Machine Learning for Relation Extraction

Feiyu Xu

Motivations of ML

- Porting to new domains or applications is expensive
- Current technology requires IE experts
  - Expertise difficult to find on the market
  - SME cannot afford IE experts
- Machine learning approaches
  - Domain portability is relatively straightforward
  - System expertise is not required for customization
  - “Data driven” rule acquisition ensures full coverage of examples

Problems

- Training data may not exist, and may be very expensive to acquire
- Large volume of training data may be required
- Changes to specifications may require reannotation of large quantities of training data
- Understanding and control of a domain adaptive system is not always easy for non-experts

Parameters

- Document structure
  - Free text
  - Semi-structured
  - Structured
- Richness of the annotation
  - Shallow NLP
  - Deep NLP
- Complexity of the template filling rules
  - Single slot
  - Multi slot
- Amount of data
- Degree of automation
  - Semi-automatic
  - Supervised
  - Semi-Supervised
  - Unsupervised
- Human interaction/contribution
- Evaluation/validation
  - during learning loop
  - Performance: recall and precision

Documents

- Unstructured (Free) Text
  - Regular sentences and paragraphs
  - Linguistic techniques, e.g., NLP
- Structured Text
  - Itemized information
  - Uniform syntactic clues, e.g., table understanding
- Semi-structured Text
  - Ungrammatical, telegraphic (e.g., missing attributes, multi-value attributes, …)
  - Specialized programs, e.g., wrappers

Outline

- Free text
  - Supervised and semi-automatic
    - AutoSlog
  - Semi-Supervised
    - AutoSlog-TS
  - Minimally supervised
    - ExDisco
    - DARE
- Semi-structured and unstructured text
  - NLP-based wrapping techniques
    - RAPIER

Free Text

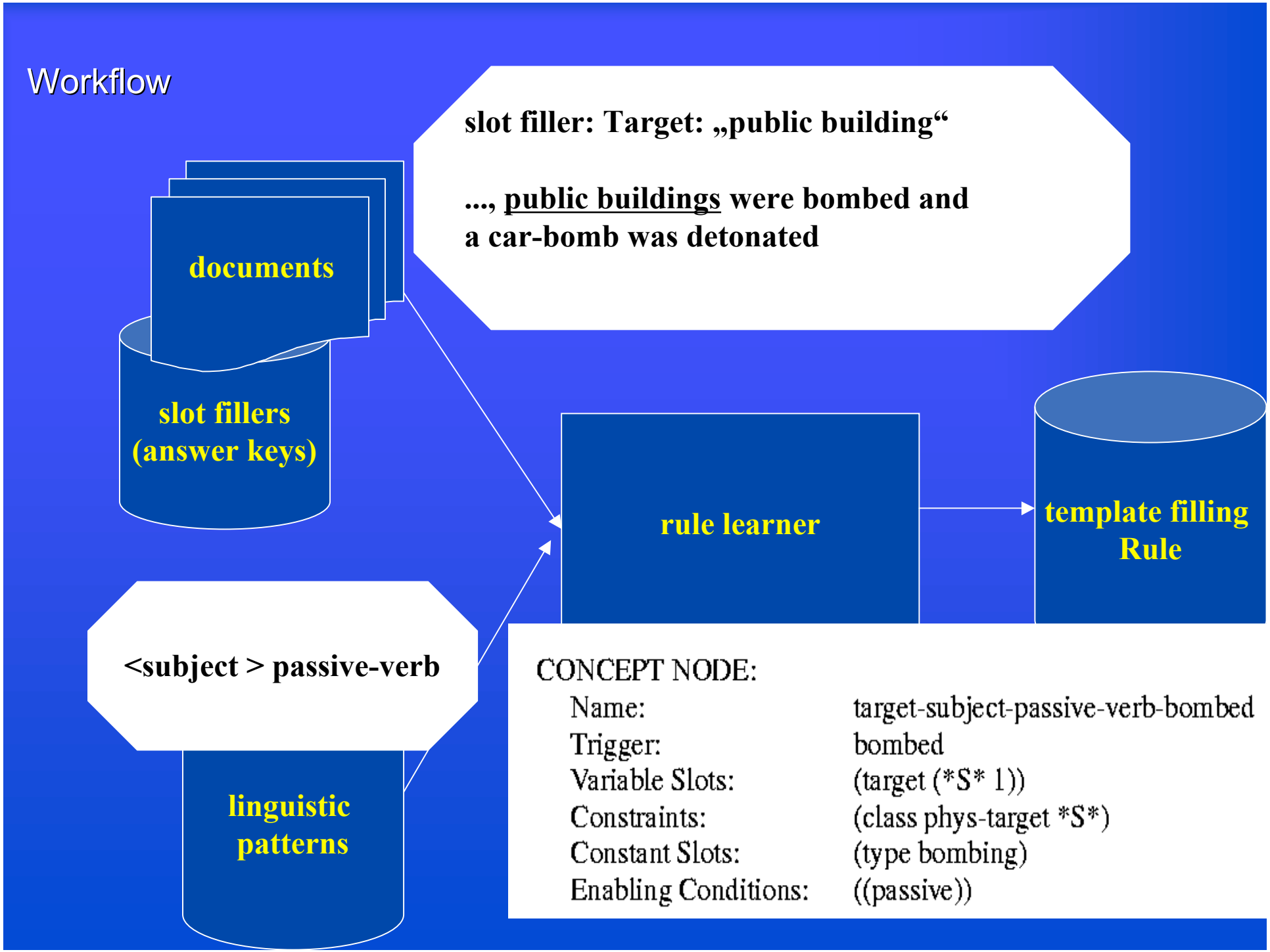
Supervised ML Approaches to Learn Relation Extraction Rules

