Dealing with Medieval Text

Hans van Halteren
Radboud University Nijmegen
hvh@let.ru.nl

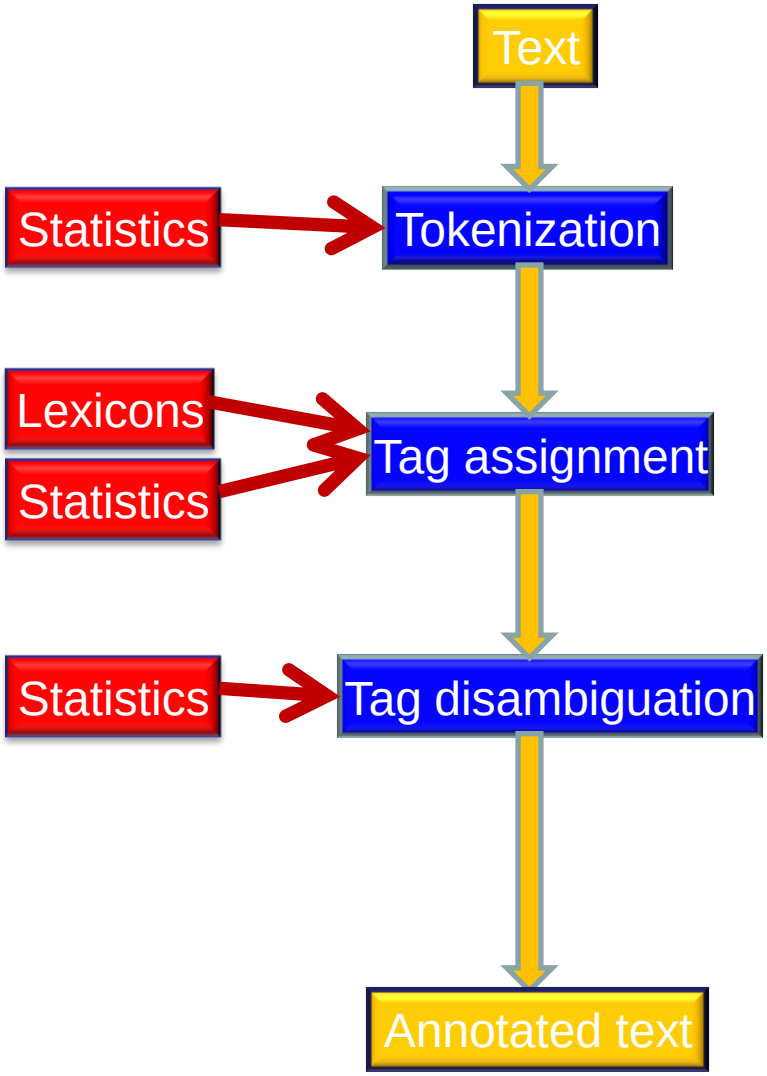
Overview

- Word Class tagging
  - Introduction to tagging
  - Special problem: orthographic variation
  - Our solutions to the orthography problem
  - The Adelheid system
- Authorship attribution
  - Introduction to attribution
  - Special opportunities: orthographic variation
  - A difficult case study: Rijmkroniek

Word Class Tagging (POS tagging)

Introduction / Reminder

Standard Architecture



Software: Stages

- Tokenization
  - Identifying tokens
    - Splitting off punctuation
    - Recognizing multi-token units (he’s)
    - Recognizing multi-unit tokens (Ministry of Education)
    - Identifying utterance boundaries

Software: Stages

- Potential tag/lemma assignment
  - Lexicon lookup
    - Known forms
    - Possibly with morphological grammar
  - Unknown word handling
    - Machine learning

Software: Stages

- Contextual disambiguation
  - Rule based
    - Human: ENGCG
    - Machine: Transformation Based Learning (TBL, Brill)
  - Probabilistic
    - HMM, SVM, Maximum Entropy, CRF, ...
    - TiMBL, WPDV
  - Combination of various models
    - With left-to-right and right-to-left tagging

State of the Art

- “POS-tagging a solved problem”
  - “English 97.xx%, other languages > 95%”
- However
  - Which other languages?
  - Which tag sets?
- And especially however for us:
  - ***On modern text!***

Medieval Text: Not as Easy

Looks quite different

Goal: Tagger-Lemmatizer

Our Goal: automatically add tags and lemmas

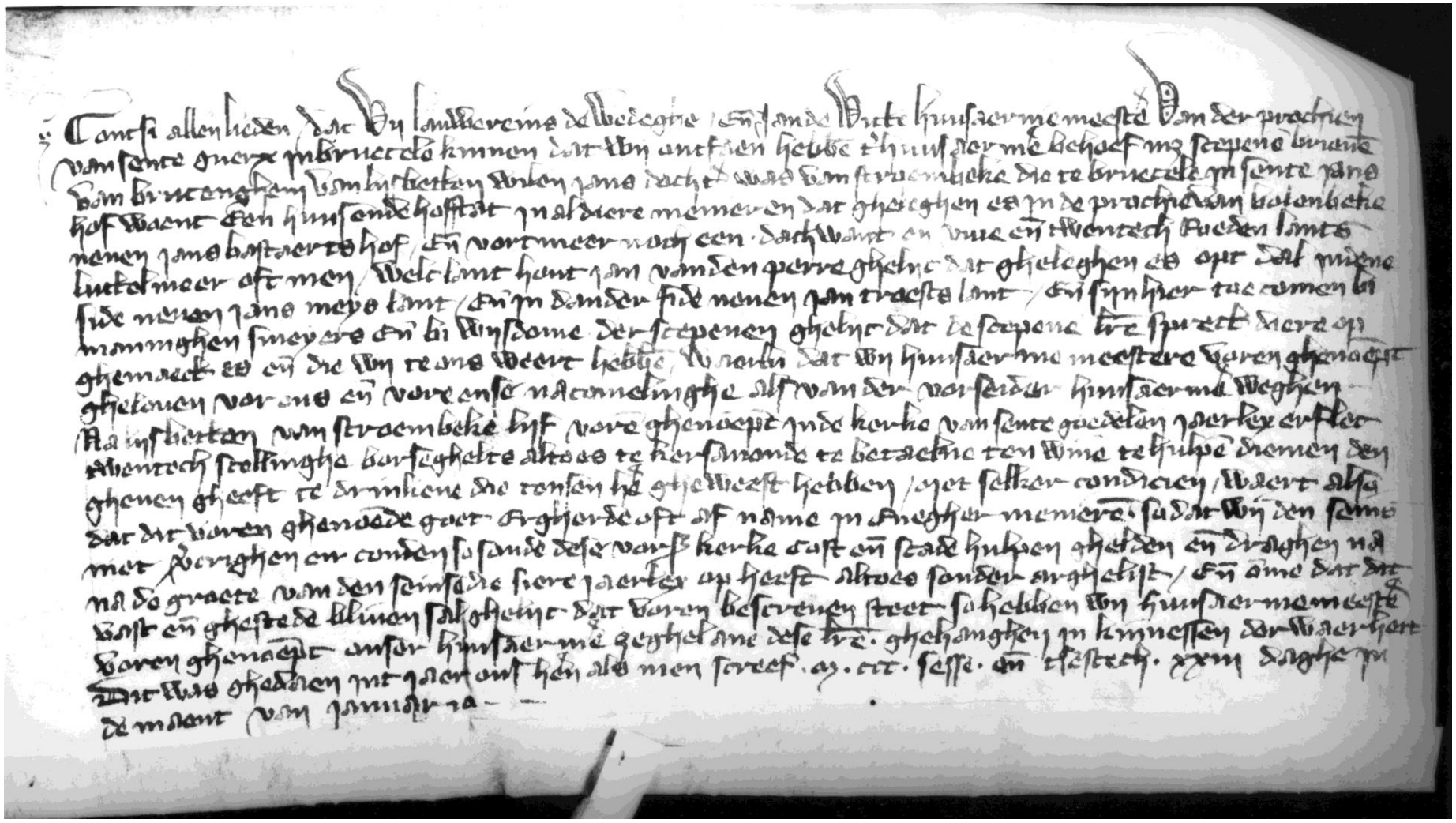
- starting from transcribed text
- tags from a reasonably complex tagset
  - 184 basic tags
  - plus combination tags for enclitic forms: 437 observed in data
- trained on 800K word corpus van Reenen - Mulder
- aiming for the psychological barrier of 95% accuracy

Token	**Tag**	**Lemma**
och	Conj(coord)	of
en	Adv(neg)	en
betalden	V(fin,past,lex,formn)	betalen
tesen	Adp()+Pron(dem,formn)	te+deze
vorsprokene	Adj(formn)	voorgesproken
tide	N(sing,forme)	tijd
.	Punc(lp)	.

Tagging Medieval Text

- Why not use “normal” existing systems?
  - Not able to properly process older Dutch
  - Assume standardized
    - Spacing
    - Punctuation
    - Spelling
  - None of these are present, thus causing problems

14th Century Dutch Charters



14th Century Dutch Charters

C108p39304 Blok862 gecollationeerd.280394.HD

wy borghermestere ende raet van groninghen bekennen ende betughen met dezen openen breue dat vor ons quam ghelmer storm ende becande dat hie heft vercoft rodetyden vyertyendehalf gras landes met al horen to behoren vor ene summe gheldes de ghelmer vorseit vol ende al betaelt js ende deze vy ertyendehalf gras landes vorseit droech ghelmer vorseit vp rodetyden vorseit ende sinen erfghenamen vrij ende quiit met allen rechte ende eghendome eweliken to bruken ende to besitten dit vorseide land js gheleghen in lywerder wolt vp de noerd zide van den wolt graue daer viif grase landes van gheleghen ziin by rodetyden erue vorseit dat an de oester zide leghet ende viif graze landes daer tette mellens erue by gheleghen js an de oester zide ende vyerdehalf gras landes an de noerd zide van den vorseiden viif grasen daer een sloet en tuschen gaet dat or kunde wy met onser stad seghel . ghegheuen jnt jaer ons heren dusent drehondert dre ende neghentich vp sente nycholaus auond do wicbolt euerdes euerd sickinc johan van den berghe ende jacob schelleghe borghermestere waren onser stad

Variation: Punctuation and Spacing

wy borghermestere ende raet van groninghen bekennen ende betughen met dezen openen breue dat vor ons quam ghelmer storm ende becande dat hie heft vercoft rodetyden vyertyendehalf gras landes met al horen to behoren vor ene summe gheldes de ghelmer vorseit vol ende al betaelt js ende deze vy ***<nl>*** ertyendehalf gras landes vorseit droech ghelmer vorseit vp rodetyden vorseit ende sinen erfghenamen vrij ende quiit met allen rechte ende eghendome eweliken to bruken ende to besitten dit vorseide land js gheleghen in lywerder wolt vp de noerd zide van den wolt graue daer viif grase landes van gheleghen ziin by rodetyden erue vorseit dat an de oester zide leghet ende viif graze landes daer tette mellens erue by gheleghen js an de oester zide ende vyerdehalf gras landes an de noerd zide van den vorseiden viif grasen daer een sloet en tuschen gaet dat or ***<nl>*** kunde wy met onser stad seghel . ghegheuen jnt jaer ons heren dusent drehondert dre ende neghentich vp sente nycholaus auond do wicbolt euerdes euerd sickinc johan van den berghe ende jacob schelleghe borghermestere waren onser stad

Variation: Spelling

Adverb *gelijk*

ghelijc (373) gheliic (86) gelijc (64) ghelike (54) ghelijch (33) ghelyc (19) gheliich (10) gelijch (9) gelike (9) gheliken (9) euengheliken (4) gelyck (4) ghelich (4) ghelic (3) gelic (2) geliic (2) ghelijck (2) dinghelike (1) euenghelike (1) evenghelike (1) evenghelyc (1) ghelijcke (1) ghelijct (1) gheljjch (1) ghelljc (1) ghlijc (1) gilycs (1) like (1)

