Adventures in Annotation Alignment and Error Detection

Adriane Boyd

Universität Tübingen

22 January 2018

Disclaimer

- Grateful for tools and resources
- Intend no disparagement of any particular project
- Hope that our work can help resources improve

Adventures in Alignment

Connection between original data and linguistic annotation is frequently not maintained in the annotation process.

PTB Normalization and Tokenization

Source: Fulton Prebon (U.S.A.) Inc.

↓

`Source : Fulton Prebon -LRB- U.S.A . -RRB- Inc .`

Adventures in Alignment

- NLP Tools: Tokenization
- User Needs: MERLIN

Tokenization: Introduction

Question: How do I choose a tokenizer?

- Can I find any documentation or guidelines?
- How do I know whether a tokenizer works well with models further down the pipeline? (cf. Eckart de Castilho 2016)
- What can I do when a tokenizer doesn't perform well?
  - Where can I find data to train a tokenizer?
  - How much data do I need?

Tokenization: Considerations

- Encoding: UTF-8 vs. ISO-8859-1
- Normalization:
  - Punctuation:
    - *``Doubled'' "ASCII" “Unicode”*
    - *-LRB- PTB-style parentheses -RRB-*
  - Whitespace within tokens:
    - *(201)␣555-1234*
    - *out_of*
- Tokenization conventions:
  - *can’t → ca n’t*
  - *U.S. → U.S. .*

(cf. Dridan & Oepen 2012; Eckart de Castilho 2016)

Tokenization: Data for German

Very little raw data (readily) available for German:

- EmpiriST (Shared Task 2015): CMC, web data (Beißwenger et al. 2016)
  - ca. 25,000 tokens with training + test data
  - annotation guidelines

Use of detokenized data:

- Jurish & Würzner (2013): report using detokenized TIGER with some manual corrections
- de Kok (2014): report using detokenized TüBa-D/Z with OpenNLP detokenizer (rule-based)

Tokenization: Re-Aligning Tokenized Corpora

Re-Alignment

Tokenized: `( vom 18. - 20. Juni )`
Original: *(vom 18.-20. Juni)*

Tokenization: Re-Aligning Tokenized Corpora

Re-Alignment

Tokenized: `( vom 18. - 20. Juni )`
Original: *(vom 18.-20. Juni)*

Tokenization: Re-Aligning Tokenized Corpora

Re-Alignment

Tokenized: `( vom 18. - 20. Juni )`
Original: *(vom 18.-20. Juni)*

Tokenization: Re-Aligning Tokenized Corpora

Re-Alignment

Tokenized: `( vom 18. - 20. Juni )`
Original: *(vom 18.-20. Juni)*

Tokenization: Re-Aligning Tokenized Corpora

Re-Alignment

Tokenized: `( vom 18. - 20. Juni )`
Original: *(vom 18.-20. Juni)*

TüBa-D/Z (version 10.0: 1,787,801 tokens)

- Raw texts from taz are available
- Tokenization in TüBa-D/Z treebank annotation
- Metadata links each sentence to taz article

Tokenization: TüBa-D/Z - taz Alignment

Assuming that tokenization only inserts boundaries:

- 98.2% of sentences can be aligned easily on the character level

Tokenization: TüBa-D/Z - taz Alignment

Assuming that tokenization only inserts boundaries:

- 98.2% of sentences can be aligned easily on the character level
- Remaining 1.8%?
  - Artifacts of newspaper format: ambiguous hyphenation
    Ost- Berlin → *Ost-Berlin*
    vs.
    An- und Abreise

    ver- spannte → *ver-spannte* (vs. *verspannte*)
  - Symbols: *niklaus§taz.de* → *niklaus@taz.de*
  - Comma quotes: `(,,)` → `ASCII "`
  - Emphasis:
    D O P P E L P O R T R A I T → *DOPPELPORTRAIT*
  - Other minor corrections/normalizations

Tokenization: Original vs. Detokenization

Comparing original vs detokenized texts:

- 5.5% of sentences have different tokenization

Trained tokenizer models using OpenNLP with original vs. detokenized (90/10 split):

		Model	
		Orig	Detok
Test Data	Orig	99.92	99.77
	Detok	99.94	99.95

Tokenization: Original vs. Detokenization

Comparing original vs detokenized texts:

- 5.5% of sentences have different tokenization

Trained tokenizer models using OpenNLP with original vs. detokenized (90/10 split):

		Model	
		Orig	Detok
Test Data	Orig	99.92	99.77
	Detok	99.94	99.95

- Detokenized model: many false negatives

Tokenization: Orig. vs. Detok. Non-Whitespace

Evaluating non-whitespace tokenization:

	Orig	Detok
Precision	99.30	98.93
Recall	99.34	96.65
F1	99.32	97.78

Tokenization: Annotated Data Uses

- Evaluate and compare tokenization approaches
- Customization of new models, e.g.:
  - without newspaper headlines
  - with custom date handling

Tokenization: Summary

Currently difficult to:

- Find tokenization annotation guidelines
- Determine how available models were trained
- Find data to evaluate and train tools

Tokenization: Summary

Currently difficult to:

- Find tokenization annotation guidelines
- Determine how available models were trained
- Find data to evaluate and train tools

Recommendations:

- Document and publish tokenization guidelines
- Prefer annotation tools and formats that preserve alignments with original data

Tokenization: Summary

Currently difficult to:

- Find tokenization annotation guidelines
- Determine how available models were trained
- Find data to evaluate and train tools

