

Multiple Sequence Alignment in Historical Linguistics

A Sound Class Based Approach

Johann-Mattis List*

* Institute for Romance Languages and Literature
Heinrich Heine University Düsseldorf

2011/01/06

Structure of the Talk

Introduction

- Sequences

- Alignments

Automatic Alignment Analyses

- Pairwise Sequence Alignment

- Multiple Sequence Alignment

Alignments in Historical Linguistics

- Similarity

- Sound Classes

LingPy

- Main Ideas

- Working Principle

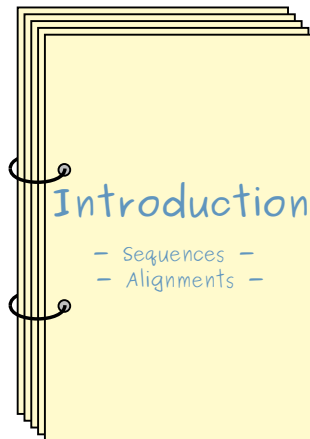
- Scoring

Performance of the Method

- Usage Example

- TPPSR

Introduction



Sequences

Sets

- ▶ Sets are **unordered** lists of **unique** objects.
- ▶ Sets are **compared** by comparing the **objects** of different sets.

Sequences

- ▶ Sequences are **ordered** lists of **non-unique** objects.
- ▶ Sequences are **compared** by comparing both the **objects** (segments) and the **structure** of different sequences.

Alignments

Sequence Alignment

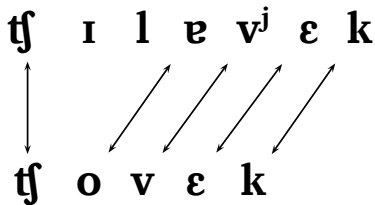
In alignment analyses, the corresponding **segments** of two or more **sequences** are ordered in such a way that they are **set against each other**. Segments which do not correspond to any other segments are marked by **gaps** (-). In this way, both, the structure and the segments of two or more sequences can be compared.

Alignments

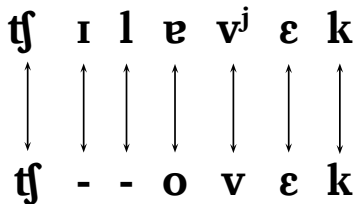
ʈʂ ɪ l ɐ vʲ ɛ k

ʈʂ o v ɛ k

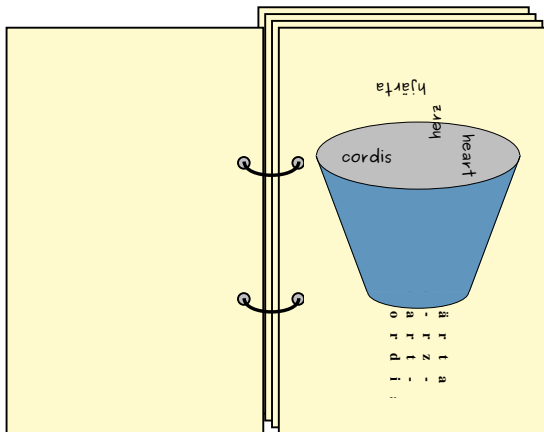
Alignments



Alignments



Automatic Alignment Analyses



Pairwise Sequence Alignment

- ▶ Create a **matrix** which confronts **all segments** of two **sequences**, either with **each other**, or with **gaps**.
- ▶ Seek the **path** through the matrix which is of the **lowest cost** (or the **highest score**).
- ▶ Calculate the cost (or the score) **cumulatively** by scoring the **matching** of segments with segments and with gaps by means of a specific **scoring function**.

Pairwise Sequence Alignment

T	E	S	T	-	-	-	-
-	-	-	-	T	E	S	T

8

Pairwise Sequence Alignment

T	E	S	T	-	-	-
-	-	-	T	E	S	T

6

Pairwise Sequence Alignment

T	E	S	-	T	-	-	-
-	-	-	T	-	E	S	T

8

Pairwise Sequence Alignment

T	E	S	T	-	-	-
-	-	T	-	E	S	T

7

Pairwise Sequence Alignment

T	E	-	S	T	-	-	-
-	-	T	-	-	E	S	T

8

Pairwise Sequence Alignment

T	E	S	T	-	-	-
-	T	-	-	E	S	T

7

Pairwise Sequence Alignment

T	-	E	S	T	-	-	-
-	T	-	-	-	E	S	T

8

Pairwise Sequence Alignment

T	E	S	T	-	-	-
T	-	-	-	E	S	T

6

Pairwise Sequence Alignment

T	E	S	T	-	-
T	-	-	E	S	T

5

Pairwise Sequence Alignment

T	E	S	-	T	-	-
T	-	-	E	-	S	T

6

Pairwise Sequence Alignment

T	E	S	T	-	-
T	-	E	-	S	T

5

Pairwise Sequence Alignment

T	E	-	S	T	-	-
T	-	E	-	-	S	T

6

Pairwise Sequence Alignment

T	E	S	T	-	-
T	E	-	-	S	T

4

Pairwise Sequence Alignment

T	E	S	T	-
T	E	-	S	T

3

Pairwise Sequence Alignment

T	E	-	S	T	-
T	E	S	-	-	T

4

Pairwise Sequence Alignment

T	E	S	T	-
T	E	S	-	T

2

Pairwise Sequence Alignment

T	E	S	T
T	E	S	T

0

Multiple Sequence Alignments

Guide Tree Heuristics

- ▶ Due to computational restrictions, multiple sequence alignment (MSA) is based on **heuristics**.
- ▶ Heuristics based on **guide-trees** are the most common ones used in computational biology.
- ▶ Based on **pairwise alignment scores**, a **guide-tree** is reconstructed, and the sequences are **stepwise** added to the MSA along it (Feng & Dolittle 1987).

