStat. Automatic detection of cognates in multilingual word-lists”. In: *Proceedings of the EACL 2012 Joint Workshop of Visualization of Linguistic Patterns and Uncovering Language History from Multilingual Resources*.“LINGVIS & UNCLH 2012” (Avignon, 04/23–04/24/2012).

- multilingual and language-specific method for cognate detection
- alignment-based detection of regular sound correspondences
- re-alignment of the data with help of correspondence-based scoring functions

LexStat

List, JM (2012): “LexStat. Automatic detection of cognates in multilingual word-lists”. In: *Proceedings of the EACL 2012 Joint Workshop of Visualization of Linguistic Patterns and Uncovering Language History from Multilingual Resources*.“LINGVIS & UNCLH 2012” (Avignon, 04/23–04/24/2012).

- multilingual and language-specific method for cognate detection
- alignment-based detection of regular sound correspondences
- re-alignment of the data with help of correspondence-based scoring functions
- flat cluster analysis for the detection of cognate sets

LexStat

ID	Taxa	Word	Gloss	GlossID	IPA	...
...	...	...	...	...	...	...
21	German	Frau	woman	20	frau	...
22	Dutch	vrouw	woman	20	vrɑu	...
23	English	woman	woman	20	wʊmən	...
24	Danish	kvinde	woman	20	kvenə	...
25	Swedish	kvinna	woman	20	kviːna	...
26	Norwegian	kvine	woman	20	kvinə	...
...	...	...	...	...	...	...

LexStat

ID	Taxa	Word	Gloss	GlossID	IPA	CogID
...	...	...	...	...	...	...
21	German	Frau	woman	20	frau	1
22	Dutch	vrouw	woman	20	vrɑu	1
23	English	woman	woman	20	wʊ̈mən	2
24	Danish	kvinde	woman	20	kvenə	3
25	Swedish	kvinna	woman	20	kviːna	3
26	Norwegian	kvine	woman	20	kvinə	3
...	...	...	...	...	...	...

LexStat

ID	Taxa	Word	Gloss	GlossID	IPA	CogID
...	...	...	...	...	...	...
21	German	Frau	woman	20	frau	1
22	Dutch	vrouw	woman	20	vrɑu	1
23	English	woman	woman	20	wʊ̈mən	2
24	Danish	kvinde	woman	20	kvenə	3
25	Swedish	kvinna	woman	20	kviːna	3
26	Norwegian	kvine	woman	20	kѵinə	3
...	...	...	...	...	...	...

Phylogeny-Based Borrowing Detection (PhyBo)

List, JM, S Nelson-Sathi, H Geisler, und W Martin (2014). “Networks of lexical borrowing and lateral gene transfer in language and genome evolution”. *BioEssays* 36.2, 141–150.

Phylogeny-Based Borrowing Detection (PhyBo)

List, JM, S Nelson-Sathi, H Geisler, und W Martin (2014). “Networks of lexical borrowing and lateral gene transfer in language and genome evolution”. *BioEssays* 36.2, 141–150.

- phylogeny-based method for borrowing detection

Phylogeny-Based Borrowing Detection (PhyBo)

List, JM, S Nelson-Sathi, H Geisler, und W Martin (2014). “Networks of lexical borrowing and lateral gene transfer in language and genome evolution”. *BioEssays* 36.2, 141–150.

- phylogeny-based method for borrowing detection
- uses parsimony analyses to detect cognate sets which cannot be explained with help of a given reference tree

Phylogeny-Based Borrowing Detection (PhyBo)

List, JM, S Nelson-Sathi, H Geisler, und W Martin (2014). “Networks of lexical borrowing and lateral gene transfer in language and genome evolution”. *BioEssays* 36.2, 141–150.

