Learning the building bits of inflectional paradigms

Sebastian Bank & Jochen Trommer

University of Leipzig

LAGB Annual Meeting 2013
August 28–31, 2013

Two Problems of Inflectional Analysis

- The Imperfect Distribution Problem
- The Subanalysis Problem

Ainu Verb Agreement (Tamura 2000)

	1sg	**1pl**	**2sg**	**2pl**	**3sg**	**3pl**	**–**
1sg			eci-	eci-	ku-	ku-	ku-
1pl			eci-	eci-	ci-	ci-	-as
2sg	en-	un-			e-	e-	e-
2pl	ecien-	eciun-			eci-	eci-	eci-
3sg	en-	un-	e-	eci-	∅-	∅-	∅-
3pl	en-	un-	e-	eci-	∅-	∅-	∅-

The Imperfect Distribution Problem

Perfect Distribution

	1sg	**1pl**	**2sg**	**2pl**	**3sg**	**3pl**	**–**
1sg			eci-	eci-	ku-	ku-	ku-
1pl			eci-	eci-	ci-	ci-	-as
2sg	en-	**un-**			e-	e-	e-
2pl	ecien-	eci-**un-**			eci-	eci-	eci-
3sg	en-	**un-**	e-	eci-	∅-	∅-	∅-
3pl	en-	**un-**	e-	eci-	∅-	∅-	∅-

un- [Acc +1 +pl]

The Imperfect Distribution Problem

Imperfect Distribution

	1sg	1pl	2sg	2pl	3sg	3pl	–
1sg			**eci-**	**eci-**	ku-	ku-	ku-
1pl			**eci-**	**eci-**	ci-	ci-	-as
2sg	en-	un-			e-	e-	e-
2pl	**eci-**en-	**eci-**un-			**eci-**	**eci-**	**eci-**
3sg	en-	un-	e-	**eci-**	Ø-	Ø-	Ø-
3pl	en-	un-	e-	**eci-**	Ø-	Ø-	Ø-

eci- [+2 +pl]

The Imperfect Distribution Problem

Imperfect Distribution

	1sg	1pl	2sg	2pl	3sg	3pl	–
1sg			**eci-**	**eci-**	ku-	ku-	ku-
1pl			**eci-**	**eci-**	ci-	ci-	-as
2sg	**en-**	**un-**			**e-**	**e-**	**e-**
2pl	**eci-**en-	**eci-**un-			**eci-**	**eci-**	**eci-**
3sg	en-	un-	**e-**	**eci-**	Ø-	Ø-	Ø-
3pl	en-	un-	**e-**	**eci-**	Ø-	Ø-	Ø-

eci- [+2]

The Subanalysis Problem

Subanalysis I: Chop off all instances of *eci*

	1sg	1pl	2sg	2pl	3sg	3pl	–
1sg			eci-	eci-	ku-	ku-	ku-
1pl			eci-	eci-	ci-	ci-	-as
2sg	en-	un-			e-	e-	e-
2pl	**eci-**en-	**eci-**un-			eci-	eci-	eci-
3sg	en-	un-	e-	eci-	Ø-	Ø-	Ø-
3pl	en-	un-	e-	eci-	Ø-	Ø-	Ø-

The Subanalysis Problem

Subanalysis II: Segment all instances of *eci* into *e-* and *ci-*

	1sg	1pl	2sg	2pl	3sg	3pl	–
1sg			**e-ci-**	**e-ci-**	ku-	ku-	ku-
1pl			**e-ci-**	**e-ci-**	ci-	ci-	-as
2sg	en-	un-			e-	e-	e-
2pl	**e-ci-**en-	**e-ci-**un-			**e-ci-**	**e-ci-**	**e-ci-**
3sg	en-	un-	e-	**e-ci-**	Ø-	Ø-	Ø-
3pl	en-	un-	e-	**e-ci-**	Ø-	Ø-	Ø-

Major Claims of this Talk

- Learning of morpheme meaning is governed by violable constraints. Constraint ranking produces affix inventories for different morphological frameworks
- Learning of meaning and subanalysis is combined by iterative optimization in the sense of Harmonic Serialism

Accuracy Measures for Affix Hypotheses

Informal and formal paradigm representation

	[+y]	[−y]
[+x]	*a*	*b*
[−x]	*a*	*ab*

$\{ \langle a, [+x +y] \rangle, \langle b, [+x -y] \rangle, \langle a, [-x +y] \rangle, \langle ab, [-x -y] \rangle \}$