Recommendations:

- Document and publish tokenization guidelines
- Prefer annotation tools and formats that preserve alignments with original data

Would like to have a tokenizer evaluation tool that:

- Compares a set of available tokenizers on a test corpus
- Shows learning curves for tokenizer training

MERLIN

Illustrates CEFR scales levels in a written learner corpus for Czech, German, and Italian in a didactically-motivated online platform

MERLIN Team

University of Technology Dresden (coordination)

Katrin Wisniewski, Maria Lieber, Claudia Woldt, Karin Schöne

European Academy Bozen

Andrea Abel, Verena Blaschitz, Verena Lyding, Lionel Nicolas, Chiara Vettori

Charles University Prague

Kateřina Vodičková, Pavel Pečený, Jirka Hana, Veronika Čurdová

telc Frankfurt/Main

Sybille Plassmann, Gudrun Klein, Louise Lauppe

Berufsförderungsinstitut Oberösterreich, Linz

Gerhard Zahrer, Pia Zaller

Eberhard Karls University Tübingen

Detmar Meurers, Adriane Boyd, Serhiy Bykh, Julia Krivanek

MERLIN Corpus

- Approx. 200 texts per CEFR level
  - Czech (A2–B2): 441 texts
  - German (A1–C1): 1033 texts
  - Italian (A1–B2): 813 texts
- Detailed re-ratings:
  - overall
  - orthography
  - grammatical accuracy
  - vocabulary range
  - vocabulary control
  - coherence & cohesion
  - sociolinguistic appropriateness
- Learner metadata
- Task descriptions

MERLIN Annotations

Manual

- transcription: task citations, greetings, closings, . . .
- target hypotheses (normalization)
- error annotation

Automatic

- tokens, sentences
- lemmas, POS
- dependency parses
- repetitions within texts

Derived

- statistical measures for error annotation
  e.g., word order errors per token

MERLIN Platform

- Target audiences
  - language teachers
  - test and curriculum developers
  - textbook authors
  - (computational) linguists
- Search engines
  - simple (solr): KWIC, formatted full texts, metadata
  - advanced (ANNIS): full TH/EA, automatic annotations, metadata

MERLIN: Simple Search Results

Key word in context

Kind krank und es **bett** bleiben muss Darum Ich

View learner info and ratings

learner text:

Liebe Julia, 12.03.2012
wie geht´s? Mir geht es gut. Ich habe eine Fahrkarte für einen Wochenendausflug gekauft. Aber Ich kann nicht fahren. Meine Kind krank und es bett bleiben muss Darum Ich kann nicht fahren. Die Fahrkarte gekauft am 16 März. Möchtest du die Fahrkarte? Rufst du diene Antwort an. Vielen Dank Julia. Bis bald
Alles Liebe
deine
Eva

target hypothesis 1:

Liebe Julia , 12.03.2012 wie geht´s ? Mir geht es gut . Ich habe eine Fahrkarte für einen Wochenendausflug gekauft . Aber ich kann nicht fahren . Mein Kind ist krank und es muss im Bett bleiben . Darum kann ich nicht fahren . Die Fahrkarte wurde am 16 . März gekauft . Möchtest du die Fahrkarte ? Rufst du deine Antwort an ? Vielen Dank , Julia . Bis bald . Alles Liebe deine Eva

MERLIN: Advanced Search Results

1 Path: MERLIN_German > 1091_0000272 (learner 1 - 62) left context: 5 right context:

TH/EA (grid)

Meine	Kind		krank	und	es	bett	bleiben	muss				
Mein	Kind	ist	krank	und	es			muss	im	Bett	bleiben	.
CHA		INS				MOVS	MOVS		INS	MOVT/CHA	MOVT	INS
Mein	Kind	ist	krank	und	es			muss	im	Bett	bleiben	.
CHA		INS				MOVS	MOVS		INS	MOVT/CHA	MOVT	INS
						G_Prep	G_Verb_compl					
G_Verb_compl					O_Punct							
						O_Capit						
G_Morphol_Wrong					G_Wo							
ambig												
						o						
o							pos					
					womaincl							
					o							

MERLIN: Annotation Pipeline

Format	Tool	Annotation
hand-written scan		
custom XML	XMLmind	transcription
PAULA	custom converter	tokens, sentences
Exmaralda XML	SaltNPepper	
	Exmaralda/Falko Excel add-ins	TH1
PAULA	custom converter	
MMAX2	SaltNPepper	EA1
PAULA	SaltNPepper	

MERLIN: Annotation Pipeline, cont.

Format	Tool	Annotation
Exmaralda XML	SaltNPepper (improved)	
	Exmaralda/Falko Excel add-ins	TH2
PAULA	custom converter	
MMAX2	SaltNPepper	EA2
PAULA	SaltNPepper	
PAULA	custom UIMA pipeline	automatic
→ solr	custom converter	
→ ANNIS	SaltNPepper	

MERLIN: Problematic Conversions

- PAULA → Exmaralda XML (SaltNPepper)
  - Tokens only, whitespace/formatting is lost

Liebe Julia, 12.03.2012

wie geht´s? Mir geht es gut. Ich habe eine Fahrkarte für einen

	A	H
1	**tok**	**ctok**
2	Liebe	Liebe
3	Julia	Julia
4	,	,
5	12.03.2012	12.03.2012
6	wie	wie
7	geht	
8	´	
9	s	geht´s