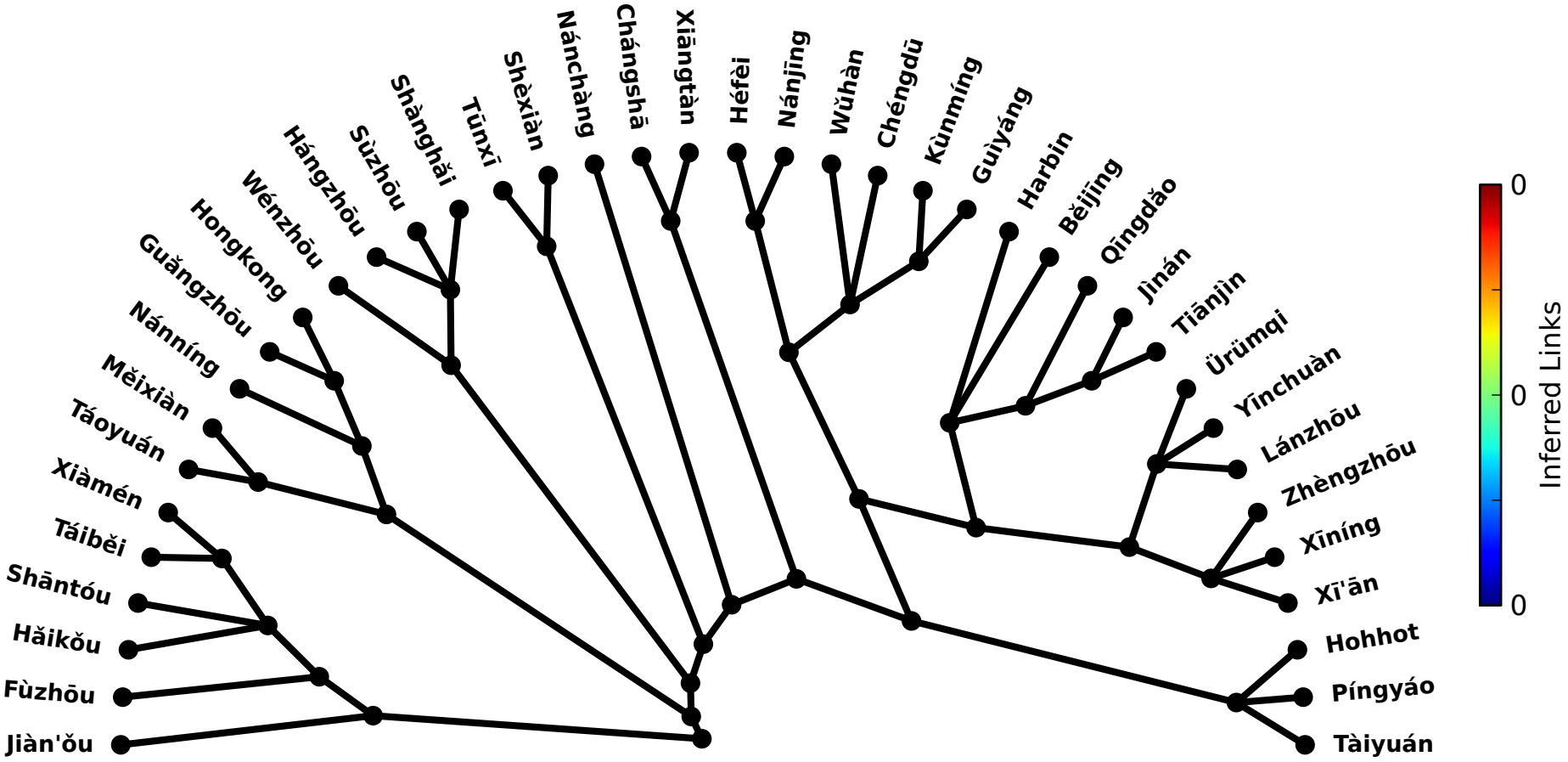
- phylogeny-based method for borrowing detection
- uses parsimony analyses to detect cognate sets which cannot be explained with help of a given reference tree
- selection of the best *weighting model* based on *similar vocabulary size distribution*

Phylogeny-Based Borrowing Detection (PhyBo)

List, JM, S Nelson-Sathi, H Geisler, und W Martin (2014). “Networks of lexical borrowing and lateral gene transfer in language and genome evolution”. *BioEssays* 36.2, 141–150.