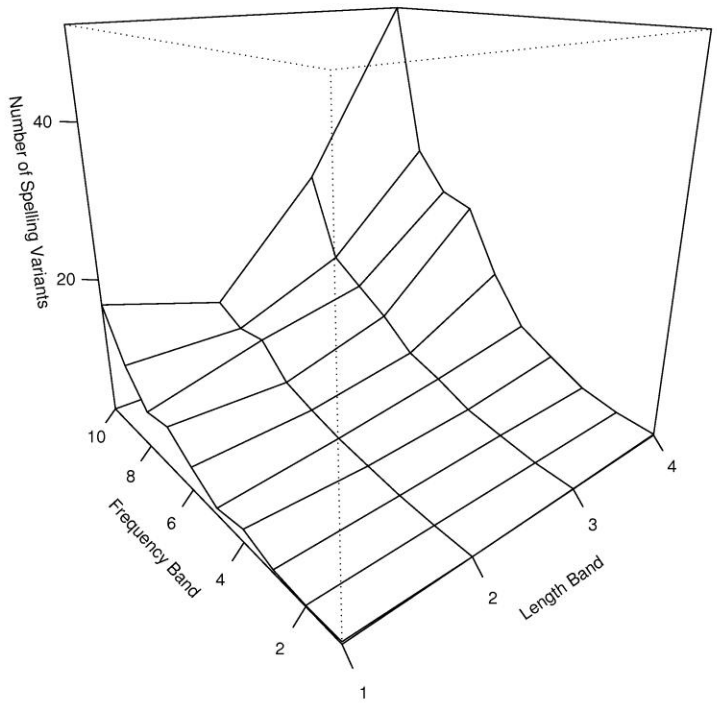
Proper name *Gerard*

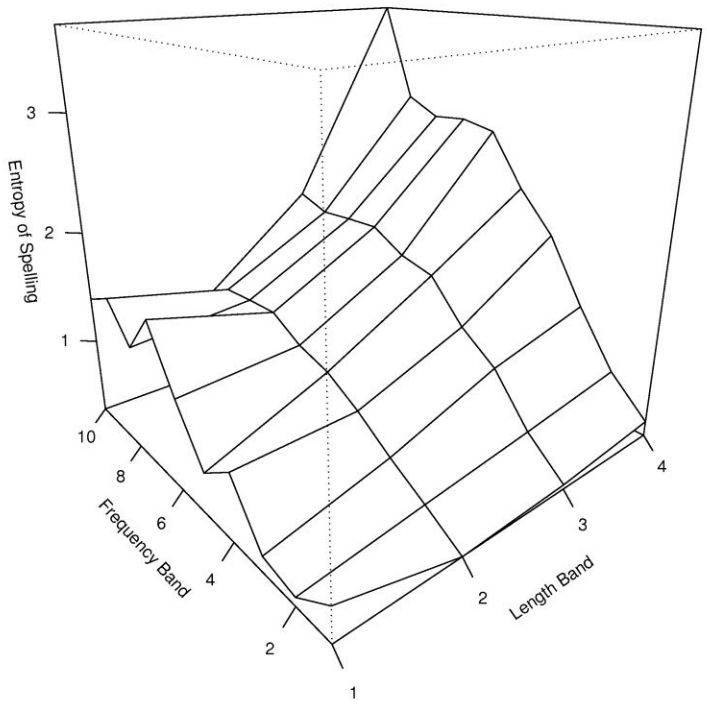
gheriit (121) gherijt (111) gherart (84) gheret (70) gherit (70) gherijd (58) gerart (56) gheert (55) gheriid (54) gheraerd (47) gherd (47) ghert (46) ghered (37) gheraet (24) gheeraed (19) gherard (18) gheraert (16) gert (12) gerat (11) gerit (10) gheerd (10) geraert (8) gerd (8) gheryt (8) gerijt (7) gheeraerd (7) gheerard (7) gherret (7) geerd (6) gherid (6) geraet (5) geret (5) gheraed (5) geert (4) gerijd (4) gheeraert (4) gheredt (4) gheryd (4) gherrijd (3) ghierart (3) gered (2) gereet (2) geyrart (2) gheeraerdt (2) gheeraet (2) gheerit (2) gher (2) gherairt (2) gherardt (2) gherat (2) gherrijt (2) gherut (2) garret (1) ger (1) geraed (1) gerairt (1) gerard (1) gerid (1) geriit (1) geryt (1) gheerlec (1) gheraird (1) gherrid (1) gherud (1) gherydijn (1) gierkijn (1)

Variation: Spelling

Number of variants **Entropy**

vs Length / Frequency





Solutions for Orthographic Variation

Study and Adapt

Software: Stages

- Tokenization
  - Identifying tokens
    - Reinterpretation of word separation
  - Identifying utterance boundaries
    - Don’t exist: just tag whole manuscript at once
- Potential tag/lemma assignment
  - Lexicon lookup
    - Expected variant forms, on basis of known variation
  - Unknown word handling
    - Nearest neighbours in expanded lexicon

Solutions for Orthographic Variation

Tokenisation: two steps

- Initial model
  - Machine learning: WPDV
  - Features: separation as written, left/right context
  - Context: 1-5 characters, string upto next whitespace
  - NB 3 ML feature slots, context features overloaded
- Reestimation after round of tagging
  - Only for positions where initial model uncertain
  - Context: direct and indirect context of split position
  - Direct: parts of token / tokens; whole + 1-3 chars
  - Indirect: adjacent tag/lemma uni-/bi-/trigrams

Solutions for Orthographic Variation

Potential tag/lemma assignment: two steps

- Expand lexicon
  - With forms predicted from observed variation
- Reexpand lexicon
  - With still missing forms from the test set
  - Using closest forms (having correct tag)

Solutions for Orthographic Variation

Phase 1: Determine character-level variation cost

- Based on form pairs with only one difference
- Levenshtein cost reduced every time observed (1→0)

	Substitution
e ↔ i	.050
i ↔ y	.086
d ↔ t	.235
c ↔ k	.598
b ↔ p	.969
b ↔ z	.997

	Insertion	**Doubling**
e	.017	.004
h	.085	.085
n	.459	.339
r	.769	.661
m	.956	.849
b	.979	.949

Solutions for Orthographic Variation

Phase 2: Build token-variation grid

- with alignment software aligning multiple forms

g	h	e	b	o	e	r	t	e	14
g		e	b	v	e	r	d	e	11
			b	o	e	r	t	e	7
g	h	e	b	o		r	t	e	3
g		e	b	o		r	t	e	3
g		e	b	v		r	t	e	2
g	h	e	b	v	e	r	t	e	1
g	h	e	b	v	e	r	d	e	1
g		e	b	o	i	r	d	e	1
			b	o		r	t	e	1

- later on, will generate combination variants

Solutions for Orthographic Variation

Phase 3: Derive rules which appear to be more general

- Character grid positions: focus char + left and right context
- Variant for position character seen for many lemmas

Substitution			**Deletion**			**Insertion**	
#s__<__e → c	.73		eg__h__#	.90		#g__ __el → h	.76
t__z__# → s	.71		f__f__#	.87		#g__ __es → h	.75
an__d__# → t	.70		t__h_en	.85		g__ __el → h	.71
l__d__# → t	.70		en__n__e#	.78		dag__ __e → h	.70
s__<__o → c	.68		ike__n__#	.78		#g__ __e → h	.66

Solutions for Orthographic Variation

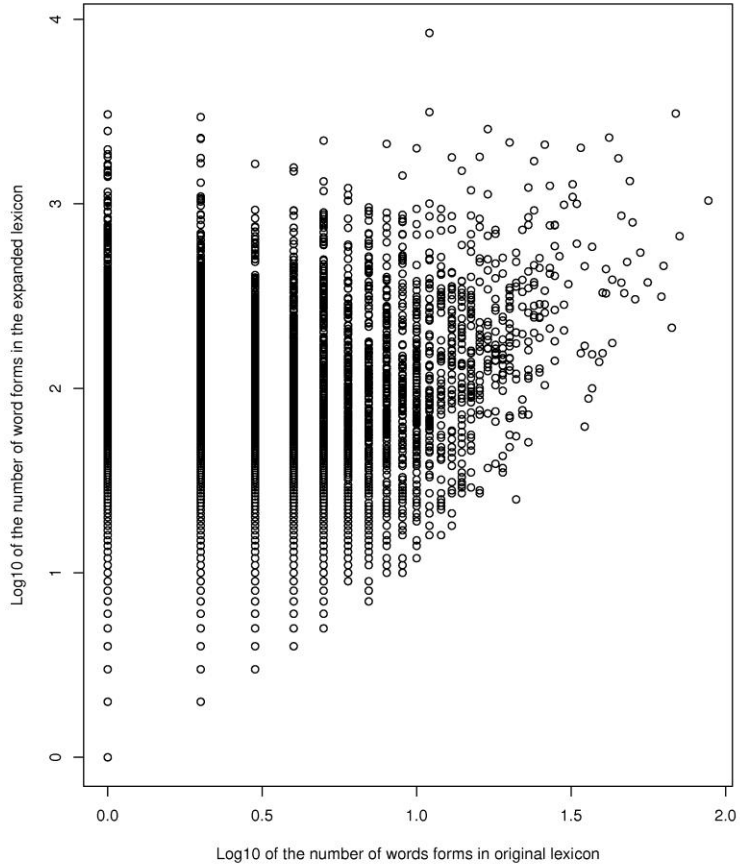
Phase 4: Generate variants

- Start from observed variant
- Allow up to √tokenlength changes
- First observed variants for token, then rule-based variants
- Keep change probability over threshold
- Filter out variants with impossible trigrams (and suffix 4-grams)
- Reassign counts, based on C(observed) and P(change)
- Expands number of lexicon tokens from ~50K to ~1.3M

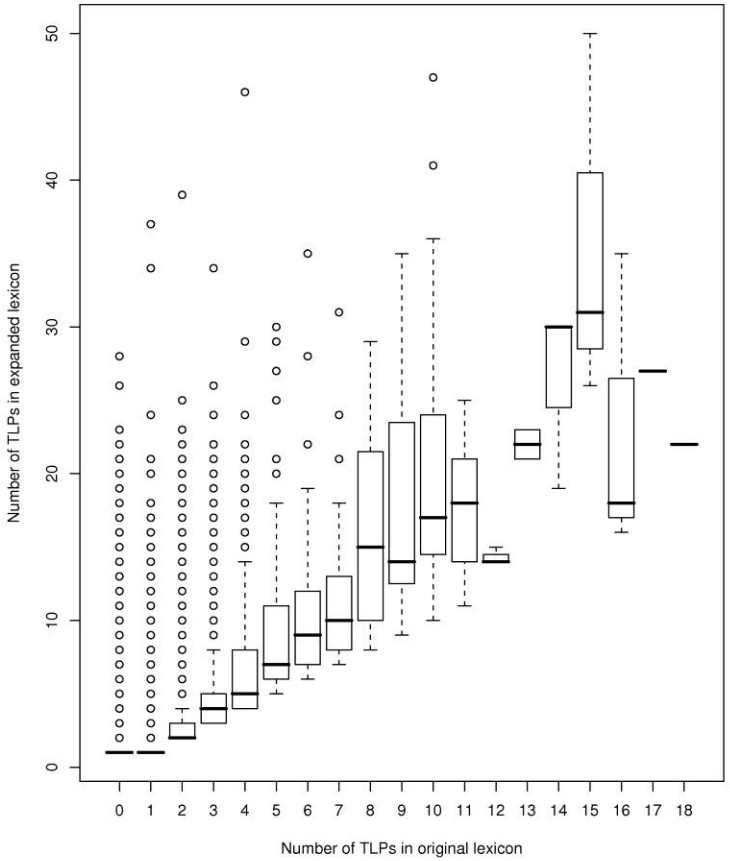
Token	**Observed**	**Generated**
gheboerte	14	7.73
gebverte	2	1.94
ghebvrde	0	0.41
boerde	0	0.14
heboeirte	0	0.0005

Solutions for Orthographic Variation

Number of variants in expansion



Ambiguity in expansion



Solutions for Orthographic Variation

Phase 5: Try to dynamically adapt lexicon

- Token-tag combinations from unknown token module
  - ~ 5K per 80Kw test set
- Find Levenshtein-closest in expanded lexicon
  - With the right tag

E.g. Lemma: *voorgezegd*

- 214 forms observed
- Expanded to 1992 forms
- Identified two further in test: *voerseiit* and *voregeseds*

Evaluation: Experimental Setup

1. Tokenize: find word boundaries
2. Determine potential tags and lemmas for each token
   - Known tokens: lexicon lookup
   - Unknown tokens: WPDV machine learner
3. Assign probabilities to potential tags and lemmas
   - SVMTool (Giménez and Márquez)
   - TnT (Brants)
   - WPDV (van Halteren)
   - Simple additive combination

Tested in 10-fold cross-validation on CRM

- Separation at charter level
- Parameter setting: partly trained, partly derived from pilot

Evaluation Results

The Bottom Line: overall accuracy

- Recall: “Gold Standard” reproduced

	Tag	Lemma
Token recognized	99.20%	99.20%
Gold annotation proposed		
Gold annotation selected		

Evaluation Results

The Bottom Line: overall accuracy

- Recall: “Gold Standard” reproduced

	Tag	Lemma
Token recognized	99.20%	99.20%
Gold annotation proposed	98.75%	97.28%
Gold annotation selected		

Evaluation Results

The Bottom Line: overall accuracy

- Recall: “Gold Standard” reproduced

	Tag	Lemma
Token recognized	99.20%	99.20%
Gold annotation proposed	98.75%	97.28%
Gold annotation selected	94.97%	94.88%

Almost! However...