Multiple Sequence Alignment

čelovek
—
“human”
Russian

člověk
—
“human”
Czech

człowiek
—
“human”
Polish

čovek
—
“human”
Bulgarian

Multiple Sequence Alignment

ѣлевѣк

Russian

ѣlovѣк

Czech

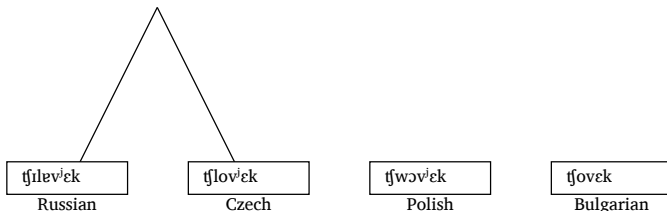
ѣwɔvѣк

Polish

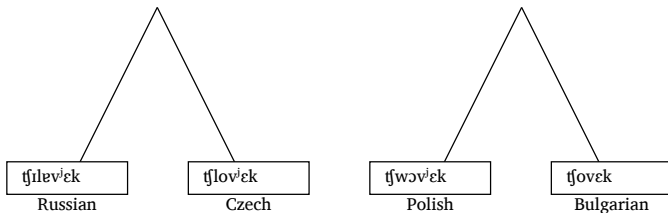
ѣlovѣк

Bulgarian

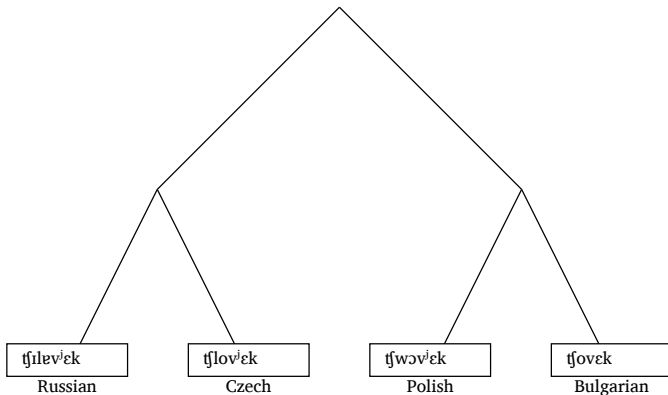
Multiple Sequence Alignment



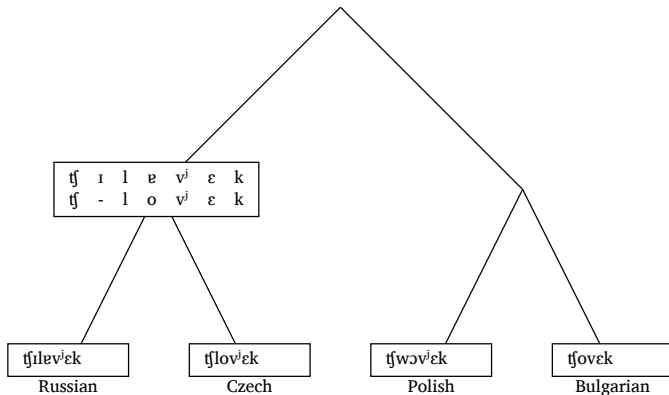
Multiple Sequence Alignment



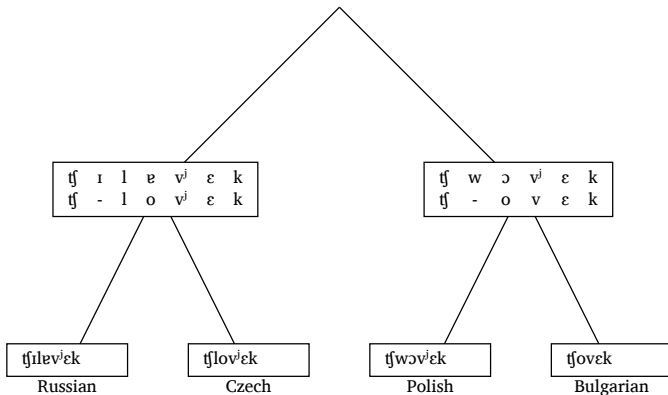
Multiple Sequence Alignment



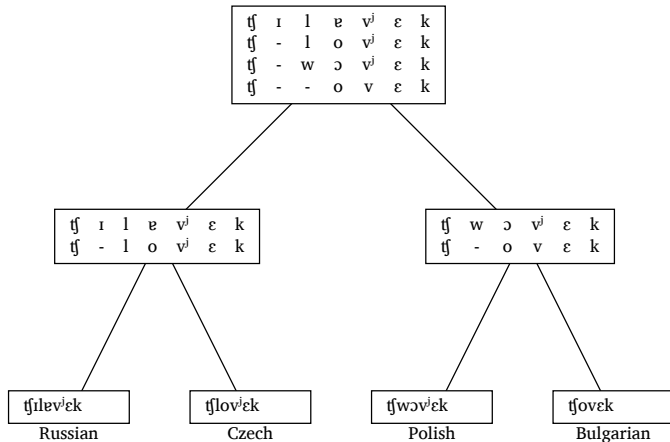
Multiple Sequence Alignment



Multiple Sequence Alignment



Multiple Sequence Alignment

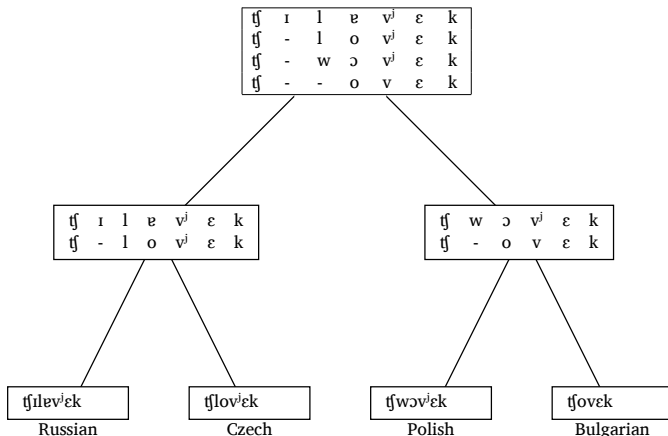


Multiple Sequence Alignment

Profiles

- ▶ The guide-tree heuristic can be enhanced by the application of **profiles**.
- ▶ A profile consists of the **relative frequency** of all segments of an MSA in all its positions, thus, a profile represents an MSA as a **sequence of vectors**.
- ▶ Aligning profiles to profiles instead of aligning two representative sequences of two given MSA yields better results, since more information can be taken into account.

Multiple Sequence Alignment



Multiple Sequence Alignment

tʃ	ɪ	l	ɐ	vʲ	ɛ	k
tʃ	-	l	o	vʲ	ɛ	k
tʃ	-	w	ɔ	vʲ	ɛ	k
tʃ	-	-	o	v	ɛ	k

Multiple Sequence Alignment

tf	i	l	e	v ^j	ε	k
tf	-	l	o	v ^j	ε	k
tf	-	w	ɔ	v ^j	ε	k
tf	-	-	o	v	ε	k

tf	1.0						
----	-----	--	--	--	--	--	--

Multiple Sequence Alignment

tʃ	ɪ	l	ɐ	vʲ	ɛ	k
tʃ	-	l	o	vʲ	ɛ	k
tʃ	-	w	ɔ	vʲ	ɛ	k
tʃ	-	-	o	v	ɛ	k

tʃ	1.0						
ɪ		.25					
-		.75					

Multiple Sequence Alignment

tʃ	ɪ	l	ɐ	vʲ	ɛ	k
tʃ	-	l	o	vʲ	ɛ	k
tʃ	-	w	ɔ	vʲ	ɛ	k
tʃ	-	-	o	v	ɛ	k

tʃ	1.0						
ɪ		.25					
-		.75	.25				
l			.5				
w			.25				

Multiple Sequence Alignment

tʃ	ɪ	l	e	vʲ	ɛ	k
tʃ	-	l	o	vʲ	ɛ	k
tʃ	-	w	ɔ	vʲ	ɛ	k
tʃ	-	-	o	v	ɛ	k

tʃ	1.0						
ɪ		.25					
-		.75	.25				
l			.5				
w			.25				
o				.5			
ɔ				.25			
e				.25			

