Multilingual Coreference Resolution with BART

Yannick Versley
SFB 833 Univ. Tübingen

26. November 2009

Eberhard Karls Universität Tübingen

Coreference

John Simon, Chief Financial Officer of Prime Corp since 1986 saw **his** pay jump 20%, to $1.3 million, as **the 37-year-old** also became **the financial service company's president**.

- Multiple descriptions refer to the same (discourse/ real-world) entity
  - John Simon
  - he
  - the 37-year-old

BART: Beautiful Anaphora Resolution Toolkit

- Johns Hopkins Summer Workshop: **Project ELERFED**
  "Encyclopedic and Lexical Knowledge For Entity Disambiguation"
- Based on a system by Simone Ponzetto
  (Ponzetto and Strube, 2006)
- Developed a version for Italian
  (in the **LiveMemories** project)
  Massimo Poesio, Kepa Rodriguez,
  Olga Uryupina, Yannick Versley
- Adaptations for German
  Samuel Broscheid, Simone Ponzetto

BART: Beautiful Anaphora Resolution Toolkit

Why modular coreference resolution?

- **Decoupling** of work on different areas
  at the same time: combined system with sum of improvements
- Lower the threshold for **realistic investigation** of coreference
- Shorten the distance between research and **application**

A framework approach to NLP

Experimental paradigm:

- Annotation
- Feature extraction and learning
- Evaluation
- Error analysis (+correction +start over)

Recurring problems $\Rightarrow$ don't always re-invent the wheel

BART Components

- **Preprocessing**
  Aggregating information in MMAX2 annotation levels: named entities, syntactic analyses, lemmas, etc.
- **Extraction of candidate NPs**
  Creating markables from chunks and named entities.
- **Extracting features of NPs/candidate pairs**
  Linguistic category, semantic class …
- **Resolution model**
  Encoding of coreference partition as classifier decisions and features.

BART Components

- **Preprocessing**
  Aggregating information in MMAX2 annotation levels:
  named entities, syntactic an
- **Extraction of candidate NPs**
  Creating markables from chunks and named entities.
- **Extracting features of NPs/candidate pairs**
  Linguistic category, semantic class …
- **Resolution model**
  Encoding of coreference partition as classifier decisions and features.

Modular Pipeline Architecture
Standoff Annotation

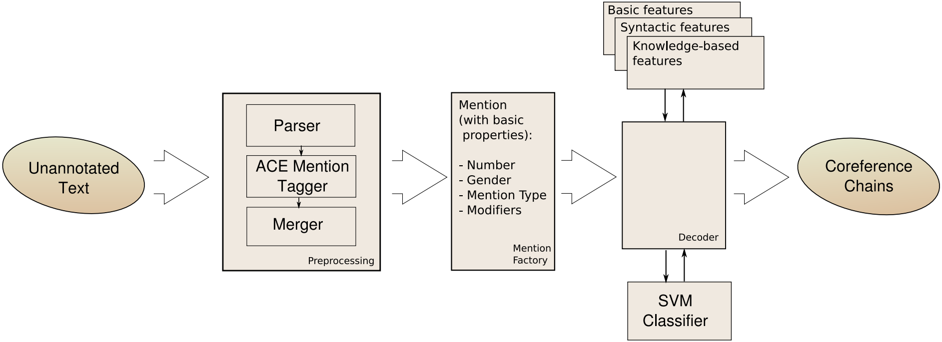
BART Components

- **Preprocessing**
  Aggregating information in MMAX2 annotation levels:
  named entities, syntactic an
- **Extraction of candidate NPs**
  Creating markables from chunks and named entities.
- **Extracting features of NPs/candidate pairs**
  Linguistic category, semantic class …
- **Resolution model**
  Encoding of coreference partition as classifier decisions and features.

Modular Pipeline Architecture
Standoff Annotation

Features and Learner
described in XML file

Data flow


Unannotated Text
Parser
ACE Mention Tagger
Merger
Preprocessing
Mention (with basic properties):
- Number
- Gender
- Mention Type
- Modifiers
Mention Factory
Basic features
Syntactic features
Knowledge-based features
Decoder
SVM Classifier
Coreference Chains

Machine Learning

Interfaces to:

- **WEKA** Toolkit (Witten and Frank, 2005)
  - C4.5 (J48), RIPPER (JRip), all others

Machine Learning

Interfaces to:

- **WEKA** Toolkit (Witten and Frank, 2005)
  - C4.5 (J48), RIPPER (JRip), all others
- **SVMlight-TK** (Moschitti, 2006)
  - SVMs, different kernels (linear, polynomial, ...)
  - Tree-Kernel
  - Build your own kernel

Machine Learning

Interfaces to:

- **WEKA** Toolkit (Witten and Frank, 2005)
  - C4.5 (J48), RIPPER (JRip), all others
- **SVMlight-TK** (Moschitti, 2006)
  - SVMs, different kernels (linear, polynomial, ...)
  - Tree-Kernel
  - Build your own kernel
- **MaxEnt**
  - Feature combination
  - Ranking: direct choice of the "best" candidate