Evaluation Results

Gold Standard not all that Gold (as usual)

- Annotation errors
  - Very modest spot check
  - Tag deviations: 1 in 3 is corpus error
    - Tag score += about 1.5%
  - Lemma deviations: 1 on 10 is corpus error
    - Lemma score += about 0.5%
- Inconsistencies / Unclear standard
  - About 30% lemma errors: proper nouns
  - Inconsistencies in first name lemmas
    - E.g. *Gerard/Gerhard/Gerrit*: 419 errors
    - Around 5% of all errors
    - Lemma score += about 0.2%

Evaluation Results

Token separation

- Recall for initial expanded lexicon

	Token	Tag	Lemma
As written	96.05%	91.73%	90.90%
First estimate	99.11%	94.85%	93.88%
Re-estimate	99.20%	94.94%	93.96%

- Initial needed
- Re-estimation not essential

Evaluation Results

Lexicon improvement

- Recall for re-estimated tokens

	Tag	Lemma
Known forms only		93.11%
Expanded lexicon		93.96%
With test token adaptation		94.88%

Evaluation Results

Lexicon improvement

- Recall for re-estimated tokens

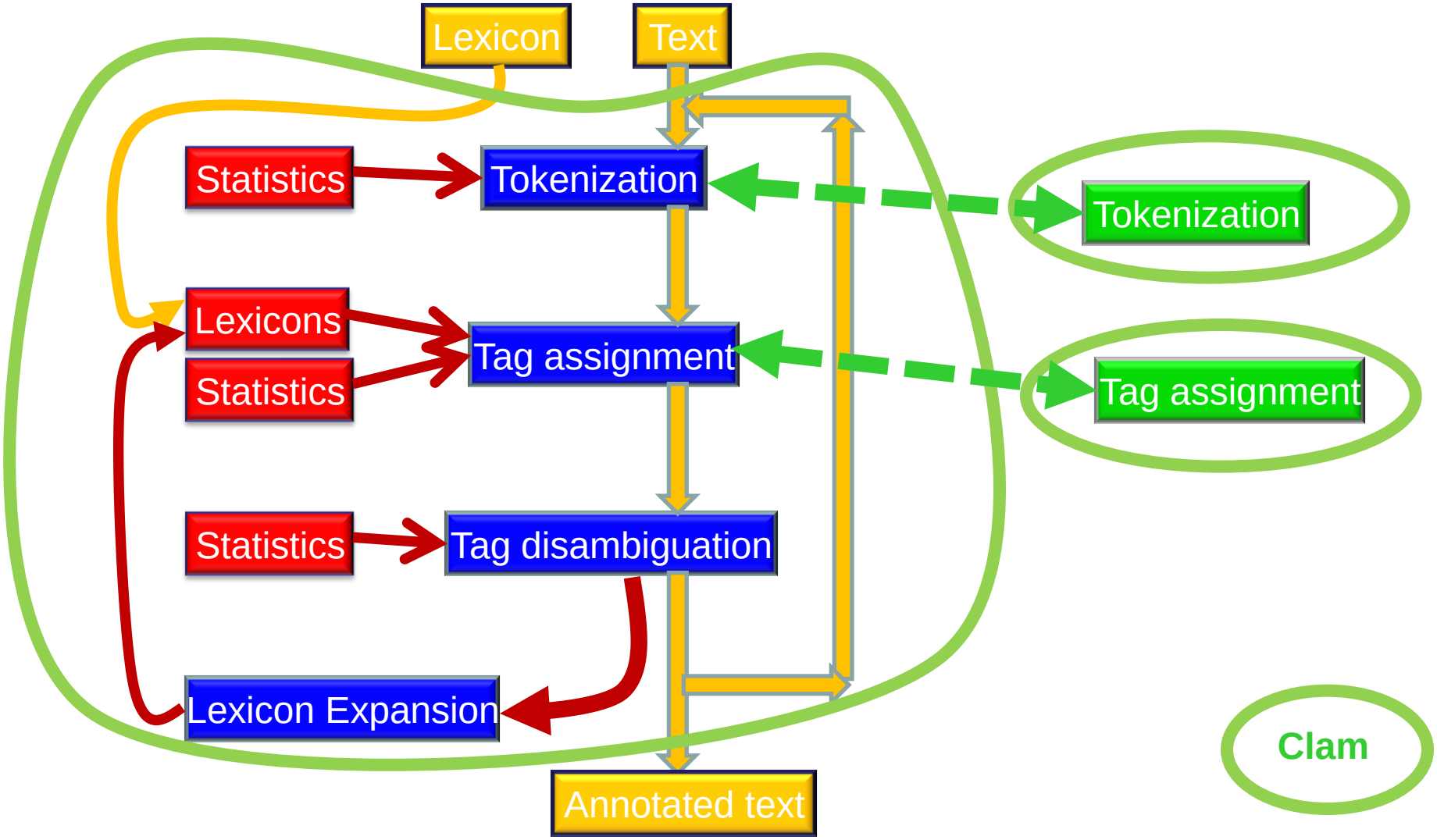
	Tag	Lemma
Known forms only	94.91%	93.11%
Expanded lexicon	94.94%	93.96%
With test token adaptation	94.97%	94.88%

- Expansion not needed for tags
  - Unknown token model coping remarkably well
- But essential for lemmas

Adelheid

Access through the Clarin Infrastructure

Adelheid Architecture



Adelheid through Clarin

- System now available
  - Through Clarin infrastructure
  - More efficient
  - Using XML data formats
  - With user manuals, incl. Demonstration scenarios
- Interface: Clam
  - http://lux17.mpi.nl/adelheid
  - http://wwwlands2.let.kun.nl/adelheid/
  - Please do not use until release announced

Visualisation and Annotation in Clarin

Annotation tool: Why?

- Example of the (XML) output

```
<token Tform="dese" Tag="Pron(dem,forme)" Lemma="deze" Tpos="1/25-28" Mform="dese" Aform="dese" Src="sys" Conf="0.7287">
  <tlp ATag="Pron(dem,forme)" ALemma="deze" AProb="0.7287"></tlp>
  <tlp ATag="Art(def,forme)" ALemma="deze" AProb="0.2190"></tlp>
  <tlp ATag="N(prop,forme)" ALemma="dieze" AProb="0.0523"></tlp>
</token>
<sep Tpos="1/29" Msep="True" Mform=" " Tsep="True" Asep="True" Src="sys" Conf="0.9992"></sep>
<token Tform="letteren" Tag="N(plu,formn)" Lemma="letter" Tpos="1/29-36" Mform="lett__en" Aform="letteren" Src="sys"
    Conf="0.6636">
  <tlp ATag="N(plu,formn)" ALemma="letter" AProb="0.6636"></tlp>
  <tlp ATag="N(sing,formn)" ALemma="letter" AProb="0.3364"></tlp>
</token>
<sep Tpos="1/37" Msep="True" Mform=" " Tsep="True" Asep="True" Src="sys" Conf="0.9994"></sep>
<token Tform="selen" Tag="V(fin,pres,aux_cop,formn)" Lemma="zullen" Tpos="1/37-41" Mform="selen" Aform="selen" Src="sys"
    Conf="0.6776">
  <tlp ATag="V(fin,pres,aux_cop,formn)" ALemma="zullen" AProb="0.6776"></tlp>
  <tlp ATag="V(infin)" ALemma="zellen" AProb="0.0943"></tlp>
  <tlp ATag="N(prop,forms)" ALemma="seel" AProb="0.0786"></tlp>
  <tlp ATag="V(fin,pres,aux_cop)+Pron(pers,3,sing)" ALemma="zullen+hij" AProb="0.0691"></tlp>
  <tlp ATag="N(plu,formn)" ALemma="ziel" AProb="0.0321"></tlp>
  <tlp ATag="N(prop,formn)" ALemma="seel" AProb="0.0269"></tlp>
  <tlp ATag="N(sing,formn)" ALemma="ziel" AProb="0.0182"></tlp>
  <tlp ATag="N(prop,formn)" ALemma="zelle" AProb="0.0031"></tlp>
</token>
```

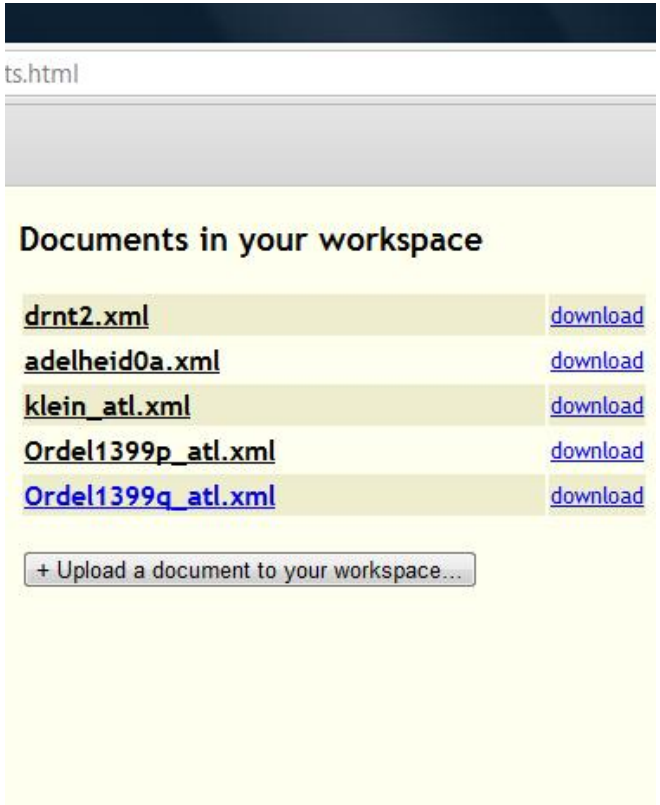
Annotation tool

- Dedicated tool for
  - Visualization
  - Adjusting annotation
  - Details below
- Tool built by Edia in Amsterdam
- Also accessible through Clarin infrastructure
  - http://lux17.let.kun.nl/adelheidanntool
  - http://adelheid.edia.nl/adelheid-tagger
  - Please do not use until release announced

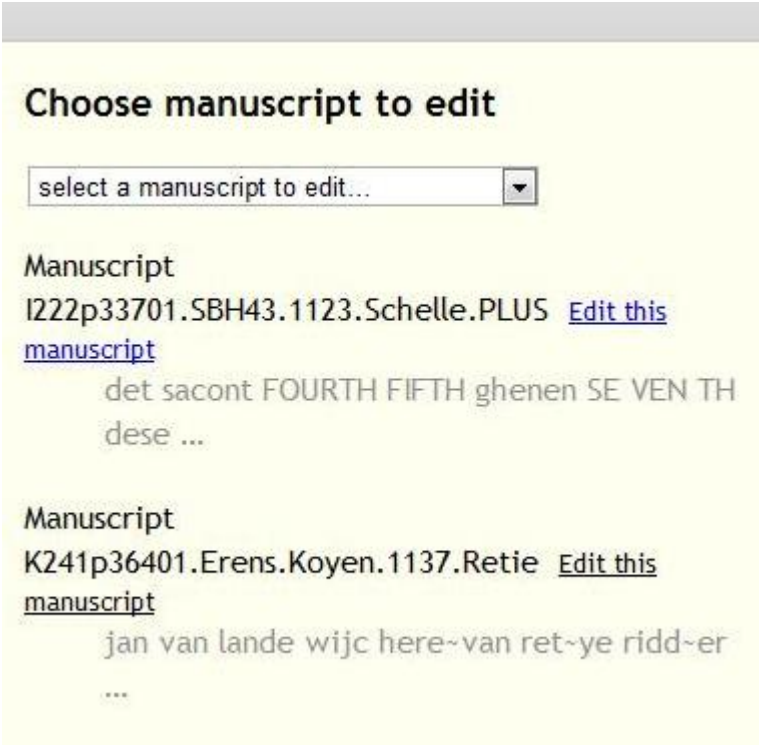
In case life demo not available:
Screenshots
(with some silly debugging artifacts)