MERLIN: Complicating Factors

One annotation step uses Excel, where annotators can (and do!) edit almost anything

- Advantage: annotators can potentially edit transcription or annotations
- Disadvantage: annotator can accidentally edit transcription or annotations

⇒ Re-aligning raw data and annotation is complicated

MERLIN: Cascading Errors

7	geht	
8	´	
9	s	geht´s

- MERLIN tokenization guidelines: *geht´s* is one token
- Cascading errors in pipeline of standard German NLP tools:
  - STTS tagset has no tag for this contraction

MERLIN: Cascading Errors

7	geht	
8	´	
9	s	geht´s

- MERLIN tokenization guidelines: *geht´s* is one token
- Cascading errors in pipeline of standard German NLP tools:
  - STTS tagset has no tag for this contraction

learner	wie	geht´s	dir	und	wie	geht´s	deine	Eltern
tok_lemma	wie	geht´s	du	und	wie	geht´s	dein	Eltern
tok_lemma_bohnet	wie	geht´	dir	und	wie	geht´	dein	Eltern
tok_pos	KOUS	VVFIN	PRF	KON	KOUS	VVFIN	PPOSAT	NN
tok_pos_bohnet	PWAV	VVFIN	PPER	KON	KOKOM	ADV	PPOSAT	NN
tok_pos_stanford	PWAV	VVFIN	PRF	KON	KOKOM	NN	PPOSAT	NN

MERLIN: Summary

- User needs may require annotations to be aligned with original formatted texts
- Maintaining whitespace / formatting in a long pipeline is difficult
  - Every single tool needs to support, e.g., character offsets

From Alignment to Error Detection

Alignment between raw data and annotation is crucial for certain tools and use cases:

Aligned Tokens

Tokenized: `( vom 18. - 20. Juni )`
Original: *(vom 18.-20. Juni)*

From Alignment to Error Detection

Alignment between raw data and annotation is crucial for certain tools and use cases:

Aligned Tokens

Tokenized: (vom 18. - 20. Juni)
Original: (vom 18.-20. Juni)

From Alignment to Error Detection

Alignment between raw data and annotation is crucial for certain tools and use cases:

Aligned Tokens

Tokenized: `( vom 18. - 20. Juni )`
Original: *(vom 18.-20. Juni)*

From Alignment to Error Detection

Alignment between raw data and annotation is crucial for certain tools and use cases:

Aligned Tokens

Tokenized: `( vom 18. - 20. Juni )`
Original: *(vom 18.-20. Juni)*

From Alignment to Error Detection

Alignment between raw data and annotation is crucial for certain tools and use cases:

Aligned Tokens

Tokenized: *(vom 18. - 20. Juni)*
Original: *(vom 18.-20. Juni)*

What about the quality of the annotation itself?

Annotated Tokens

Tokenized:	*(*	*vom*	*18.*	*-*	*20.*	*Juni*	*)*
POS Tags:	$(	APPRART	ADJA	APPR KON	ADJA	NN	$(

Error Detection: Introduction

Annotated corpora are used:

- To train and test NLP technology
- For searching for linguistically relevant patterns

Error Detection: Introduction

Annotated corpora are used:

- To train and test NLP technology
- For searching for linguistically relevant patterns

Improving corpus annotation:

- More reliable training and evaluation
- Higher precision and recall in corpus searches

Error Detection: Introduction

Automatic annotation error detection: find inconsistencies in corpus annotation with respect to

- Internal: statistical model based on data within corpus
- External: grammatical model, other external resource

Good overview in Dickinson (2015)

Error Detection: DECCA Project (2003–2008)

Co-PIs: Markus Dickinson and Detmar Meurers

Website: http://decca.osu.edu

Methods for automatic error detection and correction in:

- Part-of-speech tags (Dickinson & Meurers 2003a)
- Treebanks (Dickinson & Meurers 2003b, 2005c)
- Discontinuous treebanks (Dickinson & Meurers 2005b; Dickinson 2005)
- Spoken language corpora (Dickinson & Meurers 2005a)
- Dependencies (Boyd et al. 2008)
- And related issues (Boyd et al. 2007a,b)

Error Detection: DECCA

Detection of Errors and Correction in Corpus Annotation

Variation *n*-gram method: identify repeated material in a corpus that appears with different annotations

Error Detection: DECCA

Detection of Errors and Correction in Corpus Annotation

Variation *n*-gram method: identify repeated material in a corpus that appears with different annotations

Variation can result from:

- genuine ambiguity
- inconsistent annotation

Error Detection: DECCA

Detection of Errors and Correction in Corpus Annotation

Variation *n*-gram method: identify repeated material in a corpus that appears with different annotations

Variation can result from:

- genuine ambiguity
- inconsistent annotation

WSJ POS Annotation

n't/RB	elaborate/VB
n't/RB	elaborate/VB
n't/RB	elaborate/JJ

Error Detection: DECCA

Detection of Errors and Correction in Corpus Annotation

Variation *n*-gram method: identify repeated material in a corpus that appears with different annotations

Variation can result from:

- genuine ambiguity
- inconsistent annotation

WSJ POS Annotation

6:	would	n't/RB	elaborate/VB
2:	did	n't/RB	elaborate/VB
1:	did	n't/RB	elaborate/JJ

Error Detection: DECCA Algorithm

Extract all *n*-grams containing a token that is annotated differently in another occurrence of the *n*-gram in the corpus.