Machine Learning: Basic Setup

- **Create data**
  `Instance`, `InstanceWriter`
- Learn a classifier
- **Classification**
  `OfflineClassifier`
- Common **Description of Features**
  `FeatureType`, `FeatureDescription`,
  `FeatureExtractor`

Overview

Today:

- Kernel-based expletive detection
- Ranking classifiers with tuning
- Multilingual coreference resolution using the LanguagePlugin mechanism

Kernel Methods

Many coreference systems use impoverished information

- using only shallow information limits the possible accuracy
- need to make use of structural and lexical/ontological information

Kernel methods allow easier modeling of this information than rule-based feature extraction

Kernel Engineering

- The Kernel trick:
  Instead of directly using vectors $\in R^n$,
  use a higher-dimensional (Hilbert) space
  by replacing the dot product $\langle \cdot, \cdot \rangle$ with a kernel $\kappa(\cdot, \cdot)$
- Convolution kernels allow to use structured data
  - sequence kernel
  - (partial) tree kernel
- Operation on kernels allow us to combine different views on instances:
  - multiplication
  - addition
  - function composition

Binding Classification

Given two mentions in one sentence, determine whether they might corefer

- c-command and (non-)reflexivity:
  Peter$_i$ likes him$_j$/himself$_i$
- lexical determination of control:
  Peter$_i$ asked John$_j$ to shave himself$_j$
  Peter$_i$ threatened John$_j$ to shave himself$_i$
- Only preferences in subordinate clauses:
  Peter$_i$ said that he$_i$ likes ice cream

Binding Classification

Given two mentions in one sentence,
determine whether they might corefer

- c-command and (non-)reflexivity:
  Peter$_i$ *likes him*$_j$*/himself*$_i$
- lexical determination of control:
  Peter$_i$ *asked John*$_j$ *to shave himself*$_j$
  Peter$_i$ *threatened John*$_j$ *to shave himself*$_i$
- Only preferences in subordinate clauses:
  Peter$_i$ *said that he*$_i$ *likes ice cream*

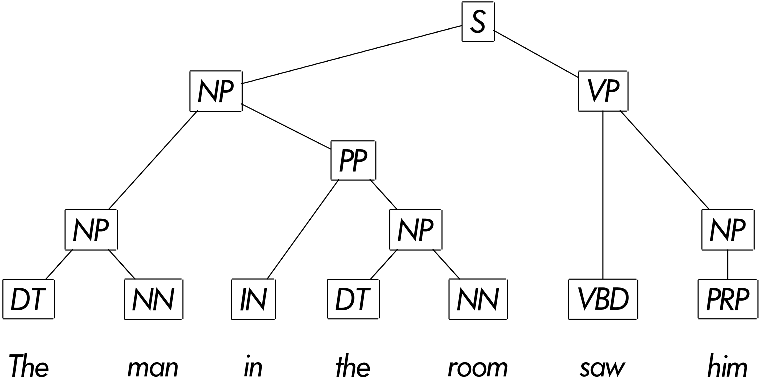
Binding Classification

Given two mentions in one sentence,
determine whether they might corefer

- c-command and (non-)reflexivity:
  Peter$_i$ likes him$_j$/himself$_i$
- lexical determination of control:
  Peter$_i$ asked John$_j$ to shave himself$_j$
  Peter$_i$ threatened John$_j$ to shave himself$_i$
- Only preferences in subordinate clauses:
  Peter$_i$ said that he$_i$ likes ice cream

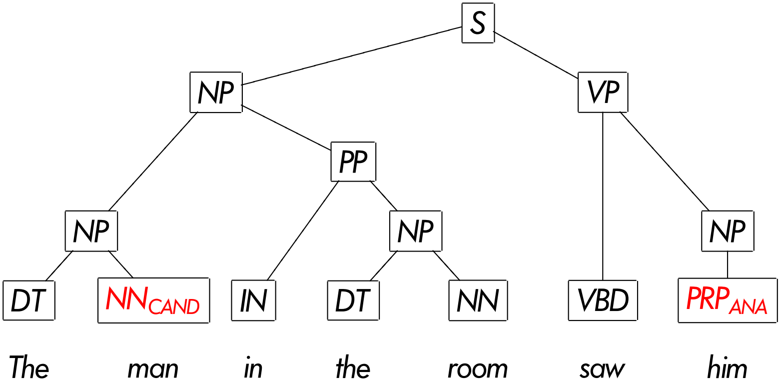
Binding Classification: Example

Q: "The man in the room" == "him"?