Affix hypotheses with accuracy evaluation

	⟨form, meaning⟩	false positives	false negatives	implica. relation	precision	recall
a.	⟨*b*, [−y]⟩	–	–	↔	1	1
b.	⟨*a*, [+y]⟩	–	yes	←	1	$\frac{2}{3}$
c.	⟨*a*, [−x]⟩	–	yes	←	1	$\frac{2}{3}$
d.	⟨*a*, []⟩	yes	–	→	$\frac{3}{4}$	1
e.	⟨*b*, [+x]⟩	yes	yes	none	$\frac{1}{2}$	$\frac{1}{2}$
f.	⟨*a*, [−y]⟩	yes	yes	none	$\frac{1}{2}$	$\frac{1}{3}$

Imperfect Recall ~ False Negatives ~ Underinsertion

	1sg	**1pl**	**2sg**	**2pl**	**3sg**	**3pl**	**–**
1sg			**eci-**	**eci-**	ku-	ku-	ku-
1pl			**eci-**	**eci-**	ci-	ci-	-as
2sg	en-	un-			e-	e-	e-
2pl	**eci-**en-	**eci-**un-			**eci-**	**eci-**	**eci-**
3sg	en-	un-	e-	**eci-**	Ø-	Ø-	Ø-
3pl	en-	un-	e-	**eci-**	Ø-	Ø-	Ø-

eci- [+2 +pl]

Imperfect Precision ~ False Positives ~ Overinsertion

	1sg	**1pl**	**2sg**	**2pl**	**3sg**	**3pl**	**–**
1sg			**eci-**	**eci-**	ku-	ku-	ku-
1pl			**eci-**	**eci-**	ci-	ci-	-as
2sg	**en-**	**un-**			**e-**	**e-**	**e-**
2pl	**eci-**en-	**eci-**un-			**eci-**	**eci-**	**eci-**
3sg	en-	un-	**e-**	**eci-**	Ø-	Ø-	Ø-
3pl	en-	un-	**e-**	**eci-**	Ø-	Ø-	Ø-

eci- [+2]

Paradigmatic Readjustment Rules

Retraction Rules: block an affix from a paradigm cell of its Primary Distribution which is not in its actual distribution

Expansion Rules: extend an affix to a paradigm cell of its actual distribution which is not in its Primary Distribution

(Primary Distribution = the distribution of a morpheme predicted by its entry abstracting away from additional rules)

Paradigmatic Readjustment Rules

Expansion Rule Analysis (Rule of Referral)

	1sg	1pl	2sg	2pl	3sg	3pl	–
1sg			**eci-**	**eci-**	ku-	ku-	ku-
1pl			**eci-**	**eci-**	ci-	ci-	-as
2sg	en-	un-			e-	e-	e-
2pl	**eci-**en-	**eci-**un-			**eci-**	**eci-**	**eci-**
3sg	en-	un-	e-	**eci-**	Ø-	Ø-	Ø-
3pl	en-	un-	e-	**eci-**	Ø-	Ø-	Ø-

$[+2 -\text{pl}] \Rightarrow [+2 +\text{pl}] \;/\; [\text{Nom} +1]$___

eci- $[+2 +\text{pl}]$

Paradigmatic Readjustment Rules

Retraction Rule Analysis (Impoverishment Rule)

	1sg	1pl	2sg	2pl	3sg	3pl	–
1sg			**eci-**	**eci-**	ku-	ku-	ku-
1pl			**eci-**	**eci-**	ci-	ci-	-as
2sg	**en-**	**un-**			e-	e-	e-
2pl	**eci-**en-	**eci-**un-			**eci-**	**eci-**	**eci-**
3sg	en-	un-	e-	**eci-**	Ø-	Ø-	Ø-
3pl	en-	un-	e-	**eci-**	Ø-	Ø-	Ø-

$[+2 -\text{pl}] \rightarrow \text{Ø} / [\text{Acc} +1]$___

eci- $[+2]$

Learning of Morpheme Meaning as Optimization

- Meaning assignment is optimality-theoretic meaning optimization for affix strings
- *UnderInsertion and *OverInsertion ensure that **perfect** distributions correspond to perfect affixes
- Ranking of *UnderInsertion/*OverInsertion determines whether **imperfect** affix meanings fit an expansionist or a retractionist analysis
- Additional constraints and their ranking allow for fine tuning

Accuracy Constraints on Meaning Assignment

*OverInsertion: Assign ∗ to every paradigm cell where the hypothesis predicts the marker but the marker does not occur

*UnderInsertion: Assign ∗ to every paradigm cell where the marker occurs but the hypothesis does not predict it