- *variation nucleus*: recurring unit with different annotation
- *variation n-gram*: variation nucleus with identical context

Error Detection: DECCA Algorithm

Extract all *n*-grams containing a token that is annotated differently in another occurrence of the *n*-gram in the corpus.

- *variation nucleus*: recurring unit with different annotation
- *variation n-gram*: variation nucleus with identical context

To be efficient, algorithm calculates variation *n*-grams based on variation $(n-1)$-grams

- Instance of Apriori algorithm (Agrawal & Srikant 1994)

Error Detection: POS Annotation

Dickinson & Meurers (2003a)

WSJ POS Annotation

-LRB- During its centennial year , The Wall Street Journal will report events of the past century **that** *T* stand as milestones of American business history . -RRB-

- 5 times as DT (determiner)
- 5 times as WDT (wh-determiner)

How to determine whether we have ambiguity or an error?

- Context: the more similar the surrounding context, the higher the likelihood of an error

Error Detection: Heuristics

Dickinson & Meurers (2003a)

To improve precision:

- Longer variation *n*-grams are more likely to be errors

that (DT vs. IN vs. RB vs. WDT)
that/??

Error Detection: Heuristics

Dickinson & Meurers (2003a)

To improve precision:

- Longer variation *n*-grams are more likely to be errors

that (DT vs. IN vs. RB vs. WDT)
events of the past century **that/WDT** *T* stand as

Error Detection: Heuristics

Dickinson & Meurers (2003a)

To improve precision:

- Longer variation *n*-grams are more likely to be errors

that (DT vs. IN vs. RB vs. WDT)

events of the past century **that/WDT** *T* stand as

- Distrust the fringe

decided (VBD vs. VBN)

decided/VB? how it will
decided/VB? how it will

Error Detection: Heuristics

Dickinson & Meurers (2003a)

To improve precision:

- Longer variation *n*-grams are more likely to be errors

that (DT vs. IN vs. RB vs. WDT)

events of the past century **that/WDT** *T* stand as

- Distrust the fringe

decided (VBD vs. VBN)

he has **decided/VBN** how it will
he **decided/VBD** how it will

Error Detection: WSJ POS Results

Dickinson & Meurers (2003a)

WSJ corpus:

- 1,289,201 tokens
- 98.2% appear more than once

Sampling 7,141 distinct non-fringe variation *n*-gram types for $3 \leq n \leq 224$:

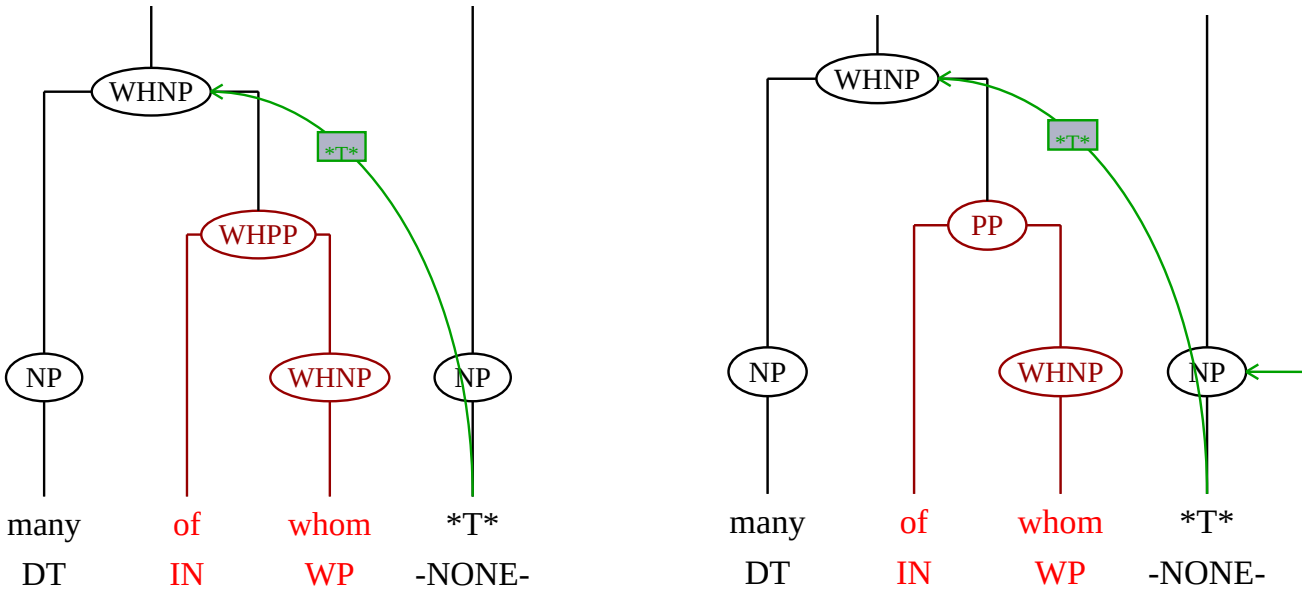
- 92.8% are errors → each at least one correction
- Given 3% estimated POS error rate in the WSJ, the method has a POS error recall of at least 17%

Error Detection: Treebank Annotation

Dickinson & Meurers (2003b)