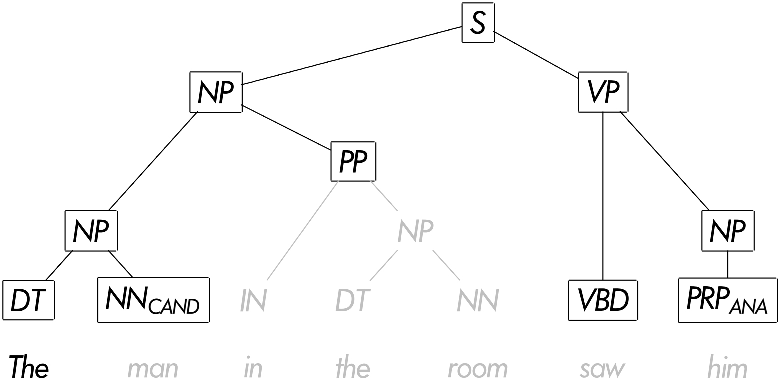
Binding Classification: Example

Mark antecedent and candidate



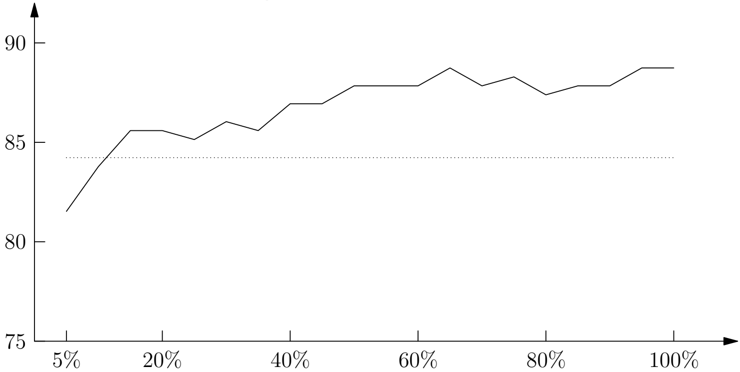
Binding Classification: Example

Prune expansions not on path



Binding Classifier: Results on ACE-2

Accuracy on same-sentence pronoun antecedent candidates:



⇒ Binding Classifier helps, but needs enough data.

Expletive Detection

Filter out occurrences of "*it*"
that do not co-refer with a previous NP

- Extrapositon
  It has been confirmed that Peter drinks beer.
- Cleft
  It is Peter who ate the ice cream.
- Weather verbs
  It was snowing.
- Idiomatic
  It is your turn.

Expletive Detection (2)

Expletive Detection with memory-based learning (Boyd et al., 2005)

- hand-crafted surface patterns:
  - extrapositional it (9 patterns)
    it VERB ADJ that, it VERB to, ...
  - cleft it (2 patterns)
    it be who/which/that, it who/which/that
  - list of weather/condition/time/place words
  - idiom patterns (10 patterns)
- shallow features
  - preceding/following full verb
  - following adjective
  - surrounding POS tags
  - previous word is preposition

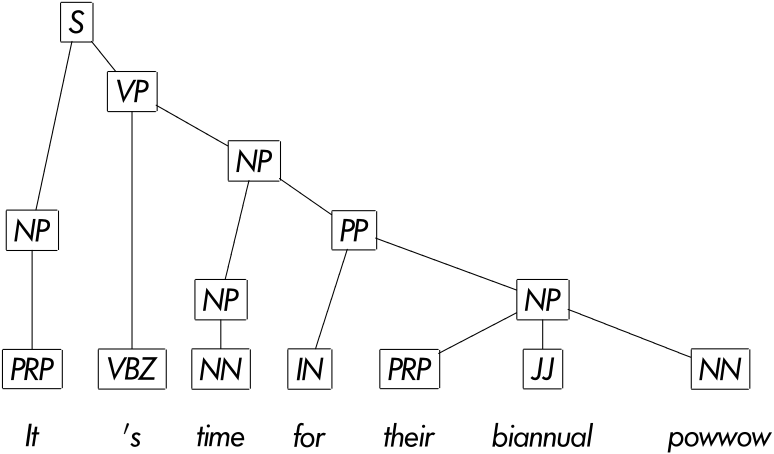
Kernel-based Expletive Detection

Capture necessary information contained in the parse tree:

- predicate (verb, noun/adj in predicative position)
- subclauses (SBAR, TO)
- rough syntactic structure
- flatten VPs and use partial tree kernel
- use both unmodified and pruned tree

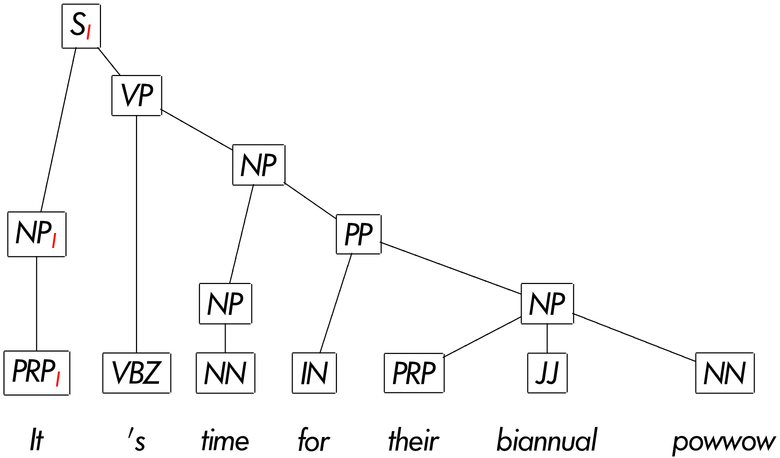
Kernel-based Expletive Detection (2)