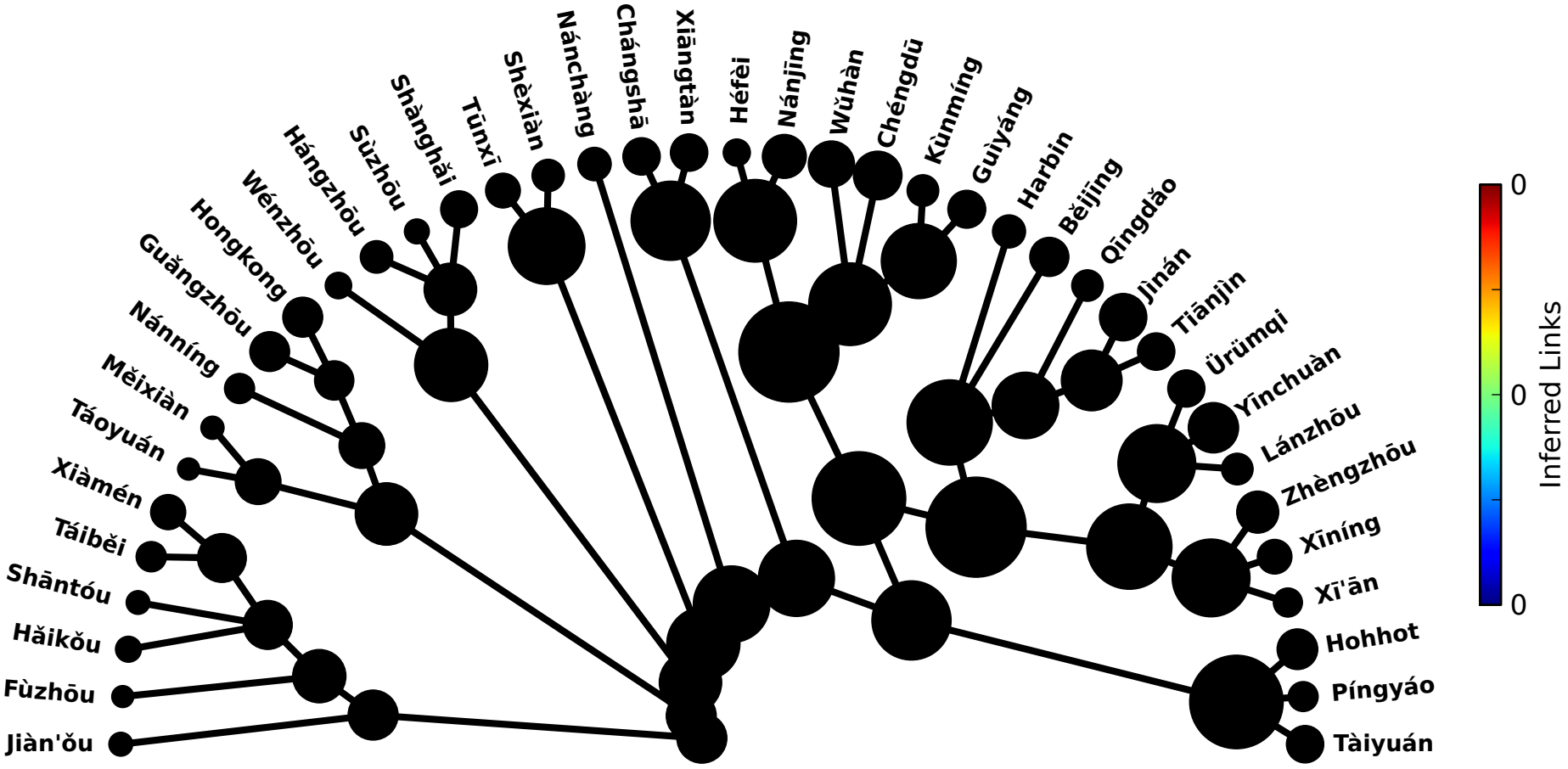
- phylogeny-based method for borrowing detection
- uses parsimony analyses to detect cognate sets which cannot be explained with help of a given reference tree
- selection of the best *weighting model* based on *similar vocabulary size distribution*
- reconstructs a *minimal lateral network* of the data in which the minimal amount of lateral connections inferred by the best model is displayed