Annotation tool: Functionality

- Up- and downloading annotation files



- Selecting manuscripts for processing

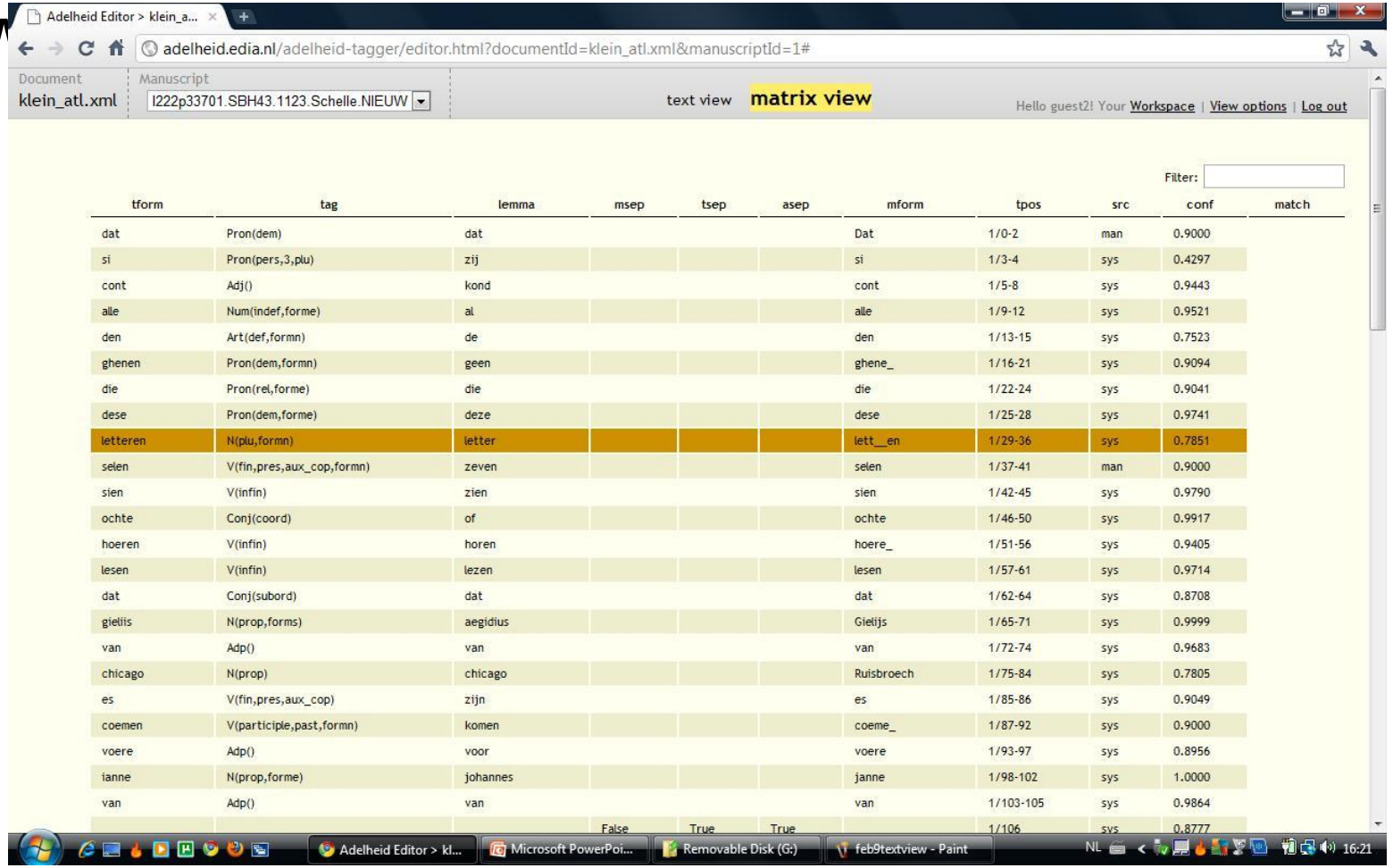


Annotation tool: Functionality

- Seeing tokens, tags and lemmas: Text View

Annotation tool: Functionality

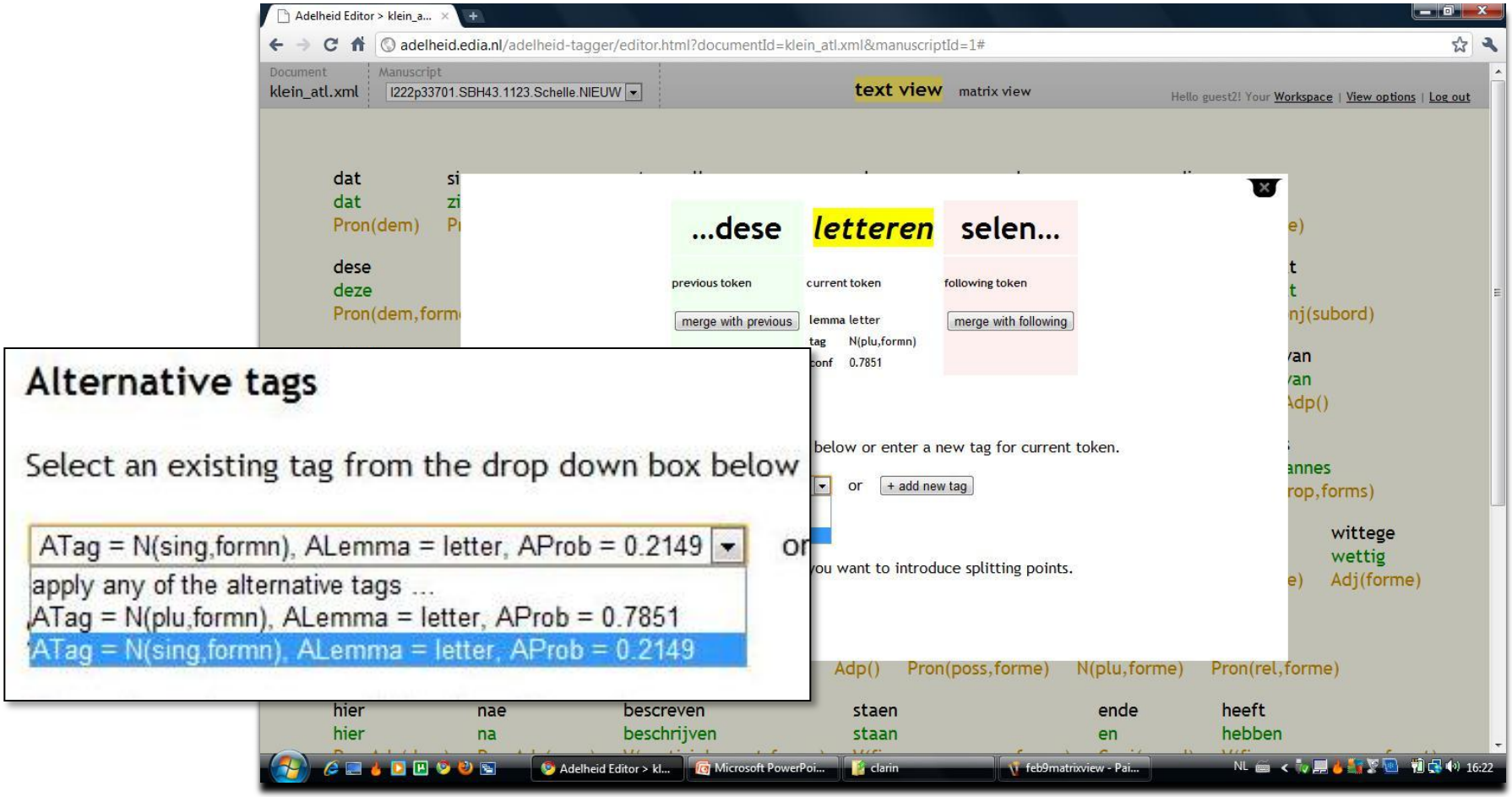
- Seeing tokens, tags and lemmas: Matrix View



tform	tag	lemma	msep	tsep	asep	mform	tpos	src	conf	match
dat	Pron(dem)	dat				Dat	1/0-2	man	0.9000	
si	Pron(pers,3,plu)	zij				si	1/3-4	sys	0.4297	
cont	Adj()	kond				cont	1/5-8	sys	0.9443	
alle	Num(indef,forme)	al				alle	1/9-12	sys	0.9521	
den	Art(def,formn)	de				den	1/13-15	sys	0.7523	
ghenen	Pron(dem,formn)	geen				ghene_	1/16-21	sys	0.9094	
die	Pron(rel,forme)	die				die	1/22-24	sys	0.9041	
dese	Pron(dem,forme)	deze				dese	1/25-28	sys	0.9741	
letteren	N(plu,formn)	letter				lett__en	1/29-36	sys	0.7851	
selen	V(fin,pres,aux_cop,formn)	zeven				selen	1/37-41	man	0.9000	
sien	V(infin)	zien				sien	1/42-45	sys	0.9790	
ochte	Conj(coord)	of				ochte	1/46-50	sys	0.9917	
hoeren	V(infin)	horen				hoere_	1/51-56	sys	0.9405	
lesen	V(infin)	lezen				lesen	1/57-61	sys	0.9714	
dat	Conj(subord)	dat				dat	1/62-64	sys	0.8708	
gielis	N(prop,forms)	aegidius				Gielijs	1/65-71	sys	0.9999	
van	Adp()	van				van	1/72-74	sys	0.9683	
chicago	N(prop)	chicago				Ruisbroech	1/75-84	sys	0.7805	
es	V(fin,pres,aux_cop)	zijn				es	1/85-86	sys	0.9049	
coemen	V(participle,past,formn)	komen				coeme_	1/87-92	sys	0.9000	
voere	Adp()	voor				voere	1/93-97	sys	0.8956	
ianne	N(prop,forme)	johannes				janne	1/98-102	sys	1.0000	
van	Adp()	van				van	1/103-105	sys	0.9864	
			False	True	True		1/106	sys	0.8777	

Annotation tool: Functionality

- Choosing alternative suggested annotation



Annotation tool: Functionality

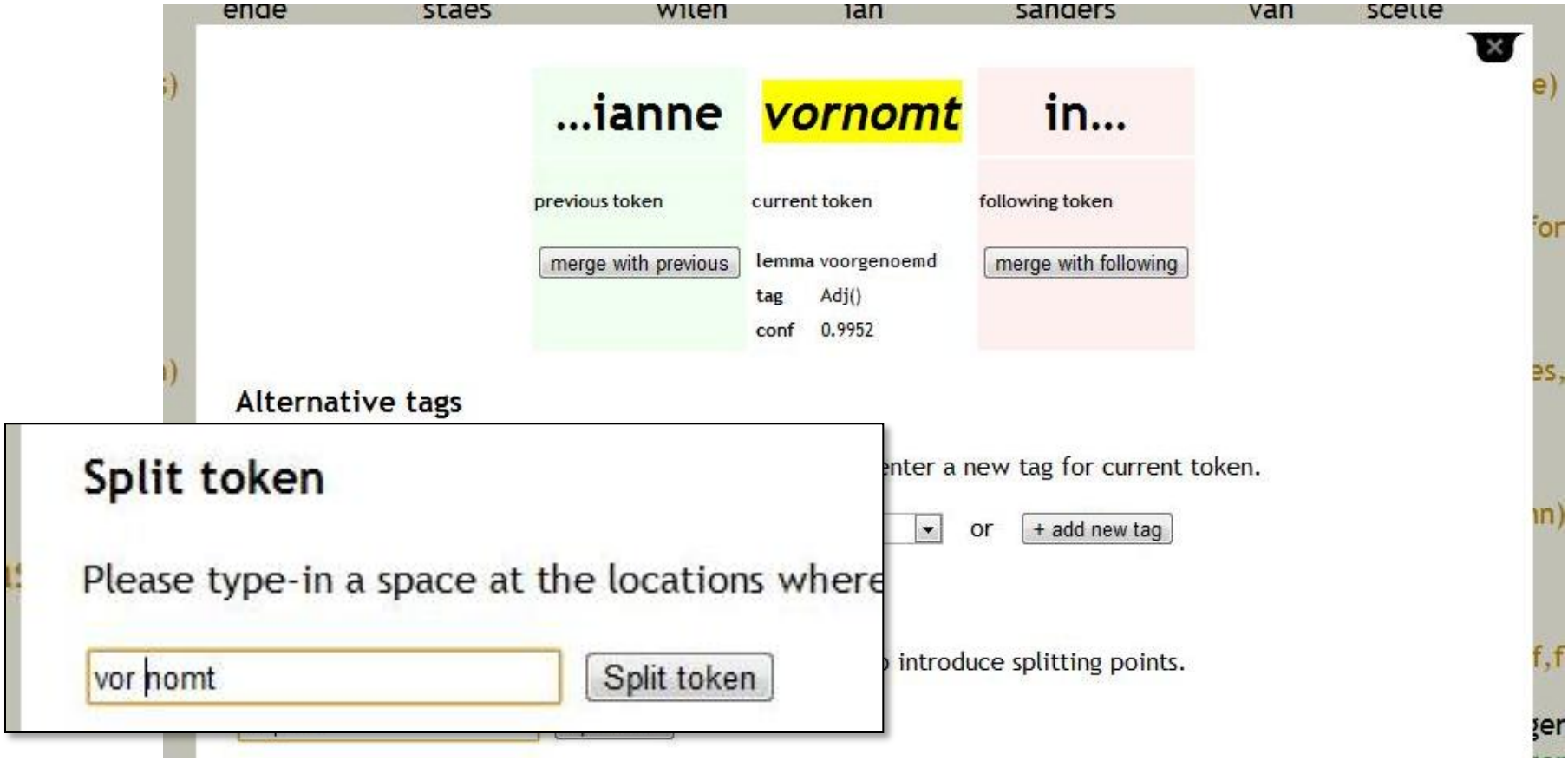
- Entering annotation not suggested by system

Annotation tool: Functionality

- Merging two (or more) tokens

Annotation tool: Functionality

- Splitting tokens into two (or more) parts



Annotation tool: Functionality

- Search for systematic corrections

Authorship Attribution

Historical Whodunits

Authorship Attribution

Definition: Determining which of a number of possible authors wrote a text, based on textual properties

AKA “stylometry”

- 1439: Lorenzo Valla
- 1890: Wincenty Lutosławski
- 1964: Mosteller and Wallace
- Last decades: Also more fundamental

Tasks

Author recognition and verification

- Literary and historical studies
- Forensics; plagiarism recognition

Author profiling

- Determining gender, age, region, psychology
- Evaluating (second) language proficiency

Authorship Attribution

Note there are two different tasks

- Authorship Identification/Recognition
  - Pick from group
  - Evaluation: accuracy
- Authorship Verification
  - Estimate probability for suggested author
  - Evaluation: False Accept/Reject Rate; Equal Error Rate

Authorship Attribution

Foundation: Everyone has their own language ("ideolect")

Task can be divided into

- Measure things that can vary ("features")
- Find identifying features
  - That are reasonably constant for author A
  - And different for other authors
- Show whether text T is by author A
  - Statistically
  - Visually: often in a two-dimensional figure

Use One or Two Features

- Can show as is
- Overall statistics
  - Usually related to vocabulary richness

$$\text{Type}-\text{Token} = V/N$$

$$K = 10^4(\Sigma i^2 V_i - N)/N^2$$

$$R = V/\sqrt{N}$$

$$C = \log V/\log N$$

$$H = (100 \log N)/(1 - V_1/V)$$

$$S = V_2/V$$

$$k = \log V/\log(\log N)$$

$$LN = (1 - V^2)/(V^2 \log N)$$

$$\text{Entropy} = -100\Sigma p_v \log p_v$$

$$\text{W} = N^{V-a}$$

Use One or Two Features

Baayen & Tweedie (1998)