It's time for their biannual powwow



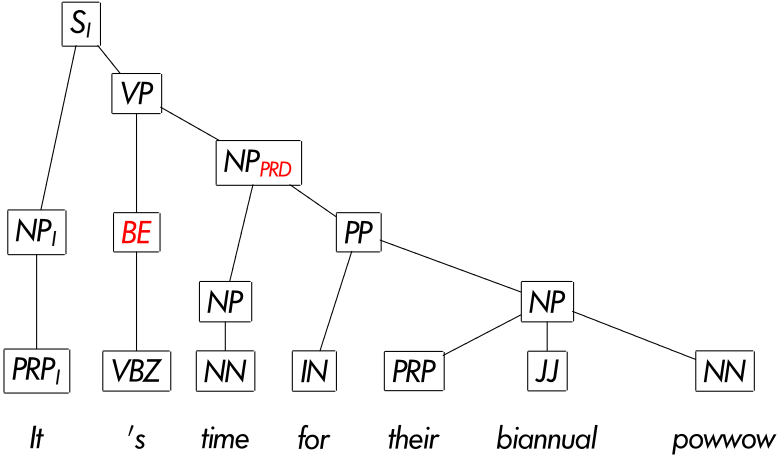
Kernel-based Expletive Detection (2)

Mark nodes on the path to the pronoun



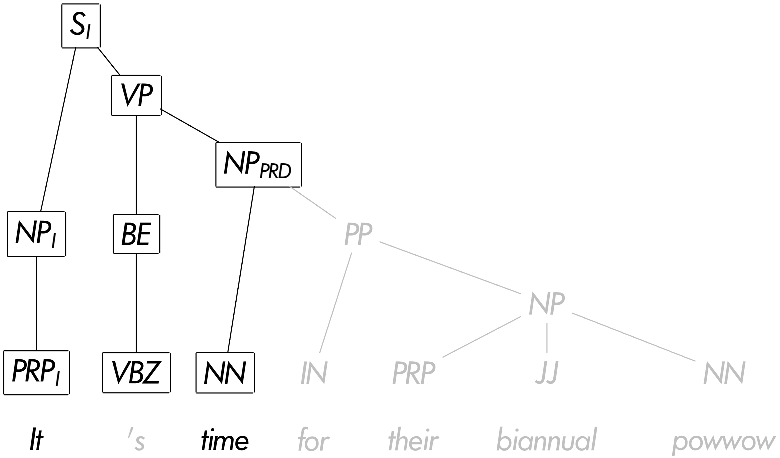
Kernel-based Expletive Detection (2)

Mark copula predicate



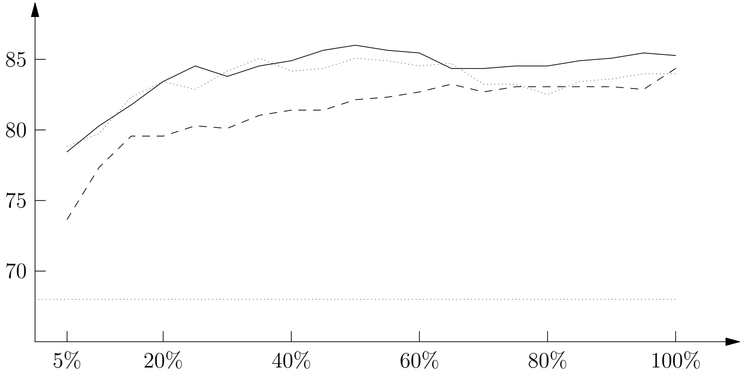
Kernel-based Expletive Detection (2)

Prune non-informative nodes



Expletive Detection Results

Accuracy on *it* in BBN Pronoun Corpus:



⇒ Similar performance to pattern-based approach.

Expletive Detection Results

Is it actually useful?

- On ACE: not useful,
  because ACE mention tagger already does this.
- On MUC: small improvement (not all that many *it* pronouns)

Soon et al

- first ML-based system with good results on MUC6
- described well in (Soon et al., 2001)
- baseline for Ponzetto and Strube (2006) and others

Soon et al: Learning

he she Clinton the men he

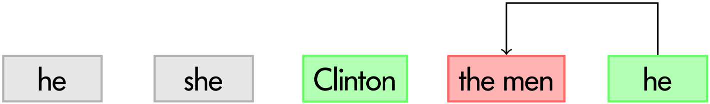
dist	*number*	*gender*	*coref*

Soon et al: Learning

he	she	Clinton	the men	he

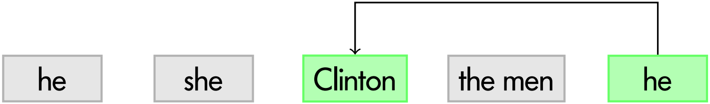
dist	*number*	*gender*	*coref*

Soon et al: Learning



dist	*number*	*gender*	*coref*
1	-	?	-

Soon et al: Learning



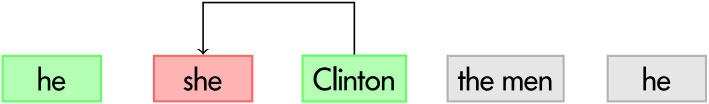
dist	*number*	*gender*	*coref*
1	-	?	-
1	+	?	+