Multiple Sequence Alignment

tʃ	ɪ	l	e	vʲ	ɛ	k
tʃ	-	l	o	vʲ	ɛ	k
tʃ	-	w	ɔ	vʲ	ɛ	k
tʃ	-	-	o	v	ɛ	k

tʃ	1.0						
ɪ		.25					
-		.75	.25				
l			.5				
w			.25				
o				.5			
ɔ				.25			
e				.25			
vʲ					.75		
v					.25		

Multiple Sequence Alignment

tʃ	ɪ	l	ɐ	vʲ	ɛ	k
tʃ	-	l	o	vʲ	ɛ	k
tʃ	-	w	ɔ	vʲ	ɛ	k
tʃ	-	-	o	v	ɛ	k

tʃ	1.0						
ɪ		.25					
-		.75	.25				
l			.5				
w			.25				
o				.5			
ɔ				.25			
ɐ				.25			
vʲ					.75		
v					.25		
ɛ						1.0	

Multiple Sequence Alignment

tʃ	ɪ	l	ɐ	vʲ	ɛ	k
tʃ	-	l	o	vʲ	ɛ	k
tʃ	-	w	ɔ	vʲ	ɛ	k
tʃ	-	-	o	v	ɛ	k

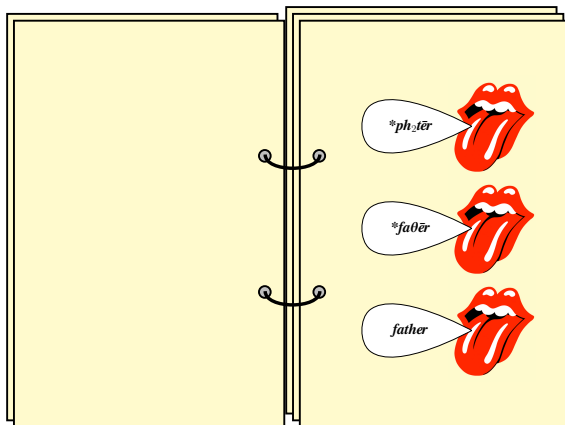
tʃ	1.0						
ɪ		.25					
-		.75	.25				
l			.5				
w			.25				
o				.5			
ɔ				.25			
ɐ				.25			
vʲ					.75		
v					.25		
ɛ						1.0	
k							1.0

Multiple Sequence Alignment

tʃ	ɪ	l	e	vʲ	ɛ	k
tʃ	-	l	o	vʲ	ɛ	k
tʃ	-	w	ɔ	vʲ	ɛ	k
tʃ	-	-	o	v	ɛ	k

tʃ	1.0						
ɪ		.25					
-		.75	.25				
l			.5				
w			.25				
o				.5			
ɔ				.25			
e				.25			
vʲ					.75		
v					.25		
ɛ						1.0	
k							1.0

Alignments in Historical Linguistics



Similarity

Synchronic Similarity

Sounds in different languages are judged to be **similar**, if they show **resemblances** regarding **the way they are produced or perceived**.

Diachronic Similarity

Sounds in different languages are judged to be **similar**, if they **go back to a common ancestor**.

Similarity

Language	Word	Meaning
Mandarin	ma ⁵⁵ ma ³	“mother”
German	mama	“mother”
Russian	tak	“in this way”
German	t ^h a:k	“day”

Similarity

Language	Word	Meaning
German	ts ^h a:n	“tooth”
English	tu:θ	“tooth”
Italian	dentɛ	“tooth”
French	dã	“tooth”

Similarity

German

ts^h a: n

English

t ʊ: θ

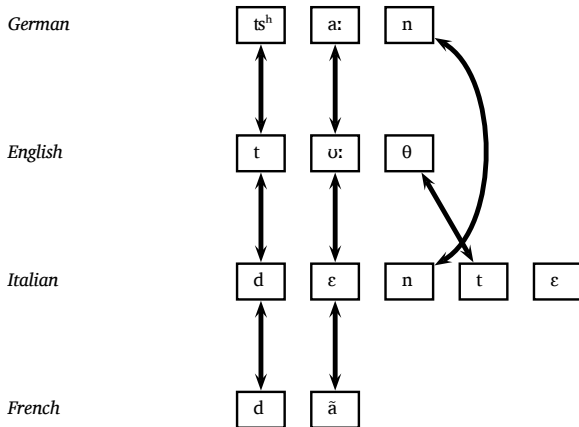
Italian

d ε n t ε

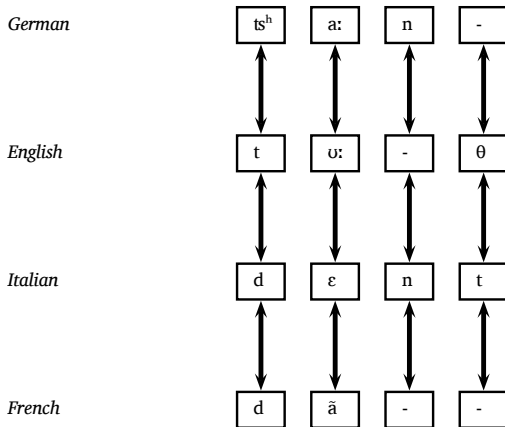
French

d ã

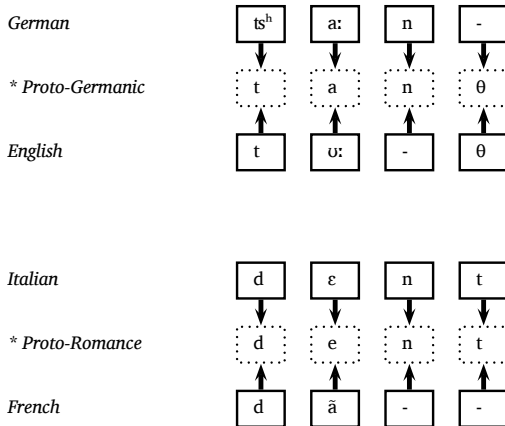
Similarity



Similarity



Similarity



Similarity

* *Proto-Germanic*

t	a	n	θ
---	---	---	---

* *Proto-Romance*

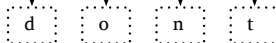
d	e	n	t
---	---	---	---

Similarity

* *Proto-Germanic*



** *Proto-Indo-European*



* *Proto-Romance*



Similarity

**** Proto-Indo-European**



Sound Classes

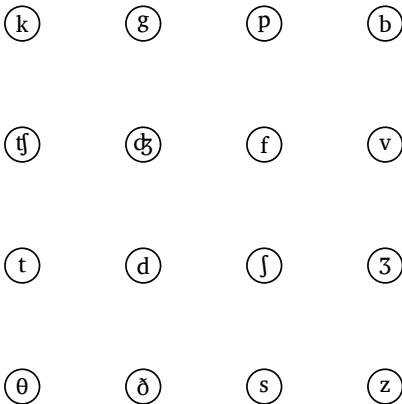
Correspondence Classes

In sound class approaches, sounds are “divided into several types and thereby distinguished in such a way that phonetic correspondences inside a ‘type’ are more regular than those between different ‘types’” (Dolgopolsky 1986: 35).