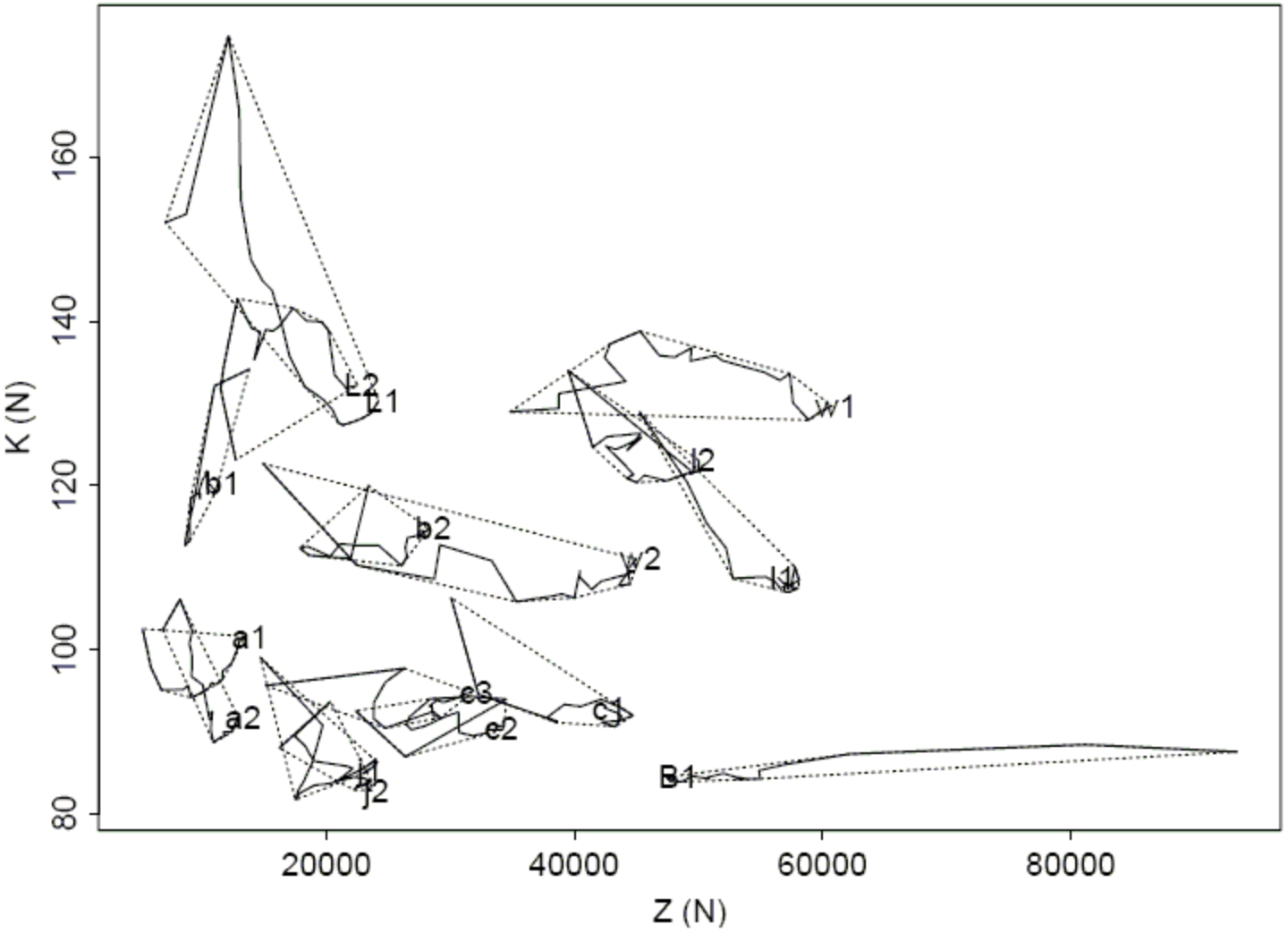


Figure 10. The behaviour of $Z(N)$ and $K(N)$ (solid lines) and their convex hulls (dotted lines) in texts by different authors.

A “Small” Number of Features

- N most frequent (function?) words
  - Once N = 50
  - Later more N = 1000
- N-dimensional vector
- Statistical methods yield division
  - And nice pictures

A “Small” Number of Features

Hoover (2006)

N>1000 (frequent)

PCA

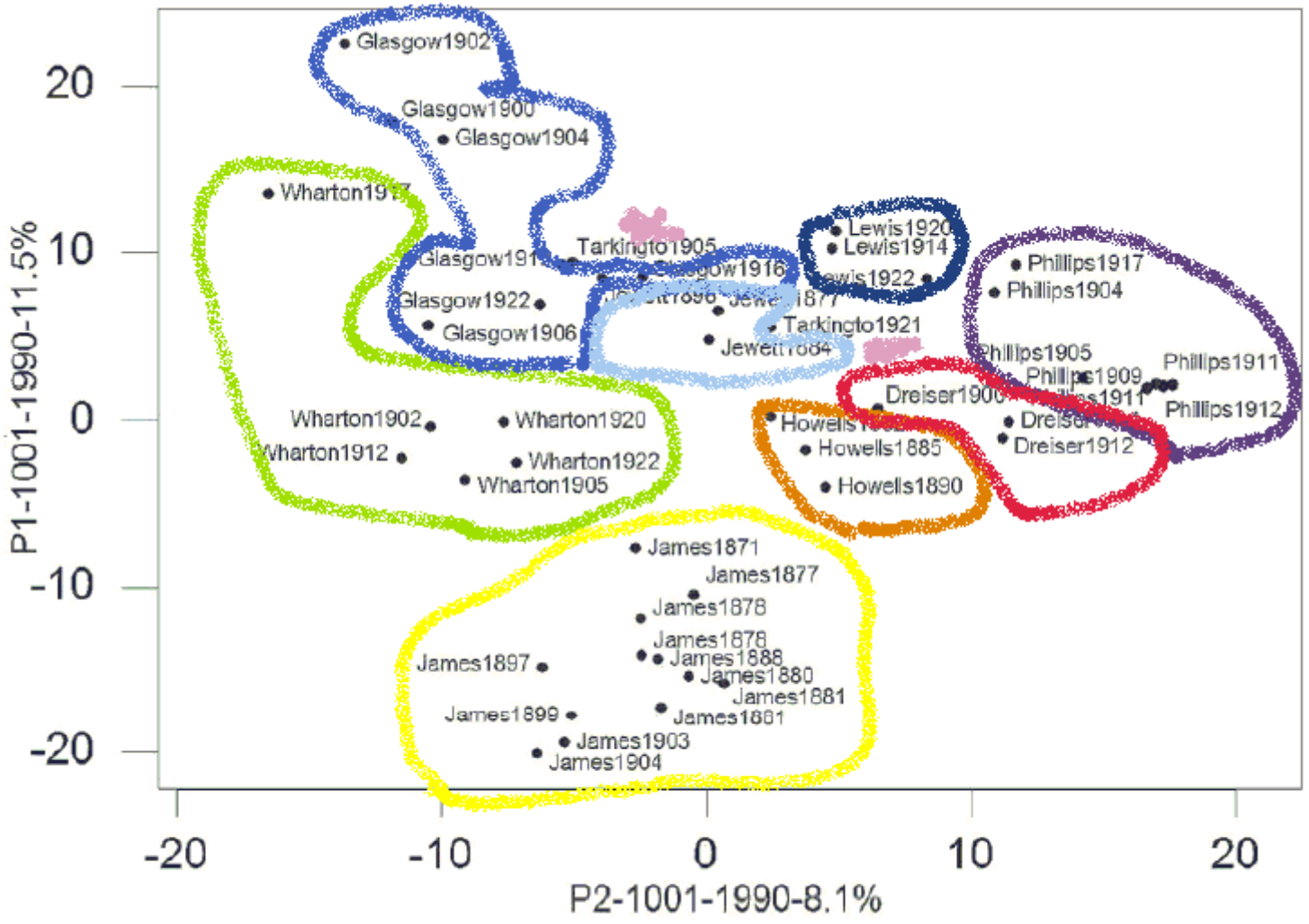


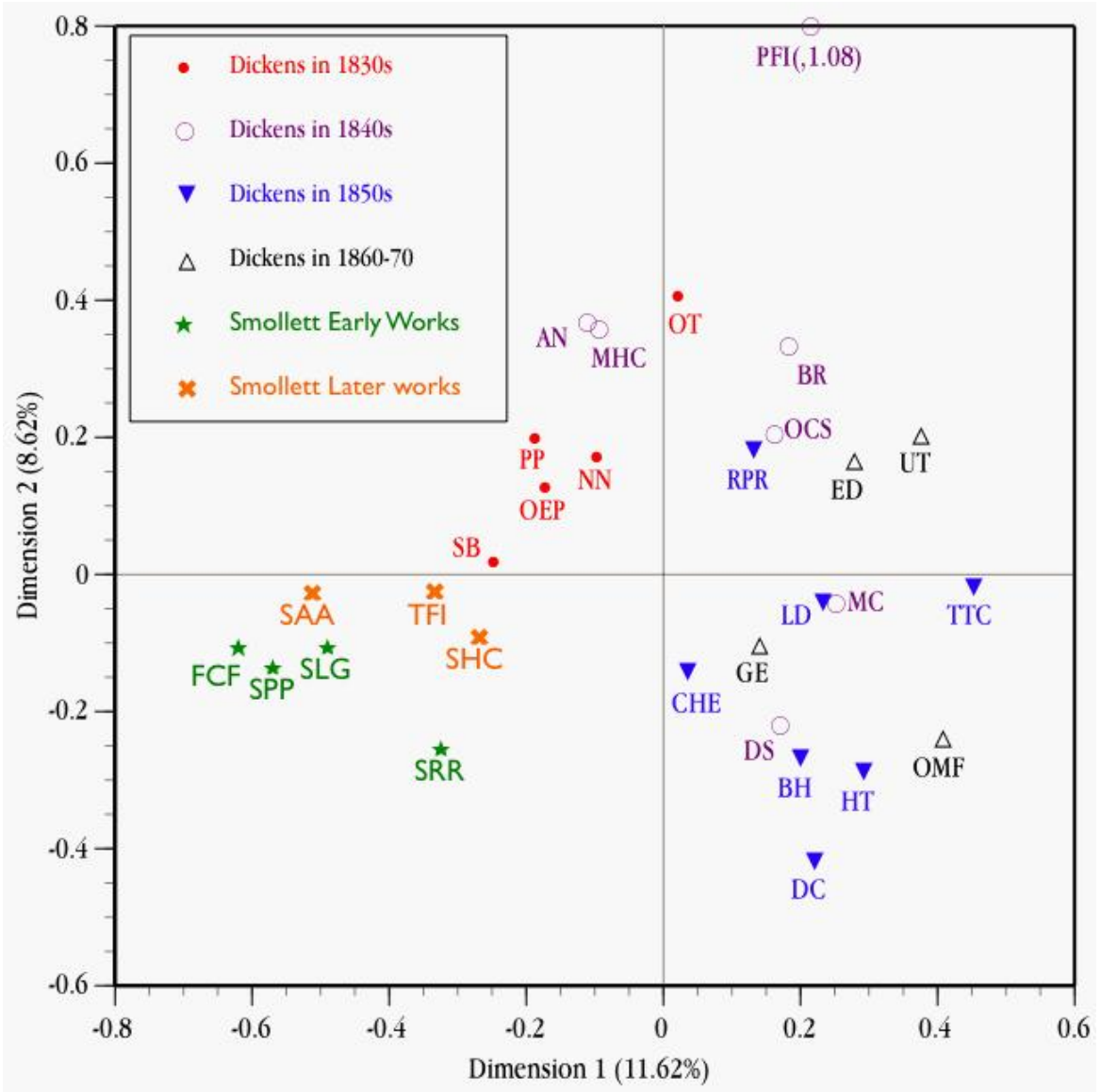
Fig. 1. 46 Novels by Henry James and 8 Other Authors

A “Small” Number of Features

Tabata
(2007)

N=242
(superlatives)

Correspondence
Analysis



Many Features: LP (Linguistic Profiling)

Lexical features

Profile includes counts for:

- sentence lengths
- words / word patterns / word classes
- bi- and trigrams of above
- (single text occurrences filtered out)

Vector of about 100K counts

Counts are:

- normalized for text length
- expressed as relative under- or overuse

LP (Linguistic Profiling)

Syntactic Features

Parse all texts (Amazon parser)
and extract all rewrites

Profile includes counts for:

- LHS label (constituent occurrence)
- LHS-RHS combos (dominance relations)
- LHS-RHS-RHS combos (linear precedence)

Vector of about 900K counts

LP (Linguistic Profiling)

Author profile =
  mean of the profiles for the known texts

Text verification score =
  distance measure text profile to author profile

LP (Linguistic Profiling)

Distance measure:

$$\left(\sum | T_i - P_i |^{D} \; | T_i |^{S} \right)^{1/(D+S)}$$

$$- \left(\sum | T_i |^{(D+S)} \right)^{1/(D+S)}$$

Orthogonalized:

- $\text{Mean}_{(\text{other author texts})}$

/ $\text{StdDev}_{(\text{other author texts})}$

Authorship Attribution Corpus

Corpus:

- 8 students (Dutch)
- 9 texts from each student
  - fixed topics
  - 3 argumentative, 3 descriptive, 3 fiction
  - about 1000 words per text
  - produced in controlled environment

Train: all texts with topic $\neq$ T

Test: all texts with topic T

LP (Linguistic Profiling)

AAC	2-way errors/504	2-way % correct	8-way errors/72	8-way % correct
50 function w., PCA		c. 50%		
+ LDA		c. 60%		
+ entropy weighting		c. 80%		
LEX	6	98.8%	5	93%
SYN	14	98.2%	10	86%
COMB	3	99.4%	2	97%

LP (Linguistic Profiling)

Ad Hoc Authorship Attribution (Juola, ACH/ALLC2004)

Top 5 finishers + Combinations

System	Score	Method
Koppel/Schler	70.6	Unstable words/SVM (++long texts)
Keselj/Cercone	69.0	Byte n-grams/k-NN
Van Halteren	66.2	Word tokens/LP
Juola	65.5	Characters/cross entropy
Coburn	61.8	Word n-grams/graph cuts
Combo5	***71.1***	

LP (Linguistic Profiling)

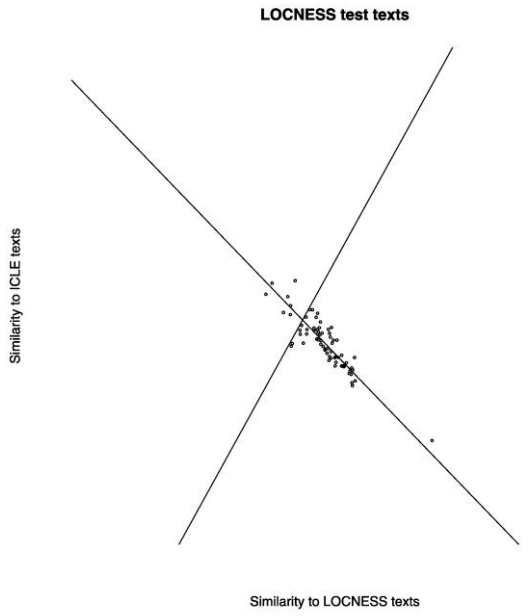
Speaker profiling on (phone dialogues from) CGN