Soon et al: Learning

he	she	Clinton	the men	he

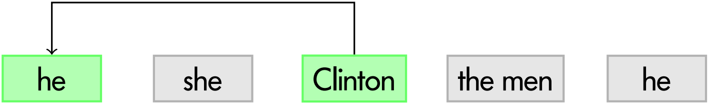
dist	*number*	*gender*	*coref*
1	-	?	-
1	+	?	+

Soon et al: Learning



dist	*number*	*gender*	*coref*
1	-	?	-
1	+	?	+
2	+	?	-

Soon et al: Learning



dist	*number*	*gender*	*coref*
1	-	?	-
1	+	?	+
2	+	?	-
3	+	?	+

Soon et al: Learning

he she Clinton the men he

dist	*number*	*gender*	*coref*
1	-	?	-
1	+	?	+
2	+	?	-
3	+	?	+

Soon et al: Decision Function

Look for closest antecedent with

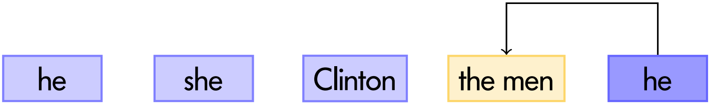
- String matching (pronouns as well as non-pronouns)
- Alias
- Apposition
- For pronouns: compatible antecedents which are
  - also pronouns or
  - in the same sentence

he	she	Clinton	the men	he

Soon et al: Decision Function

Look for closest antecedent with

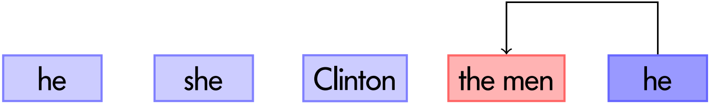
- String matching (pronouns as well as non-pronouns)
- Alias
- Apposition
- For pronouns: compatible antecedents which are
    - also pronouns or
    - in the same sentence



Soon et al: Decision Function

Look for closest antecedent with

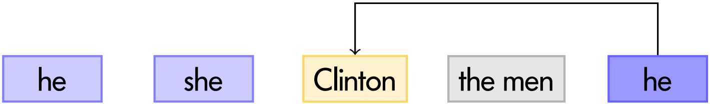
- String matching (pronouns as well as non-pronouns)
- Alias
- Apposition
- For pronouns: compatible antecedents which are
  - also pronouns or
  - in the same sentence



Soon et al: Decision Function

Look for closest antecedent with

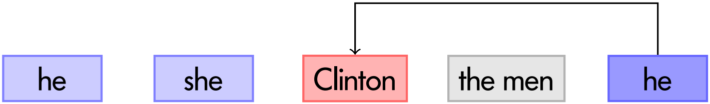
- String matching (pronouns as well as non-pronouns)
- Alias
- Apposition
- For pronouns: compatible antecedents which are
  - also pronouns or
  - in the same sentence



Soon et al: Decision Function

Look for closest antecedent with

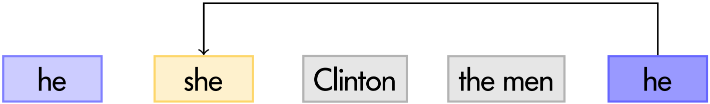
- String matching (pronouns as well as non-pronouns)
- Alias
- Apposition
- For pronouns: compatible antecedents which are
  - also pronouns or
  - in the same sentence



Soon et al: Decision Function

Look for closest antecedent with

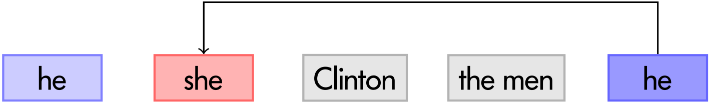
- String matching (pronouns as well as non-pronouns)
- Alias
- Apposition
- For pronouns: compatible antecedents which are
  - also pronouns or
  - in the same sentence



Soon et al: Decision Function

Look for closest antecedent with

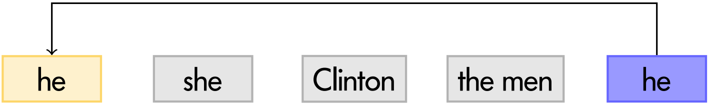
- String matching (pronouns as well as non-pronouns)
- Alias
- Apposition
- For pronouns: compatible antecedents which are
  - also pronouns or
  - in the same sentence



Soon et al: Decision Function

Look for closest antecedent with

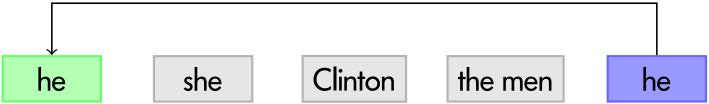
- String matching (pronouns as well as non-pronouns)
- Alias
- Apposition
- For pronouns: compatible antecedents which are
  - also pronouns or
  - in the same sentence



Soon et al: Decision Function

Look for closest antecedent with

- String matching (pronouns as well as non-pronouns)
- Alias
- Apposition
- For pronouns: compatible antecedents which are
  - also pronouns or
  - in the same sentence



The Soon Paradox

- Add an informative feature,
  the performance goes down

The Soon Paradox

- Add an informative feature,
  the performance goes down
- This is frustrating

The Soon Paradox

- Add an informative feature, the performance goes down
- This is frustrating
- (Possibly) due to a variety of reasons:
  - Soon et al. do good sample selection
  - The MUC measure encourages over-merging
  - Precision/Recall balance is delicate

Can we do better?