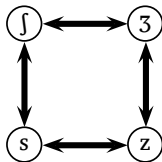
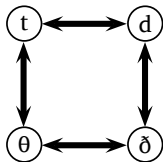
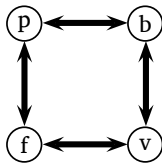
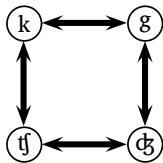
Diachronic Similarity

Similarity is not based on **synchronic resemblances** of sounds but on **class-membership**: two sounds, how dissimilar they may be from a synchronic perspective, may still belong to the same class. Class membership indicates that the **probability that sounds occur in a correspondence relationship** in genetically related languages is considerably high.

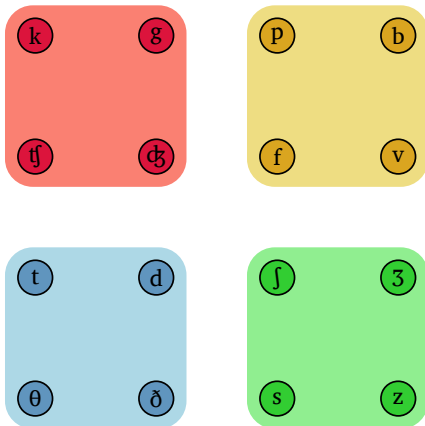
Sound Classes



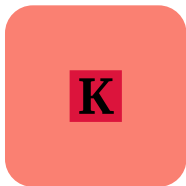
Sound Classes



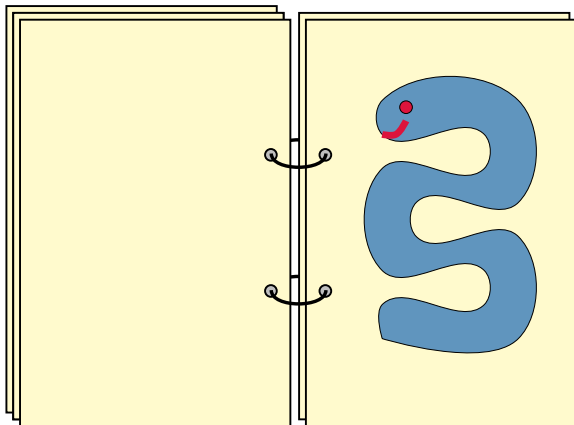
Sound Classes



Sound Classes



LingPy



LingPy

A Python Library for Sequence Alignment

LingPy (www.lingulist.de/lingpy) is a suite of open source Python modules for sequence comparison, and distance analyses in quantitative historical linguistics. The library allows to carry out both pairwise and multiple alignments of strings encoded in IPA or X-Sampa, using different methods and algorithms, such as global (Needleman & Wunsch 1970) and local (Smith & Waterman 1981) pairwise alignments, multiple alignments based on guide trees (Feng & Doolittle 1987), profiles (Thompson et al. 1994), or iteration (Barton & Sternberg 1987).

Main Ideas

Alignment of Sound Class Sequences

In contrast to previous approaches, which base the alignment on the sequences as **they are given from the input**, within the sound class approach, the input strings are first **converted to sound classes** before they are aligned.

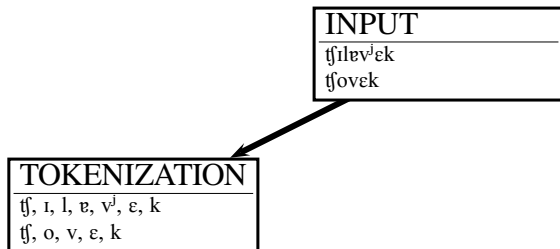
Transitions Between Sound Classes

In contrast to previous sound class approaches (cf. e.g. Turchin et al. 2010), which do not allow for **transitions between sound classes**, this approach is based on a **specific scoring function**, which defines (diachronic) similarity among different sound classes.

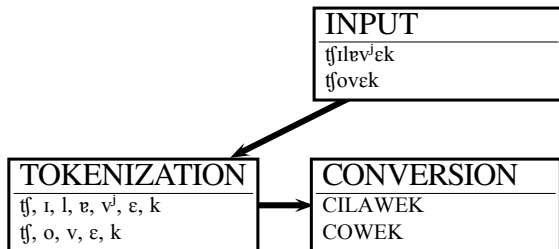
Working Principle

INPUT
ʈʂɪlɐvʲɛk
ʈʂovɛk

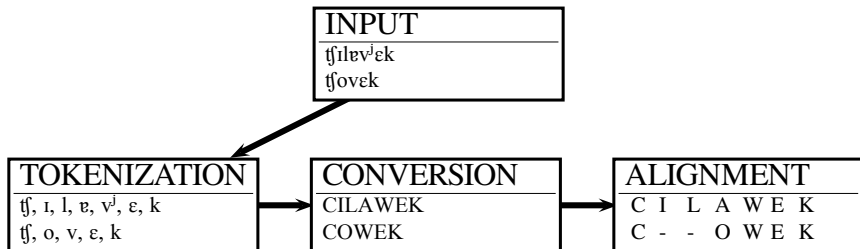
Working Principle



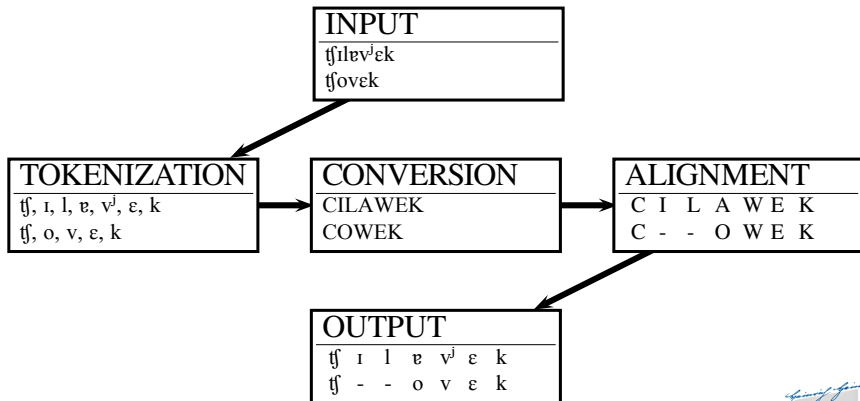
Working Principle



Working Principle



Working Principle



Scoring

Directionality of Sound Changes

One crucial characteristic of certain well-known sound changes is their **directionality**, i.e. if certain sounds change, this change will go into a certain direction and the **reverse change can rarely be attested**.