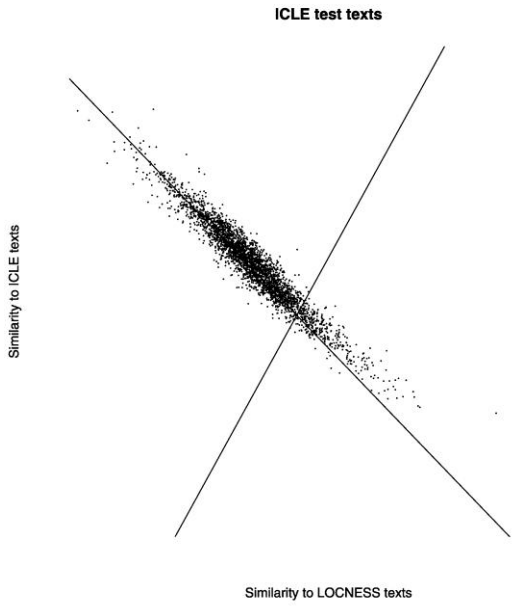
(with Christophe Van Bael)

- Gender ≈ 77%
- Year of birth <> year ≈ 80%
- Regional background varies, effective for 10/16 regions

LP (Linguistic Profiling)

Language Proficiency: LOCNESS vs ICLE





EER around 10%

Back to Medieval

Non-Standardization has advantages too

Attribution of Medieval Texts

New opportunity:

- In addition to vocabulary and syntax, use orthography to base features on

New problem:

- Works for charters
  - Unless scribe ≠ composer
- But not for literature
  - Where copiists tend to introduce changes
  - Reflecting their ideolect

Expectation:

- Exact form marks scribe; abstracted form marks author

Attribution of Medieval Texts

Experiment:

- Charters from the chancellory of the counts of Holland (1300-1340)
  - Studied by a.o. Margit Rem (2003)
- Most prominent seven scribes
  - Previously identified by handwriting
  - Hand 1 = Melis Stoke

Attribution of Medieval Texts

Features

- Vocabulary
  - Unigrams, including exact spelling
- Composition
  - Trigrams, lemmas and/or word classes

Methods

- Linguistic Profiling
- Support Vector Regression

Results Charters

14th Century Dutch Charters

LP and SVR: Choose from 7 authors

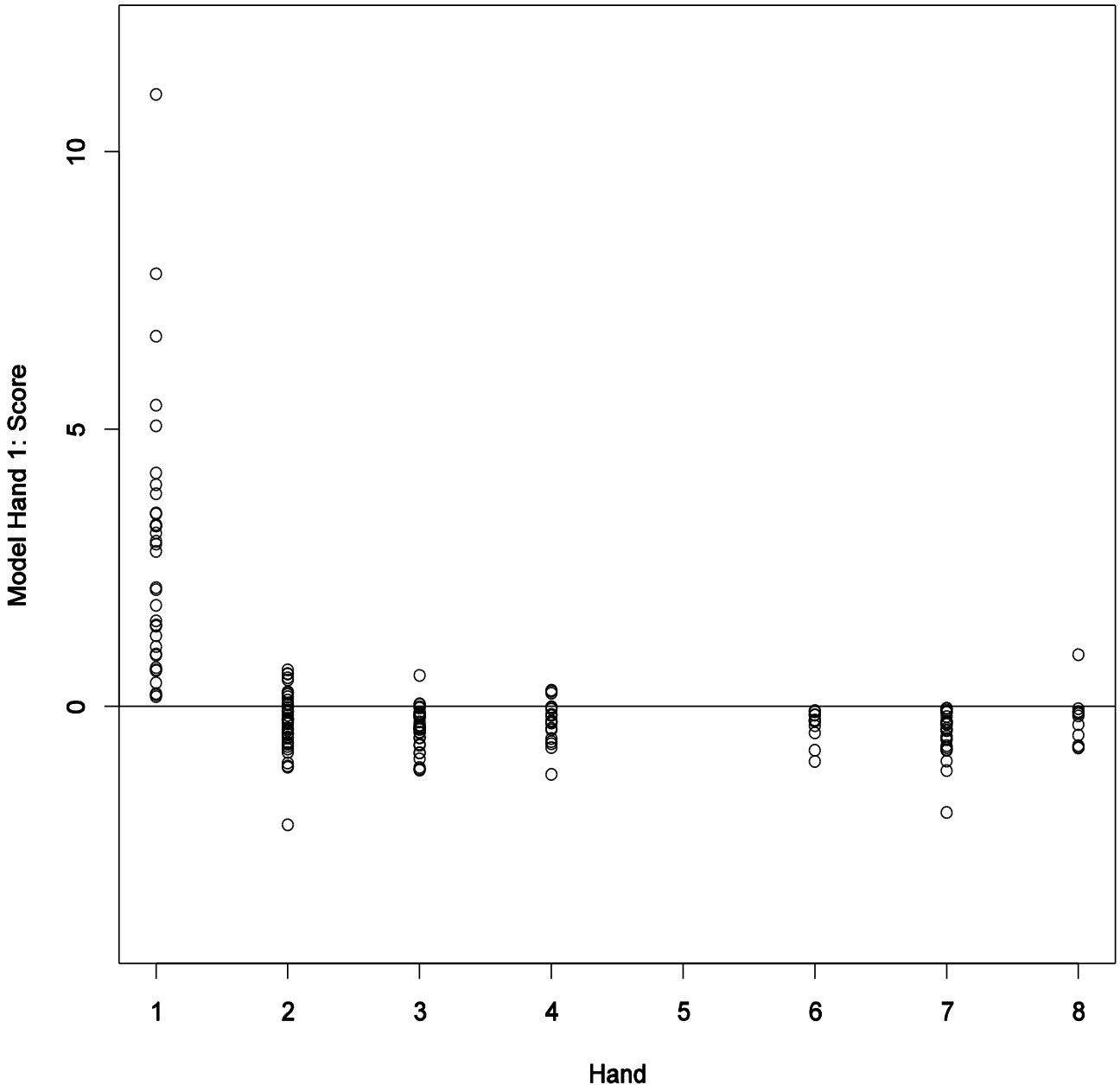
or None-of-These

System	Vocab FRR	Vocab FAR	Compos FRR	Compos FAR
LP	4%	0.2%	17%	1.2%
SVR	4%	0.2%	14%	1.1%
Combo	1%	0.1%	11%	0.9%

Result Charters

Stoke recognizable rather well

Slight overlap



Profiling of Medieval Texts

Experiment:

- Charters from Gysseling and CRM
- Location and Decade Attribution/Verification
- Using LP and kNN
- With trigram and spelling alternation features

- Master Thesis Dieter van Uytvanck (2007)
  - 81 pages; here only a few highlights

Location

Table 6.1: Overview of the FRF-scores for all localisation methods for the 14th century charters.

FRF	trigram, KNN	variants, KNN	rules, KNN	majority vote
Amersfoort	0.03	0.05	0.03	0.03
Amsterdam	0.03	0.03	0.03	0.02
Breda	0.16	0.13	0.19	**0.09**
Brugge	0.09	0.12	**0.03**	0.06
Brussel	0.21	0.43	**0.12**	0.28
Delft	0.09	0.11	**0.03**	0.06
Deventer	0.03	0.08	0.05	0.04
Dordrecht	**0.04**	0.11	0.10	0.07
Eersel	**0.00**	0.54	0.05	0.05
Egmond-Binnen	**0.33**	0.82	0.67	0.82
Gemert	**0.18**	0.33	0.33	0.33
Gent	**0.18**	0.54	0.38	0.33
Gouda	0.02	0.02	0.01	0.01
Groningen	0.02	0.08	0.03	0.03
Haarlem	0.13	0.14	**0.07**	0.1
Halen	0.11	0.25	**0.05**	0.11
Hasselt	0.43	0.43	**0.18**	0.43
Helmond	0.03	0.15	0.04	0.05

Location: Gouda

Trigrams:

1. rfr [relief-score: 0.594]: erfrenten (84), erfrecht (22), eyrfrogghen (1), erfrogghen (1), erfrogghe (1)
2. fre [0.331]: erfrenten (84), erfrecht (22), frederic (20), frederikes (5), lijfrente (4)
3. yed [0.327): lyede (88), lyeden (14), belyede (14), verlyeden (10), meyedach (8)
4. lye [0.317]: lyede (88), lyen (18), lyeden (14), lye (14), belyede (14)
5. ork [0.250]: orkonden (157), orkonde (95), orkunde (80), oorkonden (18), sporkele (15)
6. gou [0.246]: goude (246), vergouden (35), gouden (35), gouuerneerres (9), gouts (8)

Location: Gouda

Variants:

1. *lyede* [relief-score: 0.8303]: "to execute"
2. *orkonden* [0.8293]: "to publish with a charter"
3. *kennen* [0.7949]: "to know"
4. *m* [0.6730]: abbreviation of *met* ("with"), *men* ("(some)one") or *maar* ("but")[4]
5. *panden* [0.5686]: "to confiscate"[5]

Location: Gouda

Variation rules:

1. d → nd [0.7342]
2. rc → rk [0.7096]
3. lls → lls (not: llns/llnts/llts/ll) [0.5191]
4. ll → lls [0.5191]
5. ye → ye (not: ei/ie/ii/ij/ue/ey/ee/je/y/e/j/i/ije) [0.4951]

Location: Clustering

Figure 5.3: Some clusters that were found using the spelling rules features for the 14th century locations.

Period: Decades

Table 4.3: Confusion matrix for the decade (ranging from 1230 to 1400, indicated by the first 3 numbers) classifier, using KNN and the Z-score of the relative trigram frequencies. The leftmost column indicates the real class, the header row shows the predicted class.

	123	124	125	126	127	128	129	130	131	132	133	134	135	136	137	138	139	140
123	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
124	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
125	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
126	0	0	0	15	17	16	7	0	0	0	0	0	2	1	0	1	0	0
127	1	0	0	4	75	64	33	1	1	0	0	0	0	1	1	1	0	0
128	1	0	1	0	19	392	137	3	4	0	0	1	1	1	0	0	1	1
129	2	0	0	2	21	124	645	9	18	1	5	4	5	2	5	2	1	0
130	1	0	0	0	2	13	32	42	13	2	3	3	4	2	1	0	1	0
131	0	0	0	0	1	2	14	4	35	5	5	8	5	2	1	2	1	0
132	0	0	0	0	0	2	14	1	16	180	3	4	3	5	4	0	2	0
133	0	0	0	0	0	3	15	2	13	4	59	21	13	9	5	3	4	0
134	1	0	0	0	2	7	14	0	16	2	12	104	20	22	10	9	9	0
135	0	0	0	0	1	4	11	0	7	2	3	25	148	33	17	17	16	2
136	1	0	0	0	1	5	9	0	9	1	4	17	31	166	35	34	33	0
137	0	0	1	0	0	2	8	2	8	0	0	9	17	40	176	71	57	1
138	1	0	0	0	2	3	4	0	10	2	1	9	18	16	57	210	112	1
139	1	0	0	0	2	10	14	1	4	0	1	9	13	16	28	82	419	14
140	0	0	0	0	0	0	2	0	4	0	0	2	0	2	0	12	20	11

Problem: Both corpora sampling biased: more at end of century

Period: Time Slices

Alternative:

Slices containing

100 charters

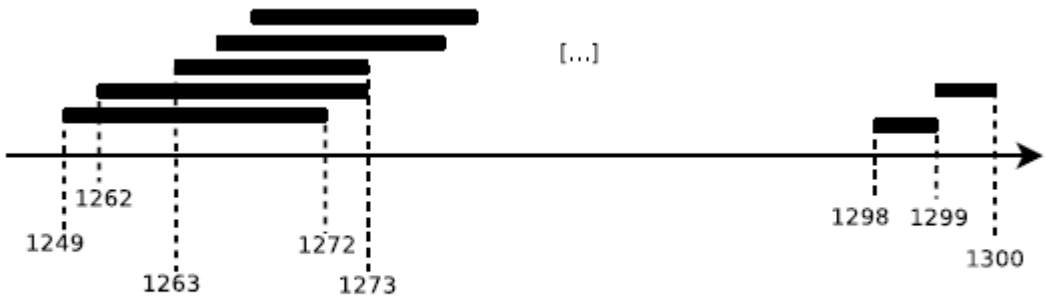


Figure 4.3: Sliding windows as dating classes.

(b) orthographic rules as features

Figure 5.4: Confusion matrix for the year interval verification in the 14th century material, using KNN.