- Ng and Cardie (2002): use confidence measure of classifier to choose among multiple positively classified items
- Yang et al. (2003, 2005): use tournament ranking to choose among compatible antecedents (pronouns) or positively classified items (nominals, Yang03) or use sample selection to get “both-lose” examples (Yang05)

$\Rightarrow$ There’s still sample selection involved!

Can we do better? (2)

What we actually want

- something simple (no ILP, reasonably non-fancy)
- allows to stick in nice features
- not care about sample selection or P/R balance

Can we do better? (3)

- MaxEnt ranking resolver
- One ranker for each major category (pronouns, 1st/2nd person, nominals, names)
- Automatically adjust P/R balance to optimize the actual evaluation metric on cross-validation (don't look at test set)

Ranking Resolution

Maximum Entropy ranking:

- Rosenfeld (1996): Cache-based language model
- Johnson et al. (1999): Parse selection for LFG

Ranking Resolution

Maximum Entropy ranking:

- Rosenfeld (1996): Cache-based language model
- Johnson et al. (1999): Parse selection for LFG

Ranking resolution:

- Morton (2000): use most-positive decision to resolve a pronoun
- Ng and Cardie (2002): use classifier confidence
- Luo et al. (2004): use classifier output plus *start penalty*

Ranking Resolution

Maximum Entropy ranking:

- Rosenfeld (1996): Cache-based language model
- Johnson et al. (1999): Parse selection for LFG

Ranking resolution:

- Morton (2000): use most-positive decision to resolve a pronoun
- Ng and Cardie (2002): use classifier confidence
- Luo et al. (2004): use classifier output plus *start penalty*

MaxEnt ranking resolution:

- Works well for definite NP anaphora (Versley, 2006)
- Works well for pronouns (Denis and Baldridge, 2007)

How it works

Score for one possible antecedent y with features $f(y)$:

$$\mu(y) := e^{\langle w, f(y) \rangle}$$

How it works

Score for one possible antecedent y with features $f(y)$:

$$\mu(y) := e^{\langle w, f(y)\rangle}$$

This defines a probability distribution

$$\hat{P}(y) := \frac{\mu(y)}{\sum_{y' \in Y} \mu(y')}$$

often written as

$$\hat{P}(y) = \frac{1}{Z} \exp(\langle w, f(y)\rangle)$$

How it works (2)

(Log-)Likelihood of the right decision according to $\hat{P}$:

$$
\begin{aligned}
\mathrm{LL}(\theta|w) &= \log \prod_{x,y\in\theta} \hat{P}(y|x) \\
&= \sum_{x,y\in\theta} \langle w, f(y)\rangle - \log \sum_{y'} e^{\langle w,f(y')\rangle}
\end{aligned}
$$

How it works (2)

(Log-)Likelihood of the right decision according to $\hat{P}$:

$$
\begin{aligned}
\mathrm{LL}(\theta|w) &= \log \prod_{x,y\in\theta} \hat{P}(y|x) \\
&= \sum_{x,y\in\theta} \langle w, f(y)\rangle - \log \sum_{y'} e^{\langle w, f(y')\rangle}
\end{aligned}
$$

If we always have one single good y, this function is convex
$\Rightarrow$ optimization is easy but (for large datasets) time-consuming

How it works (2)

(Log-)Likelihood of the right decision according to $\hat{P}$:

$$\begin{aligned}\mathrm{LL}(\theta|w) &= \log \prod_{x,y\in\theta} \hat{P}(y|x) \\ &= \sum_{x,y\in\theta} \langle w, f(y)\rangle - \log \sum_{y'} e^{\langle w,f(y')\rangle}\end{aligned}$$

If we always have one single good y, this function is convex
$\Rightarrow$ optimization is easy but (for large datasets) time-consuming

Multiple good y: may be non-convex, but mostly harmless

Small things

- feature combination: allows to (partially) account for feature inter-dependencies
- add one more candidate for *"this is discourse-new, don't resolve"*
- this allows to integrate discourse-new detection

Small things

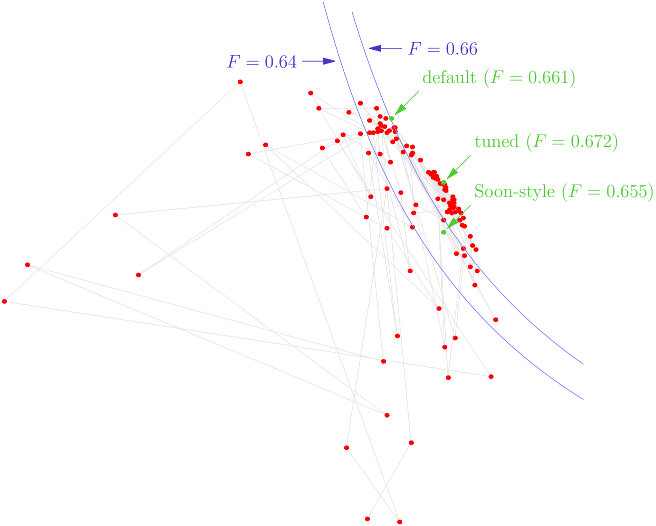
- feature combination: allows to (partially) account for feature inter-dependencies
- add one more candidate for *"this is discourse-new, don't resolve"*
- this allows to integrate discourse-new detection
- we can adjust precision/recall balance by multiplying the no-antecedent μ by a factor
  four resolvers, three distance thresholds $\Rightarrow$ 7 magic numbers