Accuracy-based Evaluation of *un-*

	*OverIns	*UnderIns
☞ a. *un*-:[Acc +1 +pl]		
b. *un*-:[Acc +1]	$*_4$! (X:1sg)	
c. *un*-:[Acc +2 +pl]	$*_4$! (X:2pl)	$*_4$ (X:1pl)

Markedness Constraint

*Portmanteaux: Assign $(n-1)$ ∗'s to an affix hypothesis which contains n feature structures

Evaluation of *un-*

	*OverIns	*PortX	*UnderIns
☞ a. *un*-:[Acc +1 +pl]			
b. *un*-:[Nom −1][Acc +1 +pl]		*!	
c. *un*-:[Acc +1]	$*_4$! (X:1sg)		
d. *un*-:[Acc +2 +pl]	$*_4$! (X:2pl)		$*_4$ (X:1pl)

Constraints on Paradigmatic Coverage/Specificity

MaximalCoverage: Assign ✓ to every paradigm cell which is subsumed by the affix hypothesis

FeatureMinimality: Assign ∗ to every feature (value pair) in the affix hypothesis

Meaning Assignment to *eci*- under different Rankings

Expansionist Ranking: *OverInsertion ≫ *UnderInsertion

	*OverIns	*UnderIns
☞ a. *eci*-:[+2 +pl]		$*_2$ (1:2sg)
b. *eci*-:[+2]	$*_7$! (2sg,2sg:X,3:2sg)	
c. *eci*-:[Acc +2]	$*_2$! (3:2sg)	$*_5$ (2pl,2pl:X)

Retractionist Ranking: *UnderInsertion ≫ *OverInsertion

	*UnderIns	*OverIns
a. *eci*-:[+2 +pl]	$*_2$ (1:2sg)	
☞ b. *eci*-:[+2]		$*_7$! (2sg,2sg:X,3:2sg)
c. *eci*-:[Acc +2]	$*_5$ (2pl,2pl:X)	$*_2$! (3:2sg)

Ainu Agreement Morphemes under Different Rankings

(1) **Expansionist Ranking:** *OVERINS ≫ *PORTMX ≫ *UNDERINS

a. *en-* [Acc +1 −pl]
b. *un-* [Acc +1 +pl]
c. *eci-* [+2 +pl]
d. *e-* [Nom Abs +2 −pl]
e. *ku-* [Nom Abs +1 −pl]
f. *ci-* [Nom +1 +pl][Acc +3]
g. *-as* [Nom Abs +1 +pl]

(2) **Retractionist Ranking:** *PORTMX ≫ *UNDERINS ≫ *OVERINS

a. *en-* [Acc +1 −pl]
b. *un-* [Acc +1 +pl]
c. *eci-* [+2]
d. *e-* [+2 −pl]
e. *ku-* [Nom +1 −pl]
f. *ci-* [Erg +1 +pl]
g. *-as* [Nom Abs +1 +pl]

Greedy Algorithm for Learning and Subsegmentation

Input: a paradigm *P*, i.e. a set of ⟨affix string, meaning⟩-pairs
an evaluation metric *E* rating ⟨affix string, meaning⟩-pairs
an empty lexicon *L*

1 build the set *M* of all potential markers for *P*
2 choose the optimal marker $O \in M$ according to *E*
3 add *O* to *L* **and** remove the affix string of *O* from all ⟨affix string, meaning⟩-pairs ∈ *P*
4 **if** any ⟨affix string, meaning⟩-pair ∈ *P* has a non-empty affix string **goto** step **1** **else** output *L*

- Iterative local optimization akin to Harmonic Serialism
- Comparing hypotheses for different affix strings
- Competing segmentation options are bled by form removal

Resolving Subanalysis Options by Iterative Optimization

Cycle 1

	1sg	1pl	2sg	2pl	3sg	3pl	–
1sg			eci	eci			
1pl			eci	eci			
2sg		un			e	e	e
2pl	eci	eciun			eci	eci	eci
3sg		un	e	eci			
3pl		un	e	eci			

Hypotheses:
un-:[Acc +1 +pl]
eciun-:[Acc +1 +pl]
eci-:[+2 +pl]
eci-:[+2]
e-:[Abs +2]
e-:[+2]
. . .

← *eci-*:[+2 +pl] | *e-*:[+2] →

Cycle 2 (alternative A)

	1sg	1pl	2sg	2pl	3sg	3pl	–
1sg							
1pl							
2sg		un			e	e	e
2pl		un					
3sg		un	e				
3pl		un	e				

Hypotheses:
un-:[Acc +1 +pl]
e-:[Nom Abs +2 −pl]
e-:[Nom +2 −pl]
. . .