Period: Time Slices

Table 6.2: Comparison between interval verification methods for the 14th century (for a sample). The average has been calculated on all test data, except for the trigram & KNN combination.

interval	trigram, KNN	trigram, LProf	variants, KNN	rules, KNN
1300-1314	0.35	0.21	0.59	0.33
1310-1322	0.35	0.23	0.60	0.37
1320-1332	0.59	0.27	0.68	0.40
1330-1336	0.43	0.36	0.56	0.41
1340-1345	0.40	0.31	0.54	0.52
1350-1353	0.53	0.29	0.54	0.47
1360-1363	0.53	0.31	0.64	0.58
1370-1372	0.53	0.39	0.58	0.50
1380-1382	0.63	0.36	0.76	0.61
1390-1391	0.60	0.39	0.68	0.65
Average	0.49	0.32	0.61	0.49

Period much more difficult than location

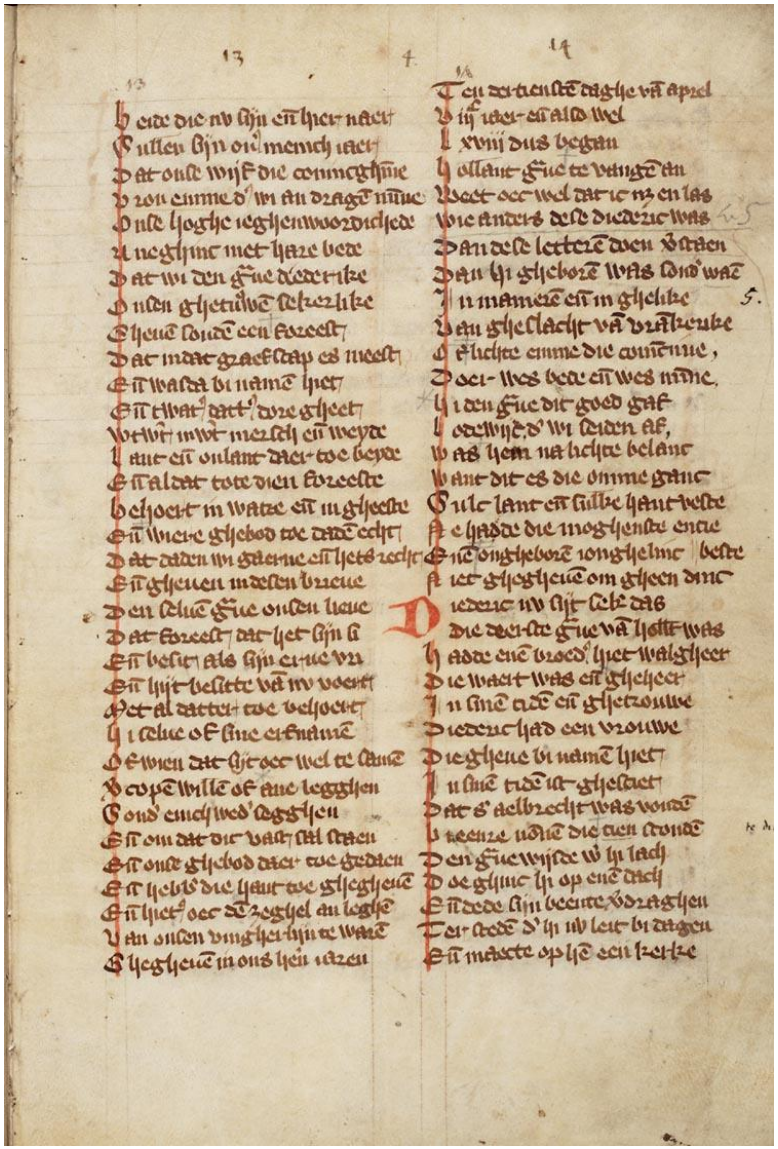
- But improves (somewhat) when restricted in location (Brugge)

Rijmkroniek

Find the break:

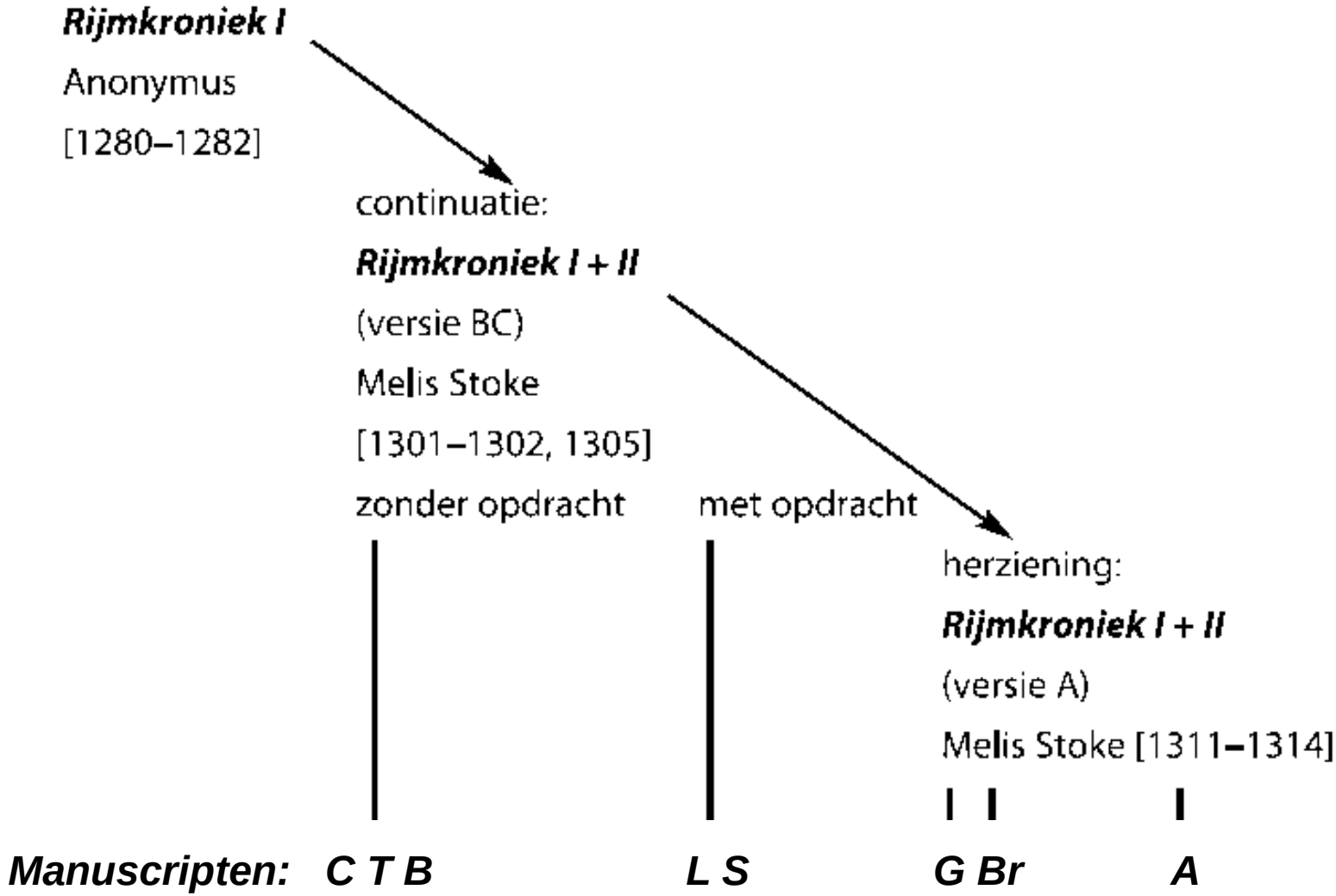
What happens with smaller chunks?

Rijmkroniek van Holland (366-1305)



Het gheuel als hi ghedochte
Ten eersten datmen riden mochte
Ouer ijs ende ouer al
Gheboet hi ende beual
Datmen ghebode hereuaert
Ende hi trecte darwaert
Jnden winter op enen dach
Na kersauonde als hi ghelach
Jn eene vorst als god woude
Ende oec emmer gheuallen soude
Die coninc voer al tot alcmare
Ende te vronen aldaer nare
Ende al op den groten yse
Die coninc stout van houghen prise

Rijmkroniek van Holland (366-1305)



Rijmkroniek van Holland (366-1305)

Question: Who wrote what?

- Studied before, especially by Jan Burgers

Used for experiment:

- Manuscript C (1390), by one copiist
  - Mattheus Gerardszoon
  - Not copied straight from Stoke
- Machine readable
  - Transcribed by Jan Burgers
  - Abbreviations interpreted

Method Rijmkroniek

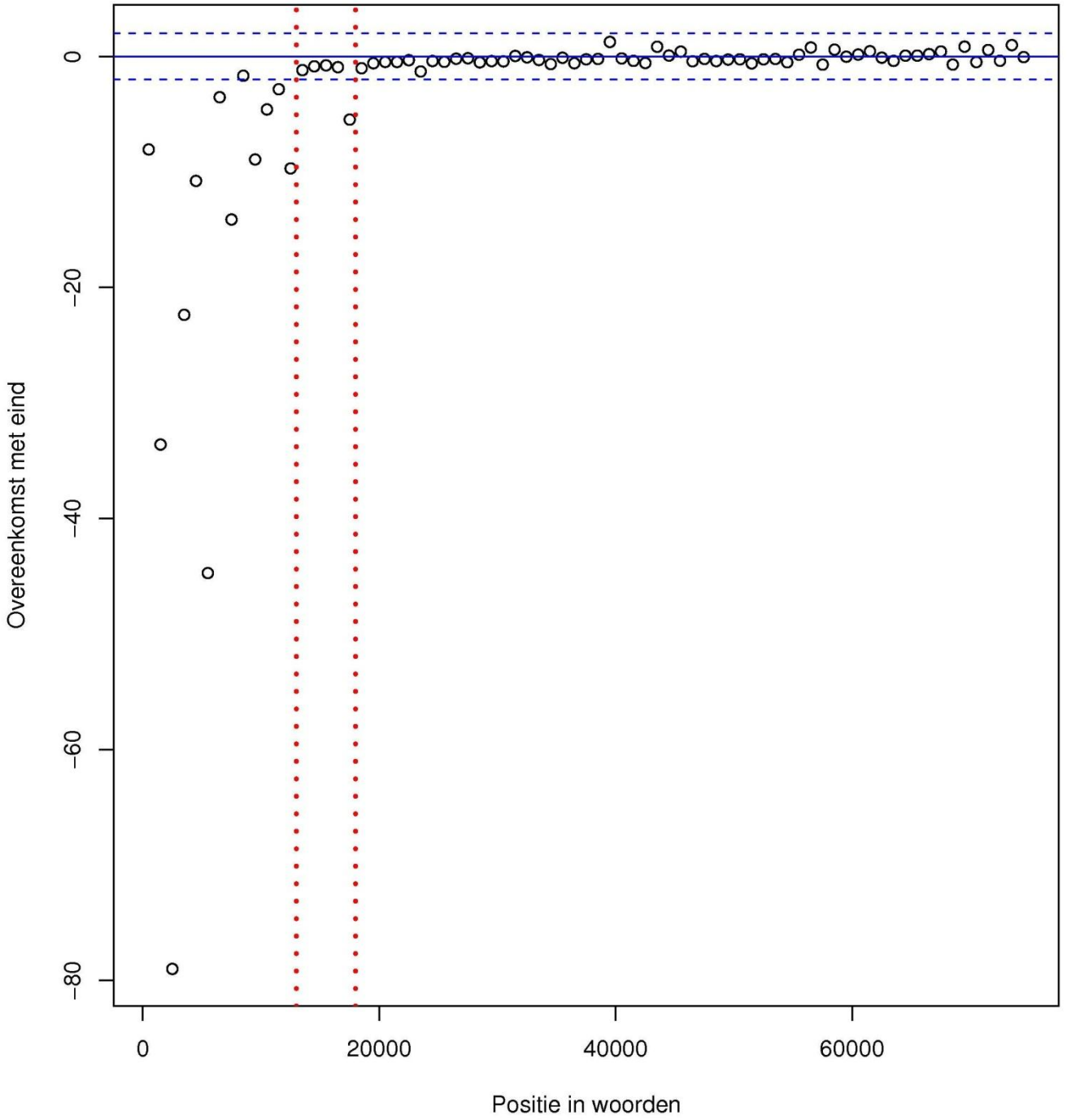
Using Support Vector Regression

Features

- Combinations as described above
- But only lemmas and word class tags
  - Ignore lemmas that occur only locally
- Compare within Rijmkroniek
  - For sure Stoke: the end (word 45000-75000)
  - For sure not (only) Stoke: the start (500-8500)