Directionality and Sound Correspondences

While the nature of certain sound changes **may be directional**, sound correspondences do not directly reflect this directionality, and neither do scoring functions for sequence alignments, since these are **not directional *per definitionem***, since the distance or similarity between two segments is always the same, regardless from which segment we start to compare.

Scoring

Reflecting Directionality in Undirected Networks

In this approach, the directionality of certain sound changes is accounted for by creating a **non-metric scoring function**.

While in a metric scoring function the distance between two segments A and B would depend on the distance of A and B to a third segment C in such a way that, according to the **triangle inequality** the distance from A to B could **not exceed the sum of the distances from A to C and from B to C** , this does not hold for the probability of those sound correspondences, which occur as a product of directional sound change.

Scoring

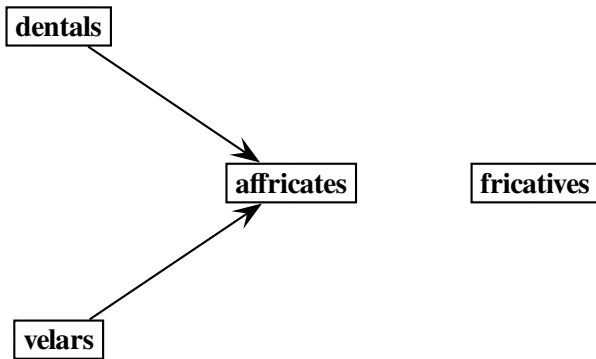
dentals

affricates

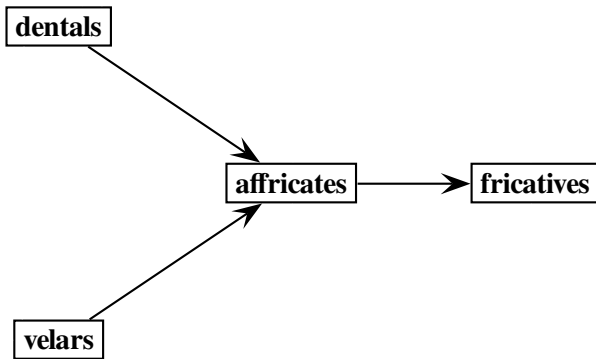
fricatives

velars

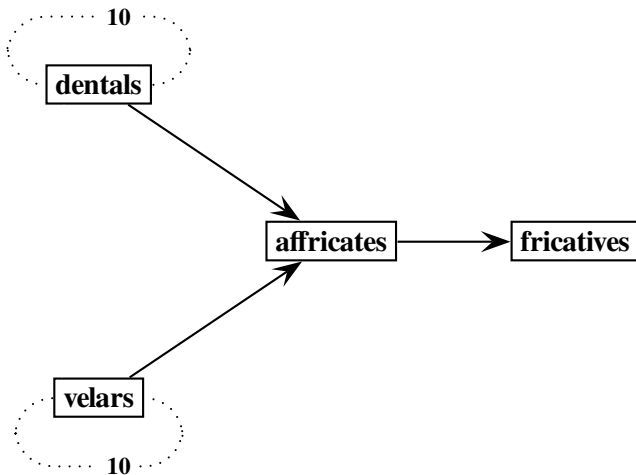
Scoring



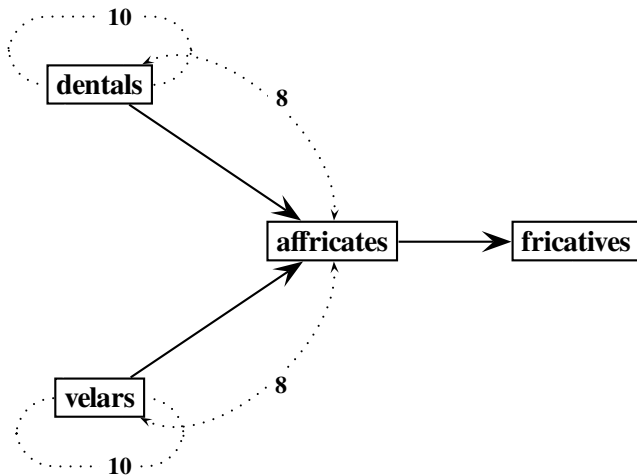
Scoring



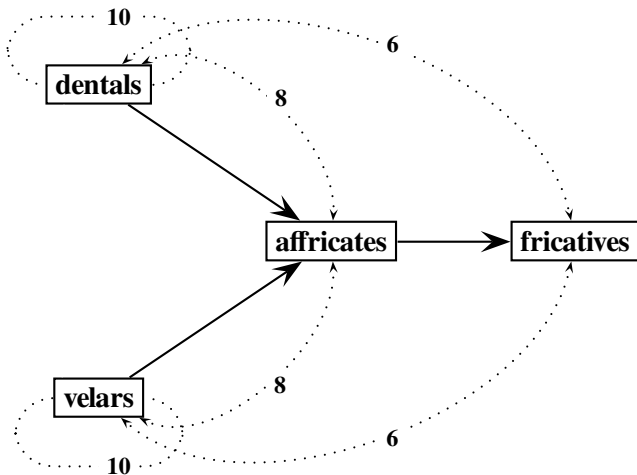
Scoring



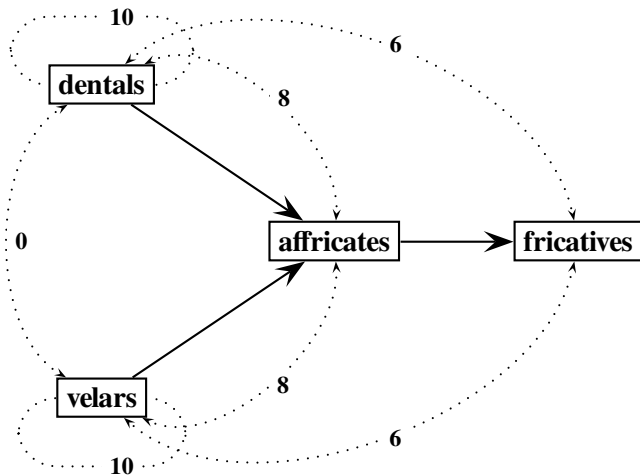
Scoring



Scoring



Scoring



Performance of the Method



Usage Example

Usage Example

```
>>> from lingpy.compare.seqcom import Multiple
```

Usage Example

```
>>> from lingpy.compare.seqcom import Multiple  
>>> mult = Multiple(['ʈwovɨɛk', 'ʈovɛk',\  
... 'ʈlovɨɛk', 'ʈilevɨɛk'])
```

Usage Example

```
>>> from lingpy.compare.seqcom import Multiple
>>> mult = Multiple(['ʈwovɨɛk', 'ʈovɛk', \
... 'ʈlovɨɛk', 'ʈilevɨɛk'])
>>> print ', '.join(mult.ipt_seqs)
```