Cycle 2 (alternative B)

	1sg	1pl	2sg	2pl	3sg	3pl	–
1sg			ci	ci			
1pl			ci	ci			
2sg		un					
2pl	ci	ciun			ci	ci	ci
3sg		un		ci			
3pl		un		ci			

Hypotheses:
un-:[Acc +1 +pl]
ciun-:[Acc +1 +pl]
ci-:[+2 +pl]
ci-:[+2]
. . .

Additional Constraints for Subsegmentation

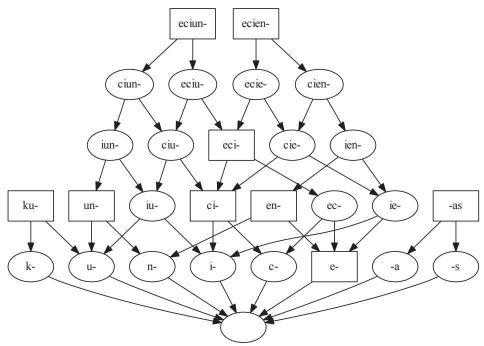
BEFREE: Assign * to an affix hypothesis whose form does not occur as an independent inflectional string in *P*

LENGTH: Assign ✓ to every segment in the form of the affix hypothesis

Rankings

BEFREE ≫ { *PORTMX *OVERINS *UNDERINS } ≫ MAXCOV! ≫ LENGTH!

Possible Forms



Cycle 1: Input to Learner & Evaluation

P =

	1sg	1pl	2sg	2pl	3sg	3pl	–
1sg	–	–	eci-	eci-	ku-	ku-	ku-
1pl	–	–	eci-	eci-	ci-	ci-	-as
2sg	en-	un-	–	–	e-	e-	e-
2pl	ecien-	eciun-	–	–	eci-	eci-	eci-
3sg	en-	un-	e-	eci-			
3pl	en-	un-	e-	eci-			

L = { }

	*OvIn	*UnIn	MaxC	Lngth
☞ a. *en-*:[Acc +1 −pl]			$\checkmark_4$	$\checkmark_2$
☞ b. *un-*:[Acc +1 +pl]			$\checkmark_4$	$\checkmark_2$
c. *-as*:[Nom Abs +1 +pl]			$\checkmark_1$!	$\checkmark_2$
d. *eci-*:[+2 +pl]		$*_2$	$\checkmark_9$	$\checkmark_3$
e. *ku-*:[Nom Abs +1 −pl]		$*_2$	$\checkmark_1$!	$\checkmark_2$
f. *ci-*:[+2 +pl]		$*_4$!	$\checkmark_9$	$\checkmark_2$

Cycle 4: Learner State & Eval *OverIns ≫ *UnderIns

P =

	1sg	1pl	2sg	2pl	3sg	3pl	–
1sg	–	–	eci-	eci-	ku-	ku-	ku-
1pl	–	–	eci-	eci-	ci-	ci-	
2sg			–	–	e-	e-	e-
2pl	eci-	eci-	–	–	eci-	eci-	eci-
3sg			e-	eci-			
3pl			e-	eci-			

L = { *en-*:[Acc +1 −pl], *un-*:[Acc +1 +pl]], *-as*:[Nom Abs +1 +pl] }

	*OvIn	*UnIn	MaxC	Lngth
☞ a. *eci-*:[+2 +pl]		$*_2$	$\checkmark_9$	$\checkmark_3$
b. *ku-*:[Nom Abs +1 −pl]		$*_2$	$\checkmark_1$!	$\checkmark_2$
c. *ci-*:[+2 +pl]		$*_4$!	$\checkmark_9$	$\checkmark_2$
d. *e-*:[Abs +2]		$*_6$!	$\checkmark_{10}$	$\checkmark_1$

Cycle 4: Learner State & Eval *UnderIns ≫ *OverIns

P =

	1sg	1pl	2sg	2pl	3sg	3pl	–
1sg	–	–	eci-	eci-	ku-	ku-	ku-
1pl	–	–	eci-	eci-	ci-	ci-	
2sg			–	–	e-	e-	e-
2pl	eci-	eci-	–	–	eci-	eci-	eci-
3sg			e-	eci-			
3pl			e-	eci-			