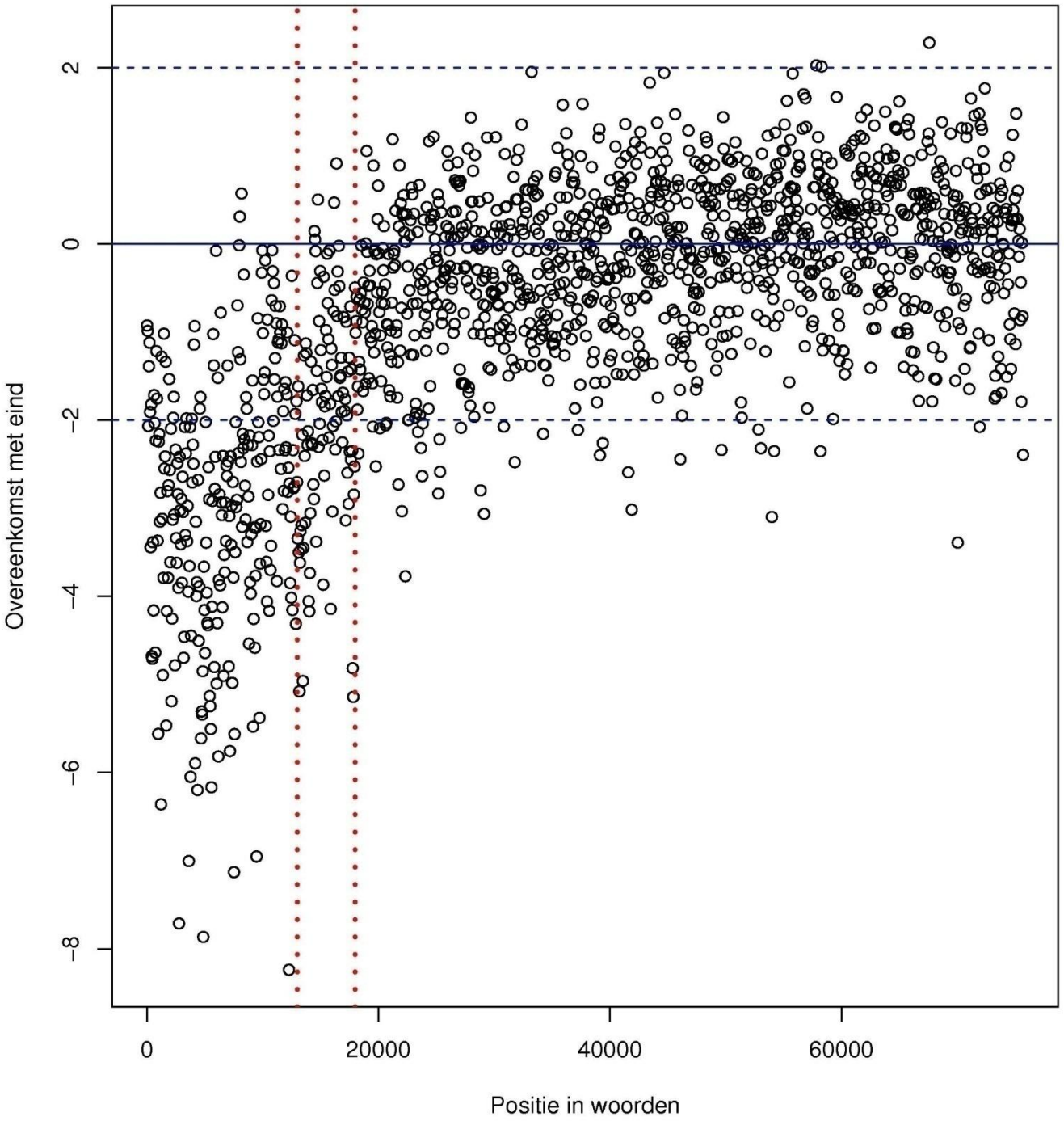
Results Rijmkroniek

1000 words



Results Rijmkroniek

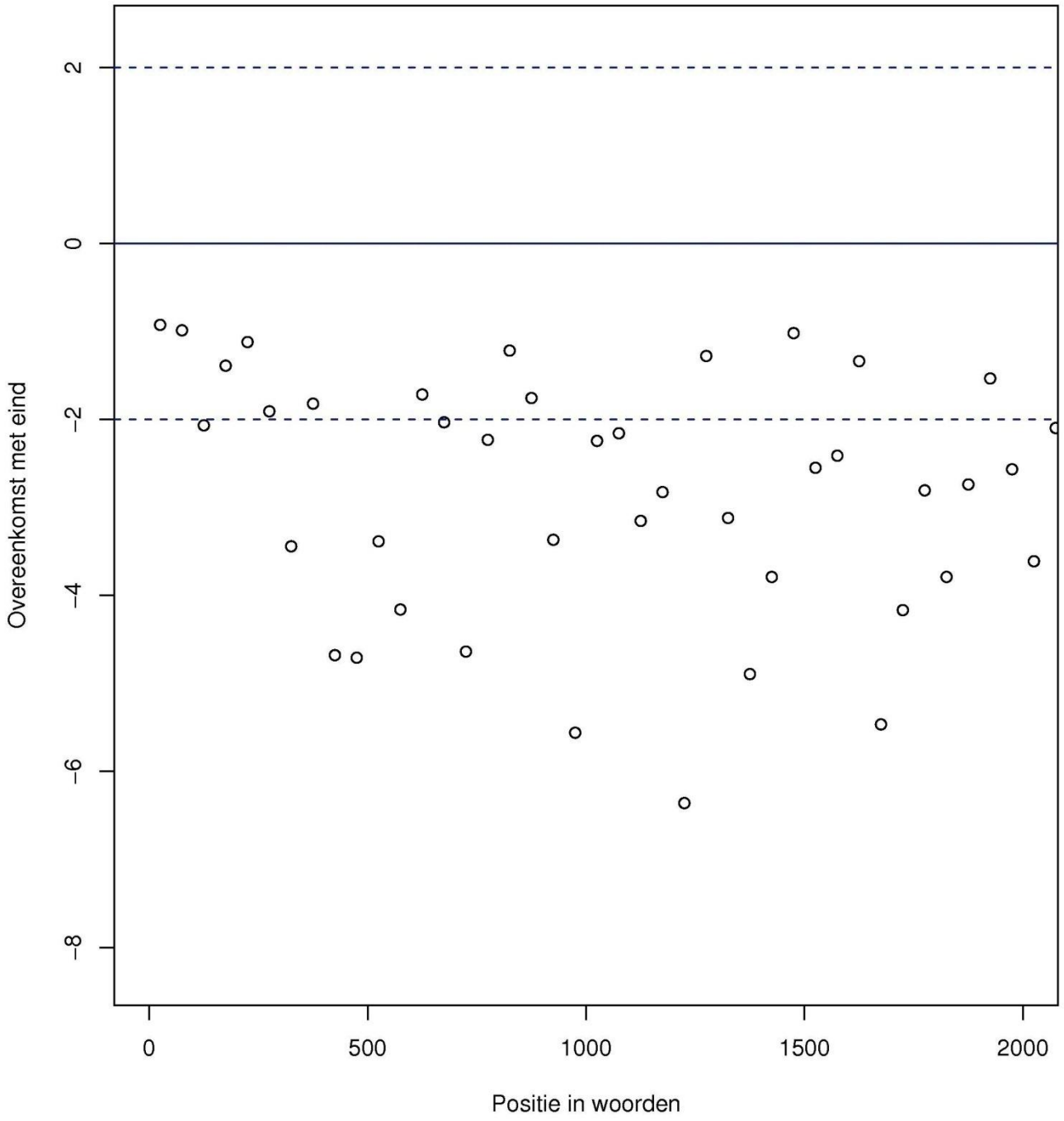
50 words



Results Rijmkroniek

50 words

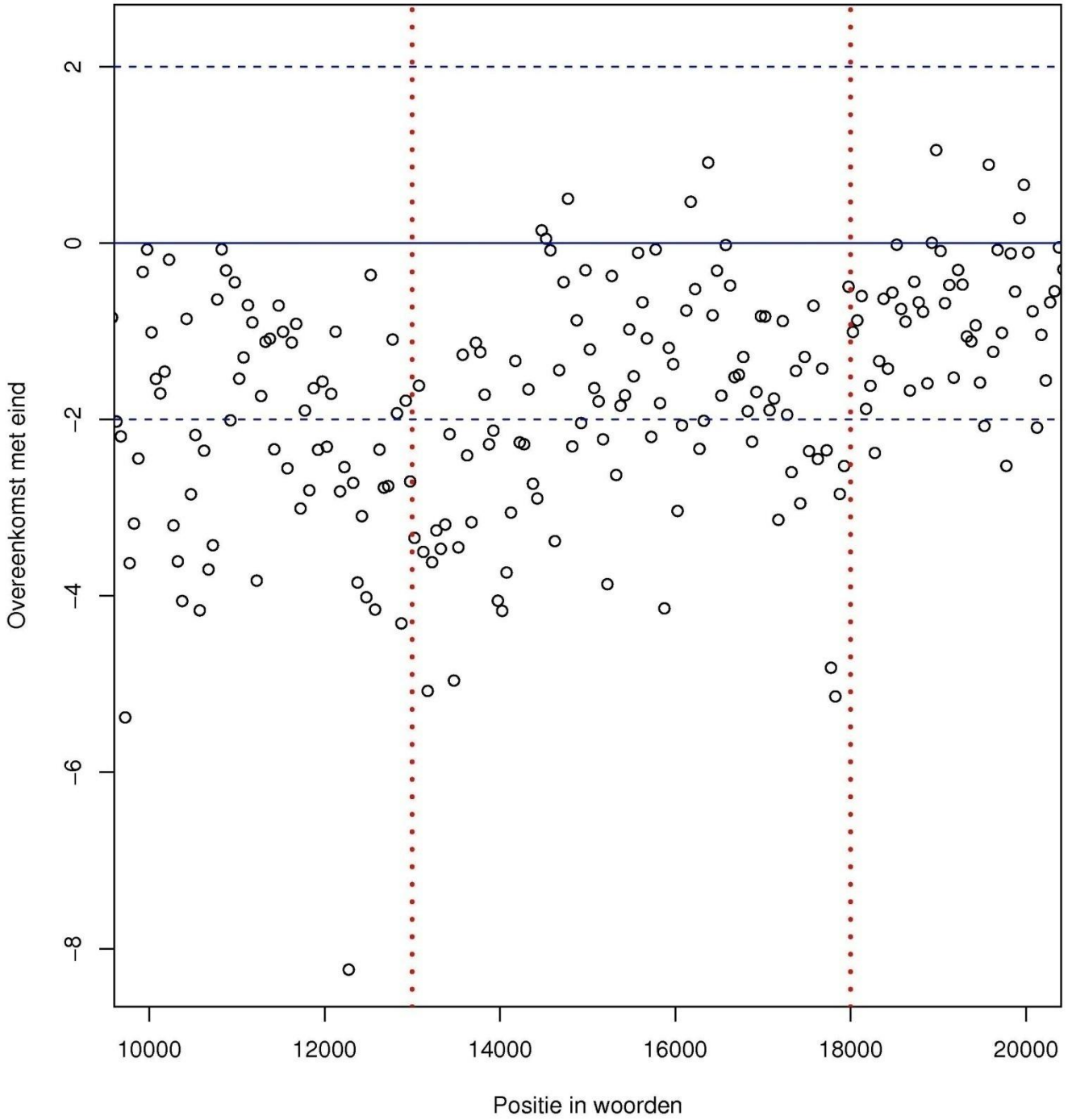
Beginning



Results Rijmkroniek

50 words

Break



Results Rijmkroniek

- Conclusion
  - Break around word 17900
  - But Stoke adjusted a lot before that point too
  - And almost certainly rewrote the beginning

Results Rijmkroniek

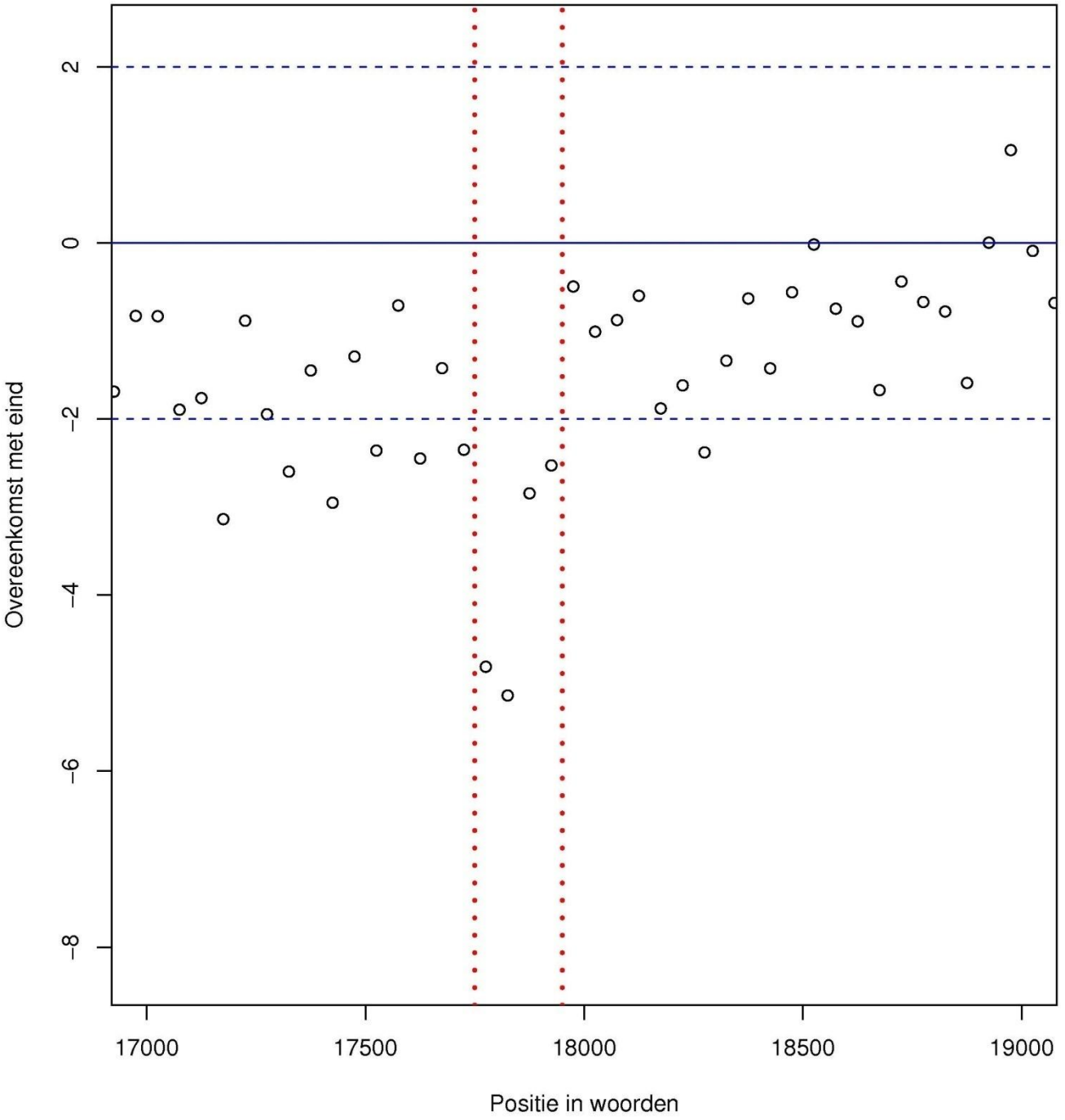
- Conclusion
  - Break around word 17900
  - But Stoke adjusted a lot before that point too
  - And almost certainly rewrote the beginning

- Burgers says
  - Anonymous author stops “on or around verse 579 of the third book”

Results Rijmkroniek

50 woorden

Break



Results Rijmkroniek

Same (more accurate?) result

- Knowledge rich
  - Vocabulary, expressions, fixed phrases (“stoplappen”), rhyme words, deviant syntaxis, text structure
  - Tendency to repeat, moralising deliberations, directly addressing listeners
- Knowledge poor
  - Lemma/wordclass n-grams

Rijmkroniek: Future

More attention for the text!

- No 50 word blocks, but verses
- More features
  - Countable things that Burgers mentions
- See what characterizes Stoke
  - NB Difficult when using SVR

- Examine where Stoke made changes