Tuning Thresholds

- First Optimizer (MaxEnt / L-BFGS):
  In: Feature weights **Out:** Likelihood
- Second Optimizer:
  In: Tuning values **Out:** MUC-Score

F-Score isn't as well-behaved as log-likelihood
$\Rightarrow$ use fewer parameters, different optimization method

Tuning Thresholds (2)



Results on system mentions

"True" mentions	All		Pronouns		Nominals		Names	
	MUC-F	Link-F	Prec	Recl	Prec	Recl	Prec	Recl
Soon/MaxEnt	**63.8**	65.1	70.2	72.8	33.7	36.6	76.0	78.1
+SemClass+LogDist	63.7	66.7	**70.7**	71.2	**46.0**	30.4	77.2	78.4
candrank	63.3	**67.8**	67.5	**76.1**	44.8	30.0	**79.7**	**80.8**
mixrank	63.1	64.1	65.5	73.8	32.6	**37.0**	75.5	78.6
purerank (untuned)	61.4	67.5	**77.9**	68.9	**48.3**	11.3	**86.1**	76.2
purerank (on sys m.)	**63.0**	**68.1**	77.3	71.2	42.0	25.5	80.5	78.1
purerank (on gold m.)	59.3	51.7	74.8	**74.0**	15.2	**52.0**	43.9	**84.2**

Results on “true” mentions

System mentions	All		Pronouns		Nominals		Names	
	MUC-F	Link-F	Prec	Recl	Prec	Recl	Prec	Recl
Soon/MaxEnt	**71.7**	75.3	**77.4**	78.2	59.0	**42.9**	92.1	83.3
+SemClass+LogDist	68.8	74.8	76.5	74.9	**75.2**	33.2	**96.6**	82.8
candrank	70.1	**76.0**	76.6	**78.7**	72.9	41.3	94.9	82.6
mixrank	70.9	73.8	72.7	76.8	58.4	41.6	91.9	**83.5**
purerank (untuned)	70.3	**77.2**	77.8	72.9	**71.8**	45.7	**95.2**	**88.9**
purerank (on sys m.)	71.3	76.2	**81.9**	73.6	56.4	53.7	92.7	84.0
purerank (on gold m.)	**75.4**	70.2	78.3	**77.0**	40.1	**62.9**	67.1	87.9

What's not to like?

- Efficiency
  - takes several days
  - …and that's without any slow stuff (SVM, expensive features)
  - maybe: use dev dataset
  - maybe: try out different optimizers
    (Ant Colonies, Differential Evolution)
- Evaluation Metric (we have a CEAF scorer)
- add more features

Multilingual coreference resolution

- Most research is done only for English
  - Everyone claims their system is language-independent
  - Yet only a small fraction looks at multiple languages and/or multiple language pairs (for MT)

Multilingual coreference resolution

- Most research is done only for English
  - Everyone claims their system is language-independent
  - Yet only a small fraction looks at multiple languages and/or multiple language pairs (for MT)
- If we take the "Linguistics" bit in CL seriously,
  - we should work not only on English, not only on WSJ/PTB/OntoNotes
  - we should also get past extremely shallow approaches
  - while still inviting in the community at large

CL Typology

- English:
  fixed word order, morphologically poor
  no syntactic gender
- German:
  flexible word order, morphologically rich(er)
  syntactic gender
- Italian:
  mostly-fixed word order, morphologically rich(er)
  syntactic gender, clitics, subject zero pronouns

Resource Situation (1)

English:

- large treebank (PTB), good parsing
- lemmatization (Minnen et al., 2001)
- WordNet, ACE/BBN Entity type, PropBank/FrameNet, you name it...

Resource Situation (2)

German:

- large treebanks (TIGER, TüBa-D/Z)
- decent parsing (but: morphology/GFs mostly uncared for)
- lemma/morph no real standard, use SMOR (Schmid et al., 2004)
- GermaNet, Salsa

Resource Situation (3)

Italian:

- small treebanks (TUT: 2k, ISST: ≈4k), parsing problematic
- TextPro (Pianta and Zanoli, 2007),
  Morph-It (Zanchetta and Baroni, 2005)
- MultiWordNet, iCab (ACE-style NER+coref corpus)

Resource Situation: Summary

- different **resource situation**:
  parsing (always) better than chunking for English,
  (probably) same for German
  chunking (usually) better than parsing for Italian
- a **common denominator**
  parsing or chunking
  morphology
  a wordnet

the LanguagePlugin idea (1)

Have a single version of BART that

- can be used for competitive experiments in multiple languages
- still allows language-specific features
- but with a language-independent baseline

the LanguagePlugin idea (2)

Changes:

- data conversion to MMAX2 format
- MMAX2 (on-disk) to Mention objects

Common denominator:

- morphology: number/gender, person
- head lemma
- semantic class
- mention type (definite/indefinite, pro/nom/nam)

First results: Italian

Evalita 2009 "Entity Detection and Recognition"
(Bernaola Biggio et al., 2009)

- ACE-style coreference on iCab corpus (only PER/ORG/GPE/LOC)
- used SVM-based mention tagger (Silvana Bernaola)
  - identifies minimal spans (1st/2nd level)
  - uses a variety of features, including MultiWordNet-based
- improved Alias feature (Olga Uryupina)

MUC: R=0.458 P=0.723 F=0.561

First results: German

Experiments on TüBa-D/Z (Broscheid, 2009)

- MMAX2 conversion of TüBa-D/Z based on mention extraction from (Versley, 2006)
  - uses TüBa-D/Z trees
  - automatic assignment of semantic classes
- baseline features based on Klenner and Ailloud (2008)
- binding restrictions
- GermaNet similarity

First results: German

Experiments on TüBa-D/Z (Broscheid, 2009)

- MMAX2 conversion of TüBa-D/Z based on mention extraction from (Versley, 2006)
  - uses TüBa-D/Z trees
  - automatic assignment of semantic classes
- baseline features based on Klenner and Ailloud (2008)
- binding restrictions
- GermaNet similarity

Results (MUC, gold mentions):

	R	P	F1
J48 baseline	0.609	0.707	0.654
MaxEnt split	0.756	0.808	0.781
+all features	0.784	0.822	0.802

Summary / Future work

BART

- Coreference resolution for multiple languages
- State-of-the-art ML: Kernels, MaxEnt ranking
- Preprocessing still hairy&difficult at times

Look out for:

- **SemEval 2010** task on Coreference Resolution
  Spanish, Catalan, English, Italian, Dutch, German

Thanks for listening!!!

ENDE

Bernaola Biggio, S. M., Giuliano, C., Poesio, M., Versley, Y., Uryupina, O., and Zanoli, R. (2009). Local entity detection and recognition task. In *Proceedings of Evalita-2009*.

Boyd, A., Gegg-Harrison, W., and Byron, D. (2005). Identifying non-referential it: a machine learning approach incorporating linguistically motivated features. In *Proceedings of the ACL Workshop on Feature Engineering for Machine Learning in Natural Language Processing*.

Broscheid, S. (2009). Adapting the anaphor resolution toolkit bart to german. B.a. thesis, University of Heidelberg.

Denis, P. and Baldridge, J. (2007). A ranking approach to pronoun resolution. In *Proc. IJCAI 2007*.

Johnson, M., Geman, S., Canon, S., Chi, Z., and Riezler, S. (1999). Estimators for stochastic "unification-based" grammars. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL-1999)*.

Klenner, M. and Ailloud, E. (2008). Enhancing coreference clustering. In *Second Bergen Workshop on Anaphora Resolution (WAR II)*.

Luo, X., Ittycheriah, A., Jing, H., Kambhatla, N., and Roukos, S. (2004). A mention-synchronous coreference resolution algorithm based on the bell tree. In *ACL 2004*.

Minnen, G., Caroll, J., and Pearce, D. (2001). Applied morphological processing of English. *Natural Language Engineering*, 7(3):207–223.

Morton, T. S. (2000). Coreference for NLP applications. In *Proc. ACL 2000*.

Moschitti, A. (2006). Making tree kernels practical for natural language learning. In *Proc. EACL 2006*.

Ng, V. and Cardie, C. (2002). Improving machine learning approaches to coreference resolution. In *40th Annual Meeting of the Asssociation for Computational Linguistics*.

Pianta, E. and Zanoli, R. (2007). Tagpro: A system for italian pos tagging based on svm. In *Proceedings of AI*IA 2007*.

Ponzetto, S. P. and Strube, M. (2006). Exploiting semantic role labeling, WordNet and Wikipedia for coreference resolution. In *Proc. HLT/NAACL 2006*.

Rosenfeld, R. (1996). A maximum entropy approach to adaptive statistical language modeling. *Computer Speech and Language*, 10:187–228.

Schmid, H., Fitschen, A., and Heid, U. (2004). SMOR: A German computational morphology covering derivation, composition and inflection. In *Proceedings of LREC 2004*.

Soon, W. M., Ng, H. T., and Lim, D. C. Y. (2001). A machine learning approach to coreference resolution of noun phrases. *Computational Linguistics*, 27(4):521–544.

Versley, Y. (2006). A constraint-based approach to noun phrase coreference resolution in German newspaper text. In *Konferenz zur Verarbeitung Natürlicher Sprache (KONVENS 2006)*.

Witten, I. and Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.

Yang, X., Su, J., and Tan, C. L. (2005). A twin-candidate model of coreference resolution with non-anaphor identification capability. In *Second International Joint Conference on Natural Language Processing (IJCNLP 2005)*.

Yang, X., Zhou, G., Su, J., and Tan, C. L. (2003). Coreference resolution using competition learning approach. In *ACL-2003*.

Zanchetta, E. and Baroni, M. (2005). Morph-it! A free corpus-based morphological resource for the Italian language. In *Corpus Linguistics 2005*.