Usage Example

```
>>> from lingpy.compare.seqcom import Multiple
>>> mult = Multiple(['ʈwovɛk', 'ʈovɛk', \
... 'ʈlovɛk', 'ʈilevɛk'])
>>> print ', '.join(mult.ipt_seqs)
ʈwovɛk, ʈovɛk, ʈlovɛk, ʈilevɛk
```

Usage Example

```
>>> from lingpy.compare.seqcom import Multiple
>>> mult = Multiple(['ʈwovʲɛk', 'ʈovɛk', \
... 'ʈlovʲɛk', 'ʈilevʲɛk'])
>>> print ', '.join(mult.ipt_seqs)
ʈwovʲɛk, ʈovɛk, ʈlovʲɛk, ʈilevʲɛk
>>> mult.prog_align(method='sca', mode='profile')
```

Usage Example

```
>>> from lingpy.compare.seqcom import Multiple
>>> mult = Multiple(['ʈwovʲɛk', 'ʈovɛk', \
... 'ʈlovʲɛk', 'ʈilevʲɛk'])
>>> print ', '.join(mult.ipt_seqs)
ʈwovʲɛk, ʈovɛk, ʈlovʲɛk, ʈilevʲɛk
>>> mult.prog_align(method='sca', mode='profile')
```

ʈ	-	w	ɔ	vʲ	ɛ	k
ʈ	-	-	o	v	ɛ	k
ʈ	-	l	o	vʲ	ɛ	k
ʈ	ɪ	l	e	vʲ	ɛ	k

Usage Example

```
>>> from lingpy.compare.seqcom import Multiple
>>> mult = Multiple(['ʈwovʲɛk', 'ʈovɛk', \
... 'ʈlovʲɛk', 'ʈilevʲɛk'])
>>> print ', '.join(mult.ipt_seqs)
ʈwovʲɛk, ʈovɛk, ʈlovʲɛk, ʈilevʲɛk
>>> mult.prog_align(method='sca', mode='profile')
ʈ      -      w      ɔ      vʲ      ɛ      k
ʈ      -      -      o      v      ɛ      k
ʈ      -      l      o      vʲ      ɛ      k
ʈ      ɪ      l      ɐ      vʲ      ɛ      k
>>> mult.show_guide_tree()
```

Usage Example

```
>>> from lingpy.compare.seqcom import Multiple
>>> mult = Multiple(['tʷovʲɛk', 'tʷovɛk', \
... 'tʷlovʲɛk', 'tʷilevʲɛk'])
>>> print ', '.join(mult.ipt_seqs)
tʷovʲɛk, tʷovɛk, tʷlovʲɛk, tʷilevʲɛk
>>> mult.prog_align(method='sca', mode='profile')
tʷ   -   w   ɔ   vʲ   ɛ   k
tʷ   -   -   o   v   ɛ   k
tʷ   -   l   o   vʲ   ɛ   k
tʷ   ɪ   l   ɐ   vʲ   ɛ   k
>>> mult.show_guide_tree()
                        /-0:tʷovʲɛk
                /-----|
            |               \-1:tʷovɛk
-----|
            |               /-3:tʷlovʲɛk
            \-----|
                        \-2:tʷilevʲɛk
```

Usage Example

```
>>> from lingpy.compare.seqcom import Multiple
>>> mult = Multiple(['ʈwovʲɛk', 'ʈovɛk', \
... 'ʈlovʲɛk', 'ʈilevʲɛk'])
>>> print ', '.join(mult.ipt_seqs)
ʈwovʲɛk, ʈovɛk, ʈlovʲɛk, ʈilevʲɛk
>>> mult.prog_align(method='sca', mode='profile')
ʈ   -   w   ɔ   vʲ   ɛ   k
ʈ   -   -   o   v   ɛ   k
ʈ   -   l   o   vʲ   ɛ   k
ʈ   ɪ   l   e   vʲ   ɛ   k
>>> mult.show_guide_tree()
          /-0:ʈwovʲɛk
        /-----|
       |           \-1:ʈovɛk
-----|
       |           /-3:ʈlovʲɛk
       |           \-----|
                \-2:ʈilevʲɛk
>>> print ', '.join([seq.cls_str for seq in \
... mult.lingpy_seqs])
```

Usage Example

```
>>> from lingpy.compare.seqcom import Multiple
>>> mult = Multiple(['řwovjek', 'řovek', \
... 'řlovjek', 'řilevjek'])
>>> print ', '.join(mult.ipt_seqs)
řwovjek, řovek, řlovjek, řilevjek
>>> mult.prog_align(method='sca', mode='profile')
ř    -    w    ɔ    vʲ    ε    k
ř    -    -    o    v    ε    k
ř    -    l    o    vʲ    ε    k
ř    r    l    e    vʲ    ε    k
>>> mult.show_guide_tree()
                /-0:řwovjek
            /-----|
        |               \-1:řovek
-----|
        |               /-3:řlovjek
        \-----|
                \-2:řilevjek
>>> print ', '.join([seq.cls_str for seq in \
... mult.lingpy_seqs])
CWOWEK, COWEK, CLOWEK, CILAWEK
```

Usage Example

Usage Example

```
>>> mult.flat_cluster(0.3,method='sca')
```

Usage Example

```
>>> mult.flat_cluster(0.3,method='sca')  
[1, 1, 1, 1]
```

Usage Example

```
>>> mult.flat_cluster(0.3,method='sca')  
[1, 1, 1, 1]  
>>> mult.prog_align(method='sca',mode='profile')\  
... # profile-based alignment
```


Usage Example

```
>>> mult.flat_cluster(0.3,method='sca')  
[1, 1, 1, 1]  
>>> mult.prog_align(method='sca',mode='profile')\  
... # profile-based alignment  
tʃ    -    w    ɔ    vʲ    ɛ    k  
tʃ    -    -    o    v    ɛ    k  
tʃ    -    l    o    vʲ    ɛ    k  
tʃ    ɪ    l    e    vʲ    ɛ    k
```

Usage Example

```
>>> mult.flat_cluster(0.3,method='sca')  
[1, 1, 1, 1]  
>>> mult.prog_align(method='sca',mode='profile')\  
... # profile-based alignment  
tʃ    -    w    ɔ    vʲ    ɛ    k  
tʃ    -    -    o    v    ɛ    k  
tʃ    -    l    o    vʲ    ɛ    k  
tʃ    ɪ    l    ɐ    vʲ    ɛ    k  
>>> mult.sum_of_pairs()
```

Usage Example

```
>>> mult.flat_cluster(0.3,method='sca')  
[1, 1, 1, 1]  
>>> mult.prog_align(method='sca',mode='profile')\  
... # profile-based alignment  
tʃ    -    w    ɔ    vʲ    ɛ    k  
tʃ    -    -    o    v    ɛ    k  
tʃ    -    l    o    vʲ    ɛ    k  
tʃ    ɪ    l    e    vʲ    ɛ    k  
>>> mult.sum_of_pairs()  
39.666666666666664
```