Phylogeny-Based Borrowing Detection (PhyBo)



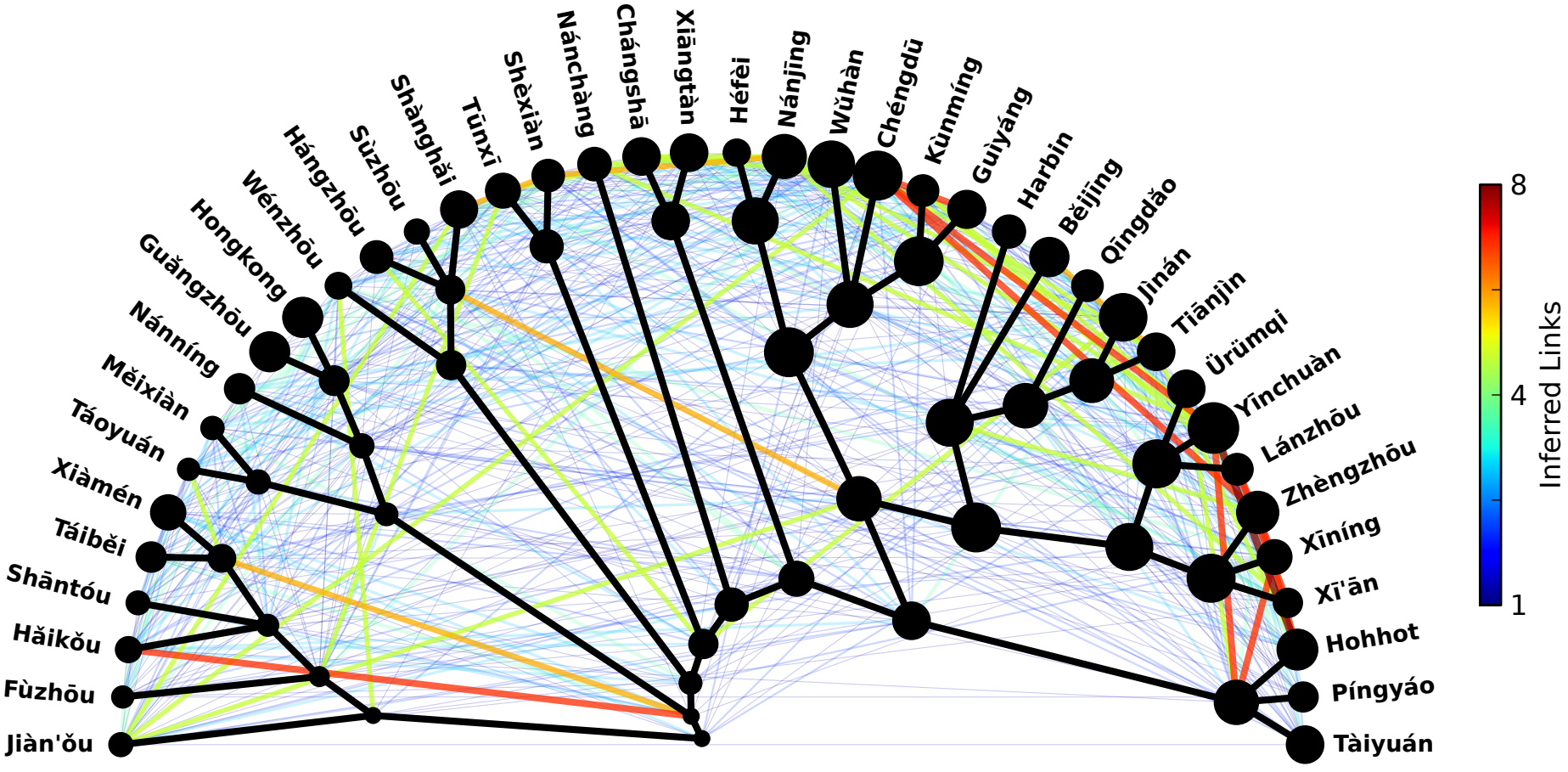
Reference tree of the Chinese dialects

Phylogeny-Based Borrowing Detection (PhyBo)



MLN analysis, no borrowing allowed

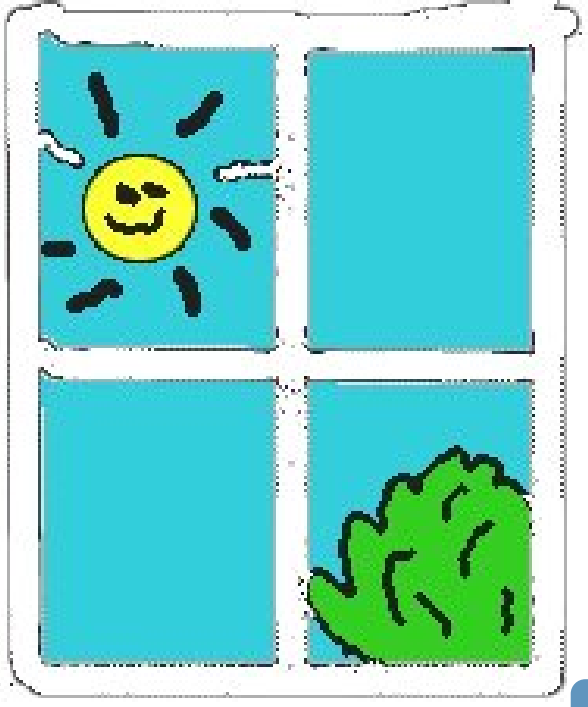
Phylogeny-Based Borrowing Detection (PhyBo)



MLN analysis, best fit of borrowing and inheritance

Examples

Examples in form of an IPython Notebook along with a HowTo-script will be uploaded to `http://lingulist.de/talks.php`.



Outlook

We need to improve both the methods we use and the way we present them to the linguistic world. The following are just a few pending problems:

We need to improve both the methods we use and the way we present them to the linguistic world. The following are just a few pending problems:

- make it easier for non-programmers to access LingPy (a GUI, or some simple terminal-based framework, a full tutorial)

We need to improve both the methods we use and the way we present them to the linguistic world. The following are just a few pending problems:

- make it easier for non-programmers to access LingPy (a GUI, or some simple terminal-based framework, a full tutorial)
- make the results of LingPy analyses more transparent (plots, findings, predictions)

We need to improve both the methods we use and the way we present them to the linguistic world. The following are just a few pending problems:

- make it easier for non-programmers to access LingPy (a GUI, or some simple terminal-based framework, a full tutorial)
- make the results of LingPy analyses more transparent (plots, findings, predictions)
- conduct rigorous testing of LingPy analyses (benchmarking, test parameter settings)

We need to improve both the methods we use and the way we present them to the linguistic world. The following are just a few pending problems:

- make it easier for non-programmers to access LingPy (a GUI, or some simple terminal-based framework, a full tutorial)
- make the results of LingPy analyses more transparent (plots, findings, predictions)
- conduct rigorous testing of LingPy analyses (benchmarking, test parameter settings)
- develop the methods further and include further methods (borrowing detection, automatic linguistic reconstruction, morpheme detection)

That’s all for now...

Thank You!