L = { *en-*:[Acc +1 −pl], *un-*:[Acc +1 +pl]], *-as*:[Nom Abs +1 +pl] }

	*OvIn	*UnIn	MaxC	Lngth
a. *eci-*:[+2]		$*_7$!	$\checkmark_{18}$	$\checkmark_3$
b. *ku-*:[Nom +1 −pl]		$*_2$!	$\checkmark_5$	$\checkmark_2$
c. *ci-*:[−3]		$*_{15}$!	$\checkmark_{28}$	$\checkmark_2$
☞ d. *e-*:[+2]		$*_2$	$\checkmark_{18}$	$\checkmark_1$

BeFree ≫ *PortMx ≫ *OverIn ≫ *UnderIn

Segmentation

	1sg	1pl	2sg	2pl	3sg	3pl	–
1sg	–	–	eci-	eci-	ku-	ku-	ku-
1pl	–	–	eci-	eci-	ci-	ci-	-as
2sg	en-	un-	–	–	e-	e-	e-
2pl	eci-en-	eci-un-	–	–	eci-	eci-	eci-
3sg	en-	un-	e-	eci-			
3pl	en-	un-	e-	eci-			

Lexicon

i.	*en-*	[Acc +1 −pl]	v.	*ku-*	[Nom Abs +1 −pl]
ii.	*un-*	[Acc +1 +pl]	vi.	*e-*	[Nom Abs +2 −pl]
iii.	*-as*	[Nom Abs +1 +pl]	vii.	*ci-*	[Nom Abs +1 +pl]
iv.	*eci-*	[+2 +pl]			

BeFree ≫ *Portmx ≫ *UnderIn ≫ *OverIn

Segmentation

	1sg	1pl	2sg	2pl	3sg	3pl	–
1sg	–	–	e-ci-	e-ci-	ku-	ku-	ku-
1pl	–	–	e-ci-	e-ci-	ci-	ci-	-as
2sg	en-	un-	–	–	e-	e-	e-
2pl	e-ci-en-	e-ci-un-	–	–	e-ci-	e-ci-	e-ci-
3sg	en-	un-	e-	e-ci-			
3pl	en-	un-	e-	e-ci-			

Lexicon

i.	*en-*	[Acc +1 −pl]	iv.	*e-*	[+2]
ii.	*un-*	[Acc +1 +pl]	v.	*ku-*	[Nom +1 −pl]
iii.	*-as*	[Nom Abs +1 +pl]	vi.	*ci-*	[−3]

*Portmx ≫ *OverIn ≫ *UnderIn ≫ ... ≫ BeFree

Cycle 1

	*OvIn	*UnIn	MaxC	Lngth
☞ a. *n-*:[Acc +1]			✓$_8$	✓$_1$
b. *en-*:[Acc +1 −pl]			✓$_4$	✓$_2$
c. *un-*:[Acc +1 +pl]			✓$_4$	✓$_2$
d. *-as*:[Nom Abs +1 +pl]			✓$_1$	✓$_2$
e. *-a*:[Nom Abs +1 +pl]			✓$_1$	✓$_1$
f. *-s*:[Nom Abs +1 +pl]			✓$_1$	✓$_1$
g. *eci-*:[+2 +pl]		*$_2$	✓$_9$	✓$_3$
h. *ec-*:[+2 +pl]		*$_2$	✓$_9$	✓$_2$
i. *ku-*:[Nom Abs +1 −pl]		*$_2$	✓$_1$	✓$_2$
j. *k-*:[Nom Abs +1 −pl]		*$_2$	✓$_1$	✓$_1$
k. *u-*:[Acc +1 +pl]		*$_3$	✓$_4$	✓$_1$
l. *ci-*:[+2 +pl]		*$_4$	✓$_9$	✓$_2$
m. *c-*:[+2 +pl]		*$_4$	✓$_9$	✓$_1$
n. *i-*:[+2 +pl]		*$_4$	✓$_9$	✓$_1$
o. *n-*:[Acc +1 −pl]		*$_4$	✓$_4$	✓$_1$
p. *n-*:[Acc +1 +pl]		*$_4$	✓$_4$	✓$_1$

Conclusion

- The laid out framework of constraints for evaluating and comparing the accuracy of marker hypotheses closely mirrors the informed linguists judgments and makes them explicit
- Their possible rankings implement the different preferences needed to fit different morphological frameworks
- Greedy iterative optimization makes it possible to combine learning and segmentation without a global search through all possible analytical options

Bank, Sebastian & Trommer, Jochen (submitted), Inflectional Learning as Local Optimization. *Morphology*.

Tamura, Suzuko (2000), *The Ainu Language*. Tokyo: Sanseido.