WSJ Treebank Annotation

many **of whom** *T*

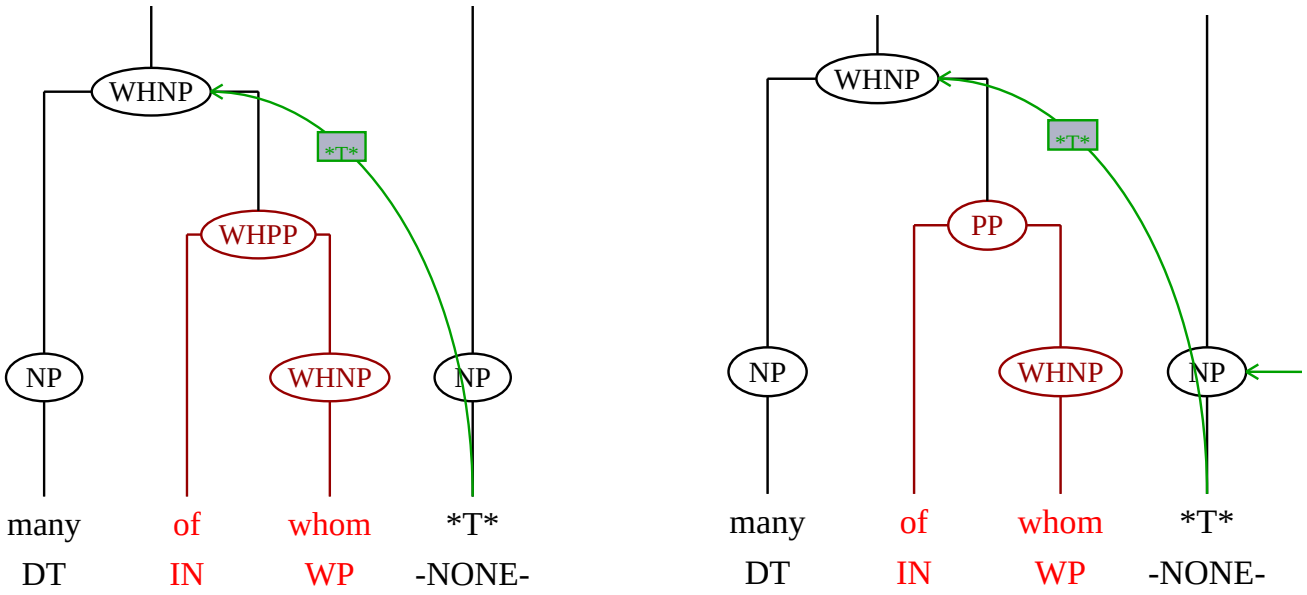


Error Detection: Treebank Annotation

Dickinson & Meurers (2003b)

WSJ Treebank Annotation

many **of whom** *T*



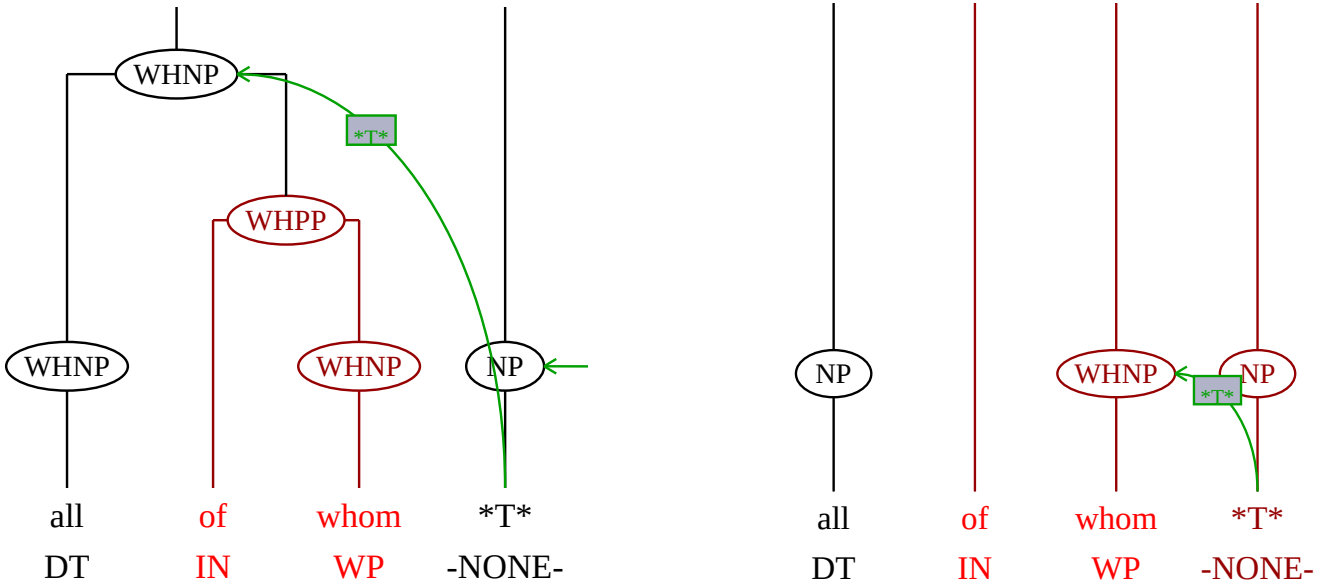
- Different tags: WHPP vs. PP

Error Detection: Treebank Annotation

Dickinson & Meurers (2003b)