Usage Example

```
>>> mult.flat_cluster(0.3,method='sca')
[1, 1, 1, 1]
>>> mult.prog_align(method='sca',mode='profile') \
... # profile-based alignment
tʃ    -    w    ɔ    vʲ    ɛ    k
tʃ    -    -    o    v    ɛ    k
tʃ    -    l    o    vʲ    ɛ    k
tʃ    ɪ    l    e    vʲ    ɛ    k
>>> mult.sum_of_pairs()
39.666666666666664
>>> mult.prog_align(method='sca',mode='fd') \
... # simple guide-tree alignment
```

Usage Example

```
>>> mult.flat_cluster(0.3,method='sca')
[1, 1, 1, 1]
>>> mult.prog_align(method='sca',mode='profile')\
... # profile-based alignment
tf      -      w      ɔ      vʲ      ε      k
tf      -      -      o      v      ε      k
tf      -      l      o      vʲ      ε      k
tf      ɪ      l      e      vʲ      ε      k
>>> mult.sum_of_pairs()
39.666666666666664
>>> mult.prog_align(method='sca',mode='fd') \
... # simple guide-tree alignment
tf      w      -      ɔ      vʲ      ε      k
tf      -      -      o      v      ε      k
tf      -      l      o      vʲ      ε      k
tf      ɪ      l      e      vʲ      ε      k
```

Usage Example

```
>>> mult.flat_cluster(0.3,method='sca')
[1, 1, 1, 1]
>>> mult.prog_align(method='sca',mode='profile')\
... # profile-based alignment
tf    -    w    ɔ    vʲ    ε    k
tf    -    -    o    v    ε    k
tf    -    l    o    vʲ    ε    k
tf    ɪ    l    e    vʲ    ε    k
>>> mult.sum_of_pairs()
39.666666666666664
>>> mult.prog_align(method='sca',mode='fd') \
... # simple guide-tree alignment
tf    w    -    ɔ    vʲ    ε    k
tf    -    -    o    v    ε    k
tf    -    l    o    vʲ    ε    k
tf    ɪ    l    e    vʲ    ε    k
>>> mult.iterate()
```

Usage Example

```
>>> mult.flat_cluster(0.3,method='sca')
[1, 1, 1, 1]
>>> mult.prog_align(method='sca',mode='profile')\
... # profile-based alignment
tf      -      w      ɔ      vʲ      ε      k
tf      -      -      o      v      ε      k
tf      -      l      o      vʲ      ε      k
tf      ɪ      l      e      vʲ      ε      k
>>> mult.sum_of_pairs()
39.666666666666664
>>> mult.prog_align(method='sca',mode='fd') \
... # simple guide-tree alignment
tf      w      -      ɔ      vʲ      ε      k
tf      -      -      o      v      ε      k
tf      -      l      o      vʲ      ε      k
tf      ɪ      l      e      vʲ      ε      k
>>> mult.iterate()
Old SoP score: 37.8333333333
New SoP score: 39.6666666667
```

Usage Example

```
>>> mult.flat_cluster(0.3,method='sca')
[1, 1, 1, 1]
>>> mult.prog_align(method='sca',mode='profile')\
... # profile-based alignment
tf  -   w   ɔ   vʲ   ε   k
tf  -   -   o   v   ε   k
tf  -   l   o   vʲ   ε   k
tf  ɪ   l   e   vʲ   ε   k
>>> mult.sum_of_pairs()
39.666666666666664
>>> mult.prog_align(method='sca',mode='fd') \
... # simple guide-tree alignment
tf  w   -   ɔ   vʲ   ε   k
tf  -   -   o   v   ε   k
tf  -   l   o   vʲ   ε   k
tf  ɪ   l   e   vʲ   ε   k
>>> mult.iterate()
Old SoP score: 37.8333333333
New SoP score: 39.6666666667
tf  -   w   ɔ   vʲ   ε   k
tf  -   -   o   v   ε   k
tf  -   l   o   vʲ   ε   k
tf  ɪ   l   e   vʲ   ε   k
```


TPPSR

IPA-Encoding of the TPPSR

The *Tableaux phonétiques des patois suisses romand* (TPPSR, Gauchat et al. 1925) is a collection of phonetic dialect data, which was digitized in an earlier research project of the Institute for Romance Languages and Literature (Heinrich Heine University Düsseldorf). The original data was converted to IPA in order make it suitable for alignment analyses using the LingPy library. The dataset consists of 480 charts (480 words and phrases) which contain phonetic information for 62 dialect points.

Analysis within LingPy

The analysis within LingPy is done via a simple terminal-based interface which takes text-files as input and outputs the results of the alignment analyses as text-files.

TPPSR

```
tppsr
69, sont tout près, pressu
2   sɔ̃.tɔ.pri:
3   i.sɔ̃.tɔ.ʤe:
5   ei.səɔ̃.tɔ.prei
8   sɔ̃.pre
11  sɔ̃.tɔ.pru:tɔ
18  sɔ̃.pre
19  sɔ̃.tɔ.pre
30  ʃɔ̃n.pre
31  ʃɔ̃n.tɔ.prei
34  i.sɔ̃.tɔ.pre
54  ɛ.sɔ̃.tɔ.pre
55  prɛj
56  a.sãɔ̃.tɔ.d.ko:t
57  sɔ̃.tɔ.pre:
58  a.sɔ̃.tɔ.prenj
```

TPPSR

```

tppsr
69, sont tout près, pressu
2 1 - s õ - t o p r i: - -
3 2 i.sõ.to.ʒe:
5 1 ei s əõ - t o p r ei - -
8 1 - s õ - - - p r e - -
11 1 - s õ - t o p r u: ts o
18 1 - s õ - - - p r e - -
19 1 - s õ - t o p r e - -
30 1 - ʃ õ n - - p r e - -
31 1 - ʃ õ n t o p r ei - -
34 1 i s õ - t o p r e - -
54 1 e s õ - t o p r e - -
55 1 - - - - - p r e j -
56 3 a.sãõ.to.d.ko:t
57 1 - s õ - t o p r e: - -
58 1 a s õ - t o p r e ŋ -

```

TPPSR

tppsr

69, sont tout près, pressu

2	1	-	s	õ	-	t	o	p	r	i:	-	-
3	2	i.sõ.to.ðe:										
5	1	ei	s	əõ	-	t	o	p	r	ei	-	-
8	1	-	s	õ	-	-	-	p	r	e	-	-
11	1	-	s	õ	-	t	o	p	r	u:	ts	o
18	1	-	s	õ	-	-	-	p	r	e	-	-
19	1	-	s	õ	-	t	o	p	r	e	-	-
30	1	-	ʃ	õ	n	-	-	p	r	e	-	-
31	1	-	ʃ	õ	n	t	o	p	r	ei	-	-
34	1	i	s	õ	-	t	o	p	r	e	-	-
54	1	e	s	õ	-	t	o	p	r	e	-	-
55	1	-	-	-	-	-	-	p	r	e	j	-
56	3	a.sãõ.to.d.ko:t										
57	1	-	s	õ	-	t	o	p	r	e:	-	-
58	1	a	s	õ	-	t	o	p	r	e	ŋ	-