WSJ Treebank Annotation

all **of whom** *T*

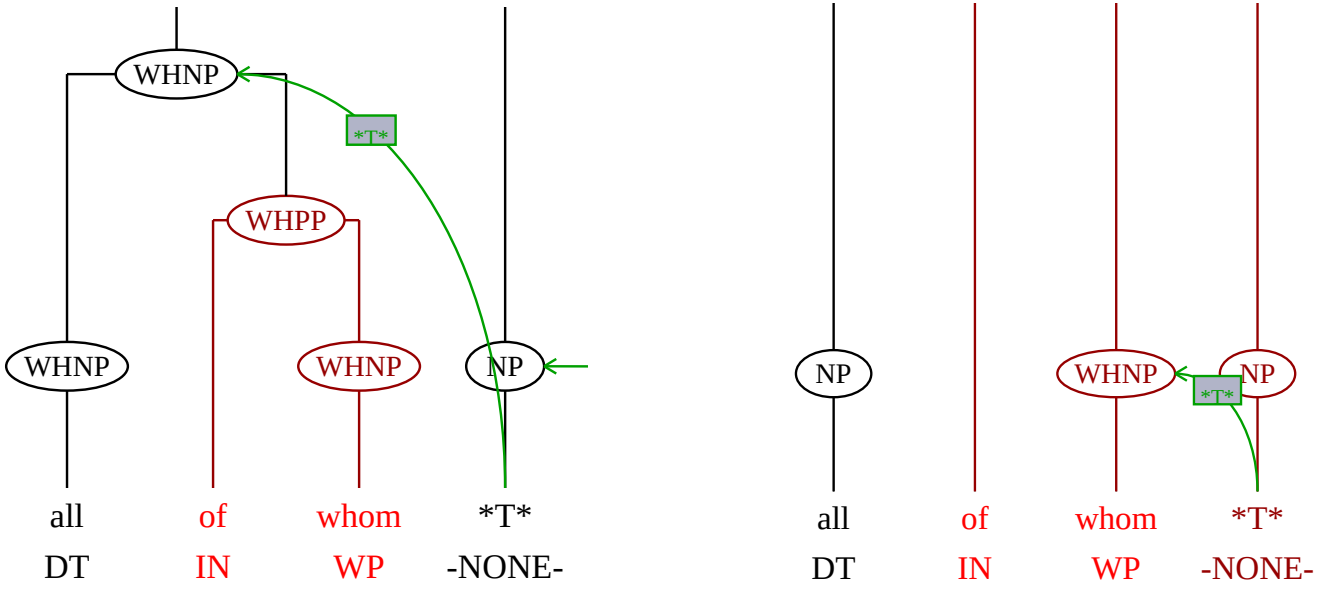


Error Detection: Treebank Annotation

Dickinson & Meurers (2003b)

WSJ Treebank Annotation

all **of whom** *T*



- Annotated vs. not: WHPP vs. NIL

Error Detection: Treebank Output

Sample WSJ 3-gram output:

to say *T*	VP	good 0 * **to say *T*** about the pound 's
to say *T*	NIL	good 0 * **to say *T*** about the pound 's
an example of	NP	as **an example of** the
an example of	NIL	as **an example of** the
year to year	NIL	from **year to year** .
year to year	NP	from **year to year** .

- Algorithm is run separately for each possible constituent length 1..*n*

Error Detection: Spans Beyond Treebanks

DECCA treebank algorithm can be applied to any continuous span annotation.

Examples:

- Named entities (TüBa-D/Z 10.0)
- Error annotation (EFCAMDAT2)

Error Detection: Named Entities in TüBa-D/Z

NIL	" **Roten Rad** "
NX=OTH	" **Roten Rad** "
NX=PER	der Befreiungsfront **Farabundo Marti** (FMLN)
NX=PER	Befreiungsfront **Farabundo Marti** " (FMLN)
NX=PER	der Befreiungsfront **Farabundo Marti** (FMLN)
NX=ORG	Befreiungsfront **Farabundo Marti** " (FMLN)
SIMPX=OTH	" **Indien verstehen** "
SIMPX=ORG	" **Indien verstehen** "
NX=PER	" **Forrest Gump** "
NX=OTH	" **Forrest Gump** "

Error Detection: Error Annotation in EFCAMDAT

EFCAMDAT2 (Geertzen et al. 2013): English L2 learner corpus with 83 million words from 1 million assignments written by 174,000 learners (A1 – C2)

- Partially annotated with feedback provided by language teachers

Text with Feedback

I'm from Brazil, So Paulo {XC: So Paulo, in Brazil}

...

I'm married and my wife is twenty-eighty {SP: eight} .

...

Glad to meet you __ {PU: } !

Error Detection: EFCAMDAT Output

Category	Fringe	Correction	Example
NIL	0	--	years old . I 'm from **Brazil , So Paulo** . There
NIL	0	--	. I 'm from **Brazil , So Paulo** . There are
NIL	0	--	old . I 'm from **Brazil , So Paulo** . There
NIL	0	--	years old . I 'm from **Brazil , So Paulo** ,
NIL	0	--	years old . I 'm from **Brazil , So Paulo** .
NIL	0	--	. I 'm from **Brazil , So Paulo** . There
NIL	0	--	I 'm from **Brazil , So Paulo** . There are
NIL	0	--	from **Brazil , So Paulo** , but I live in
WO	0	Sao Paulo, Brazil	in **Brazil , So Paulo** .
XC	0	So Paulo, in Brazil	years old . I 'm from **Brazil , So Paulo** . There are
WO	0	So Paulo, Brazil	years old . I 'm from **Brazil , So Paulo** ,
WO	0	So Paulo, Brazil	from **Brazil , So Paulo** , but I live in

Another instance:

Category	Correction	Example
D / MW	– / Brazil	Im from Brazil , So Paulo ____

→ DECCA can be used to explore/evaluate crowd-sourced annotations

Error Detection: Increasing Recall

Boyd et al. (2007a)

Two ways to increase recall:

- Redefine *variation nuclei* to extend the set of what counts as recurring data

Variation Nuclei

many **of whom** → many **of {which/whom}**

- Redefine *context* and *heuristics* to obtain more variation *n*-grams

Context

many **of whom** → {some/many/most/all}/DT **of whom**

Error Detection: Discontinuous Spans

Dickinson & Meurers (2005b)

From TIGER Treebank

in diesem Punkt seien ***sich*** *Bonn und London nicht* ***einig***

vs.

in diesem Punkt seien ***sich*** *Bonn und London offensichtlich nicht* ***einig***

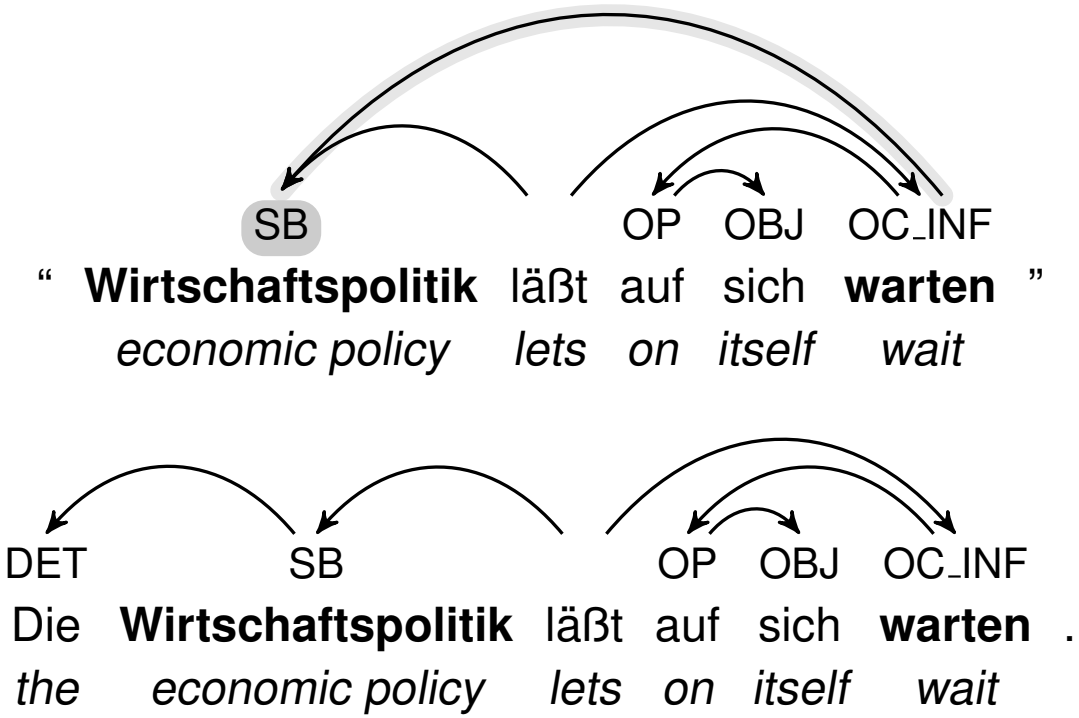
'Bonn and London (clearly) do not agree on this point'

- AP vs. NIL

Error Detection: Dependencies

Boyd et al. (2008)

From TigerDB (Forst et al. 2004):



‘Economic policy is a long time coming.’

- SB vs. NIL

Error Detection: Summary

Automatic error detection:

- Leads to improved corpus quality for NLP / search
- Provides feedback to corpus developers for annotation scheme design and documentation

DECCA variation *n*-gram approach:

- Finds errors in in token, span, discontinuous span, and dependency annotation
- Does not depend on language, corpus, or tagset

Website: http://decca.osu.edu

Download DECCA code:

http://github.com/adrianeboyd/decca

Conclusion

Corpus annotation with

- explicit links to the original data
- attention to consistency through error detection, etc.
  - which informs annotation guidelines

has a wider range of

- potential users
  - non-specialists (e.g., language teachers)
- potential uses
  - gold standard in evaluations
  - high-quality, customizable training data

References

Agrawal, R. & R. Srikant (1994). Fast Algorithms for Mining Association Rules in Large Databases. In J. B. Bocca, M. Jarke & C. Zaniolo (eds.), VLDB 1994. Morgan Kaufmann, pp. 487–499.

Beinborn, L., T. Zesch & I. Gurevych (2016). Predicting the Spelling Difficulty of Words for Language Learners. In Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications. San Diego, CA.

Beißwenger, M., S. Bartsch, S. Evert & K.-M. Würzner (2016). EmpiriST 2015: A Shared Task on the Automatic Linguistic Annotation of Computer-Mediated Communication and Web Corpora. In Proceedings of the 10th Web as Corpus Workshop. Association for Computational Linguistics, pp. 44–56. http://www.aclweb.org/anthology/W16-2606.

Boyd, A., M. Dickinson & D. Meurers (2007a). Increasing the Recall of Corpus Annotation Error Detection. In Proceedings of the Sixth Workshop on Treebanks and Linguistic Theories (TLT-07). Bergen, Norway. http://purl.org/dm/papers/boyd-et-al-07b.html.