Taxon-ID

TPPSR

```

tppsr
69, sont tout près, pressu
2 1 - s õ - t o p r i: - -
3 2 i.sõ.to.ʒe:
5 1 ei s əõ - t o p r ei - -
8 1 - s õ - - - p r e - -
11 1 - s õ - t o p r u: ts o
18 1 - s õ - - - p r e - -
19 1 - s õ - t o p r e - -
30 1 - ʃ õ n - - p r e - -
31 1 - ʃ õ n t o p r ei - -
34 1 i s õ - t o p r e - -
54 1 e s õ - t o p r e - -
55 1 - - - - - p r e j -
56 3 a.sãõ.to.d.ko:t
57 1 - s õ - t o p r e: - -
58 1 a s õ - t o p r e ŋ -

```

Cluster-ID

TPPSR

```

tppsr
69, sont tout près, pressu
2 1 - s õ - t o p r i: - -
3 2 i.sõ.to.ɕe: ← Singleton
5 1 ei s əõ - t o p r ei - -
8 1 - s õ - - - p r e - -
11 1 - s õ - t o p r u: ts o
18 1 - s õ - - - p r e - -
19 1 - s õ - t o p r e - -
30 1 - ʃ õ n - - p r e - -
31 1 - ʃ õ n t o p r ei - -
34 1 i s õ - t o p r e - -
54 1 e s õ - t o p r e - -
55 1 - - - - - p r e j -
56 3 a.sãõ.to.d.ko:t ← Singleton
57 1 - s õ - t o p r e: - -
58 1 a s õ - t o p r e ŋ -

```

TPPSR

tppsr

69, sont tout près, pressu

2	1	-	s	õ	-	t	o	p	r	i:	-	-
5	1	ei	s	əõ	-	t	o	p	r	ei	-	-
8	1	-	s	õ	-	-	-	p	r	e	-	-
11	1	-	s	õ	-	t	o	p	r	u:	ts	o
18	1	-	s	õ	-	-	-	p	r	e	-	-
19	1	-	s	õ	-	t	o	p	r	e	-	-
30	1	-	ʃ	õ	n	-	-	p	r	e	-	-
31	1	-	ʃ	õ	n	t	o	p	r	ei	-	-
34	1	i	s	õ	-	t	o	p	r	e	-	-
54	1	e	s	õ	-	t	o	p	r	e	-	-
55	1	-	-	-	-	-	-	p	r	e	j	-
57	1	-	s	õ	-	t	o	p	r	e:	-	-
58	1	a	s	õ	-	t	o	p	r	e	ŋ	-

TPPSR

tppsr

69, sont tout près, pressu

2	1	-	s	õ	-	t	o	p	r	i:	-	-
5	1	ei	s	əõ	-	t	o	p	r	ei	-	-
8	1	-	s	õ	-	-	-	p	r	e	-	-
11	1	-	s	õ	-	t	o	p	r	u:	ts	o
18	1	-	s	õ	-	-	-	p	r	e	-	-
19	1	-	s	õ	-	t	o	p	r	e	-	-
30	1	-	ʃ	õ	n	-	-	p	r	e	-	-
31	1	-	ʃ	õ	n	t	o	p	r	ei	-	-
34	1	i	s	õ	-	t	o	p	r	e	-	-
54	1	e	s	õ	-	t	o	p	r	e	-	-
55	1	-	-	-	-	-	-	p	r	e	j	-
57	1	-	s	õ	-	t	o	p	r	e:	-	-
58	1	a	s	õ	-	t	o	p	r	e	ŋ	-

← Boring Site!

TPPSR

```

tppsr
69, sont tout près, pressu
2 1 - s õ - t o p r i: - -
5 1 ei s əõ - t o p r ei - -
8 1 - s õ - - - p r e - -
11 1 - s õ - t o p r u: ts o
18 1 - s õ - - - p r e - -
19 1 - s õ - t o p r e - -
30 1 - ʃ õ n - - p r e - -
31 1 - ʃ õ n t o p r ei - -
34 1 i s õ - t o p r e - -
54 1 e s õ - t o p r e - -
55 1 - - - - - p r e j -
57 1 - s õ - t o p r e: - -
58 1 a s õ - t o p r e ŋ -

```

← Interesting Site!

TPPSR

```
tppsr  
66,est étroite,stricta  
1   ε.etra:t  
2   ε.stræ:te  
3   ε.stre:ta  
5   ε.stras:ta  
8   ε.stra:et  
11  l.ε.stræ:tə  
19  l.ε.etra:tə  
30  l.εθ.εθreliti  
31  lj.ε.shri:ti  
34  st.e:tra:to  
55  ε.stræit  
56  ε.stra:æt  
57  ε.stret  
58  j.ε.stre:t
```

TPPSR

```

tppsr
66,est étroite,stricta
1 1 - ε - e t r a: t -
2 1 - ε - ε t r æ: t ε
3 1 - ε - ε t r e: t a
5 1 - ε - ε t r æ: t a
8 1 - ε - ε t r a:ε t -
11 1 l ε - ε t r æ: t ə
19 1 l ε - e t r a: t ə
30 1 l ε θ ε θ r ei t i
31 1 lj ε - ε h r i: t i
34 1 - ε t e: t r a: t o
55 1 - ε - ε t r æi t -
56 1 - ε - ε t r a:ə t -
57 1 - ε - ε t r ε t -
58 1 j ε - ε t r e: t -

```

TPPSR

```
tppsr
195,une feuille,folia
1   ðna.folj
2   na.folje
3   na.folj
5   una.foljə
8   ðna.fɔjə
11  ðna.foljə
19  na.føðə
30  folje
31  fɔle
34  na.fwolj
55  ɔn.fɔdj
56  ɛn.fuj
57  ɛn.fuj
58  ɛn.fæj
```

TPPSR

```

tppsr
195, une feuille, folia
1 1 ð n a f - ɔ lj -
2 1 - n a f - o lj ɛ
3 1 - n a f - ɔ lj -
5 1 u n a f - ɔ lj ə
8 1 ð n a f - ɔ j ə
11 1 ð n a f - ɔ lj ə
19 1 - n a f - ø ð ə
30 1 - - - f - o lj e
31 1 - - - f - ɔ ʔ e
34 1 - n a f w o lj -
55 1 ɔ n - f - ɔ dj -
56 1 ɛ n - f - u j -
57 1 ɛ n - f - u j -
58 1 ɛ n - f - œ j -

```