Boyd, A., M. Dickinson & D. Meurers (2007b). On Representing Dependency Relations – Insights from Converting the German TiGerDB. In Proceedings of the Sixth Workshop on Treebanks and Linguistic Theories (TLT-07). Bergen, Norway. http://purl.org/dm/papers/boyd-et-al-07b.html.

Boyd, A., M. Dickinson & D. Meurers (2008). On Detecting Errors in Dependency Treebanks. Research on Language and Computation 6(2), 113–137. http://purl.org/dm/papers/boyd-et-al-08.html.

Boyd, A., J. Hana et al. (2014). The MERLIN corpus: Learner language and the CEFR. In Proceedings of LREC 2014. Reykjavik, Iceland: European Language Resources Association (ELRA).

de Kok, D. (2014). TüBa-D/W: a large dependency treebank for German. In Proceedings of the Thirteenth International Workshop on Treebanks and Linguistic Theories (TLT13). Tübingen, Germany.

Dickinson, M. (2005). Error detection and correction in annotated corpora. Ph.D. thesis, The Ohio State University. http://www.ohiolink.edu/etd/view.cgi?osu1123788552.

Dickinson, M. (2015). Detection of Annotation Errors in Corpora. Language and Linguistics Compass 9(3), 119–138.

Dickinson, M. & W. D. Meurers (2003a). Detecting Errors in Part-of-Speech Annotation. In Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics (EACL-03). Budapest, Hungary, pp. 107–114. http://aclweb.org/anthology/E03-1068.

Dickinson, M. & W. D. Meurers (2003b). Detecting Inconsistencies in Treebanks. In Proceedings of the Second Workshop on Treebanks and Linguistic Theories (TLT-03). Växjö, Sweden, pp. 45–56. http://purl.org/dm/papers/dickinson-meurers-tlt03.html.

Dickinson, M. & W. D. Meurers (2005a). Detecting Annotation Errors in Spoken Language Corpora. In The Special Session on treebanks for spoken language and discourse at NODALIDA-05. Joensuu, Finland. http://purl.org/~dm/papers/dickinson-meurers-nodalida05.html.

Dickinson, M. & W. D. Meurers (2005b). Detecting Errors in Discontinuous Structural Annotation. In Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL-05). pp. 322–329. http://aclweb.org/anthology/P05-1040.

Dickinson, M. & W. D. Meurers (2005c). Prune Diseased Branches to Get Healthy Trees! How to Find Erroneous Local Trees in a Treebank and Why It Matters. In Proceedings of the Fourth Workshop on Treebanks and Linguistic Theories (TLT-05). Barcelona, Spain. http://purl.org/dm/papers/dickinson-meurers-tlt05.html.

Dridan, R. & S. Oepen (2012). Tokenization: Returning to a Long Solved Problem – A Survey, Contrastive Experiment, Recommendations, and Toolkit. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). Jeju Island, Korea: Association for Computational Linguistics, pp. 378–382. http://www.aclweb.org/anthology/P12-2074.

Eckart de Castilho, R. (2016). Automatic Analysis of Flaws in Pre-Trained NLP Models. In Proceedings of the Third International Workshop on Worldwide Language Service Infrastructure and Second Workshop on Open Infrastructures and Analysis Frameworks for Human Language Technologies (WLSI3nOIAF2) at COLING 2016. pp. 19–27.

Forst, M., N. Bertomeu, B. Crysmann, F. Fouvry, S. Hansen-Schirra & V. Kordoni (2004). Towards a Dependency-Based Gold Standard for German Parsers. The TIGER Dependency Bank. In S. Hansen-Schirra, S. Oepen & H. Uszkoreit (eds.), 5th International Workshop on Linguistically Interpreted Corpora (LINC-04) at COLING. Geneva, Switzerland: COLING, pp. 31–38. http://aclweb.org/anthology/W04-1905.

Geertzen, J., T. Alexopoulou & A. Korhonen (2013). Automatic linguistic annotation of large scale L2 databases: The EF-Cambridge Open Language Database (EFCAMDAT). In Proceedings of the 31st Second Language Research Forum (SLRF). Cascadilla Press. http://purl.org/icall/efcamdat.

Hana, J., A. Rosen, S. Škodová & B. Štindlová (2010). Error-Tagged Learner Corpus of Czech. In Proceedings of the Fourth Linguistic Annotation Workshop. Uppsala, Sweden: Association for Computational Linguistics.

He, Y. & M. Kayaalp (2006). A Comparison of 13 Tokenizers on MEDLINE. Tech. Rep. LHNCBC-TR-2006-003, Lister Hill National Center for Biomedical Communications.

Jurish, B. & K.-M. Würzner (2013). Word and sentence tokenization with Hidden Markov Models. JLCL. Journal for Language Technology and Computational Linguistics 28(2), 61–83.

Reznicek, M., A. Lüdeling, C. Krummes & F. Schwantuschke (2012). Das Falko-Handbuch. Korpusaufbau und Annotationen Version 2.0. http://purl.org/net/Falko-v2.pdf.

Wisniewski, K., K. Schöne, L. Nicolas, C. Vettori, A. Boyd, D. Meurers, A. Abel & J. Hana (2013). MERLIN: An online trilingual learner corpus empirically grounding the European Reference Levels in authentic learner data. In ICT for Language Learning. Florence, Italy.