NLP-based Supervised Approaches

- Input is an annotated corpus
  - Documents with associated templates
- A parser
  - Chunk parser
  - Full sentence parser
- Learning the mapping rules
  - From linguistic constructions to template fillers

AutoSlog (1993)

- Extracting a concept dictionary for template filling
- Full sentence parser
- One slot filler rules
- Domain adaptation performance
  - Before AutoSlog: hand-crafted dictionary
    - two highly skilled graduate students
    - 1500 person-hours
  - AutoSlog:
    - A dictionary for the terrorist domain: 5 person hours
    - 98% performance achievement of the hand-crafted dictionary


Workflow
slot filler: Target: „public building“
..., public buildings were bombed and
a car-bomb was detonated
documents
slot fillers
(answer keys)
rule learner
template filling
Rule
<subject > passive-verb
linguistic
patterns
CONCEPT NODE:
Name: target-subject-passive-verb-bombed
Trigger: bombed
Variable Slots: (target (*S* 1))
Constraints: (class phys-target *S*)
Constant Slots: (type bombing)
Enabling Conditions: ((passive))

Linguistic Patterns

Linguistic Pattern	Example
<subject> passive-verb	<victim> was murdered
<subject> active-verb	<perpetrator> bombed
<subject> verb infinitive	<perpetrator> attempted to kill
<subject> auxiliary noun	<victim> was victim
passive-verb <dobj>[1]	killed <victim>
active-verb <dobj>	bombed <target>
infinitive <dobj>	to kill <victim>
verb infinitive <dobj>	threatened to attack <target>
gerund <dobj>	killing <victim>
noun auxiliary <dobj>	fatality was <victim>
noun prep <np>	bomb against <target>
active-verb prep <np>	killed with <instrument>

Id: DEV-MUC4-1192 **Slot filler:** "gilberto molasco"
Sentence: (they took 2-year-old gilberto molasco, son of patricio rodriguez, and 17-year-old andres argueta, son of emimesto argueta.)

CONCEPT NODE

Name:	victim-active-verb-dobj-took
Trigger:	took
Variable Slots:	(victim (*DOBJ* 1))
Constraints:	(class victim *DOBJ*)
Constant Slots:	(type kidnapping)
Enabling Conditions:	((active))

A bad concept node definition

Error Sources

- A sentence contains the answer key string but does not contain the event
- The sentence parser delivers wrong results
- A heuristic proposes a wrong conceptual anchor

Training Data

- MUC-4 corpus
- 1500 texts
- 1258 answer keys
- 4780 string fillers
- 1237 concept node definition

- Human in loop for validation to filter out bad and wrong definitions: 5 hours

- 450 concept nodes left after human review

System/Test Set	Recall	Precision	F-measure
MUC-4/TST3	46	56	50.51
AutoSlog/TST3	43	56	48.65
MUC-4/TST4	44	40	41.90
AutoSlog/TST4	39	45	41.79

Comparative Results

Summary

- Advantages
  - Semi-automatic
  - Less human effort
- Disadvantages
  - Human interaction
  - Still very naive approach
  - Need a big amount of annotation
  - Domain adaptation bottelneck is shifted to human annotation
  - No generation of rules
  - One slot filling rule
  - No mechanism for filtering out bad rules

Other NLP-based ML Approaches

- LIEP (Huffman, 1995)
- PALKA (Kim & Moldovan, 1995)
- HASTEN (Krupka, 1995)
- CRYSTAL (Soderland et al., 1995)

LIEP [1995]

The Parliament building was bombed by ***Carlos***.

TARGET-was-bombed-by-PERPETRATOR:
  noun-group(TRGT, head(isa(physical-target))),
  noun-group(PERP, head(isa(perpetrator)))
  verb-group(VG, type(passive), head(bombed))
  preposition(PREP, head(by))

  subject(TRGT, VG),
  post-verbal-prep(VG, PREP),
  prep-object(PREP, PERP)
  $\Longrightarrow$ bombing-event(BE, target(TRGT), agent(PERP))

PALKA [1995]

The Parliament building was bombed by ***Carlos***.

FP-structure = MeaningFrame + PhrasalPattern

Meaning Frame: (BOMBING agent: ANIMATE
target: PHYS-OBJ
instrument: PHYS-OBJ
effect: STATE)

Phrasal Pattern: ((PHYS-OBJ) was bombed by (PERP))

FP-structure:
(BOMBING target: PHYS-OBJ
agent: PERP
pattern: ((target) was bombed by (agent))

HASTEN [1995]

The Parliament building was bombed by ***Carlos***.

BOMBING:	
TARGET:	NP "semantic = physical-object"
ANCHOR:	VG "root = bomb"
PERPETRATOR:	NP "semantic = terrorist-group"

- Egraphs
- (*SemanticLabel, StructuralElement*)

CRYSTAL [1995]

The Parliament building was bombed by ***Carlos***.

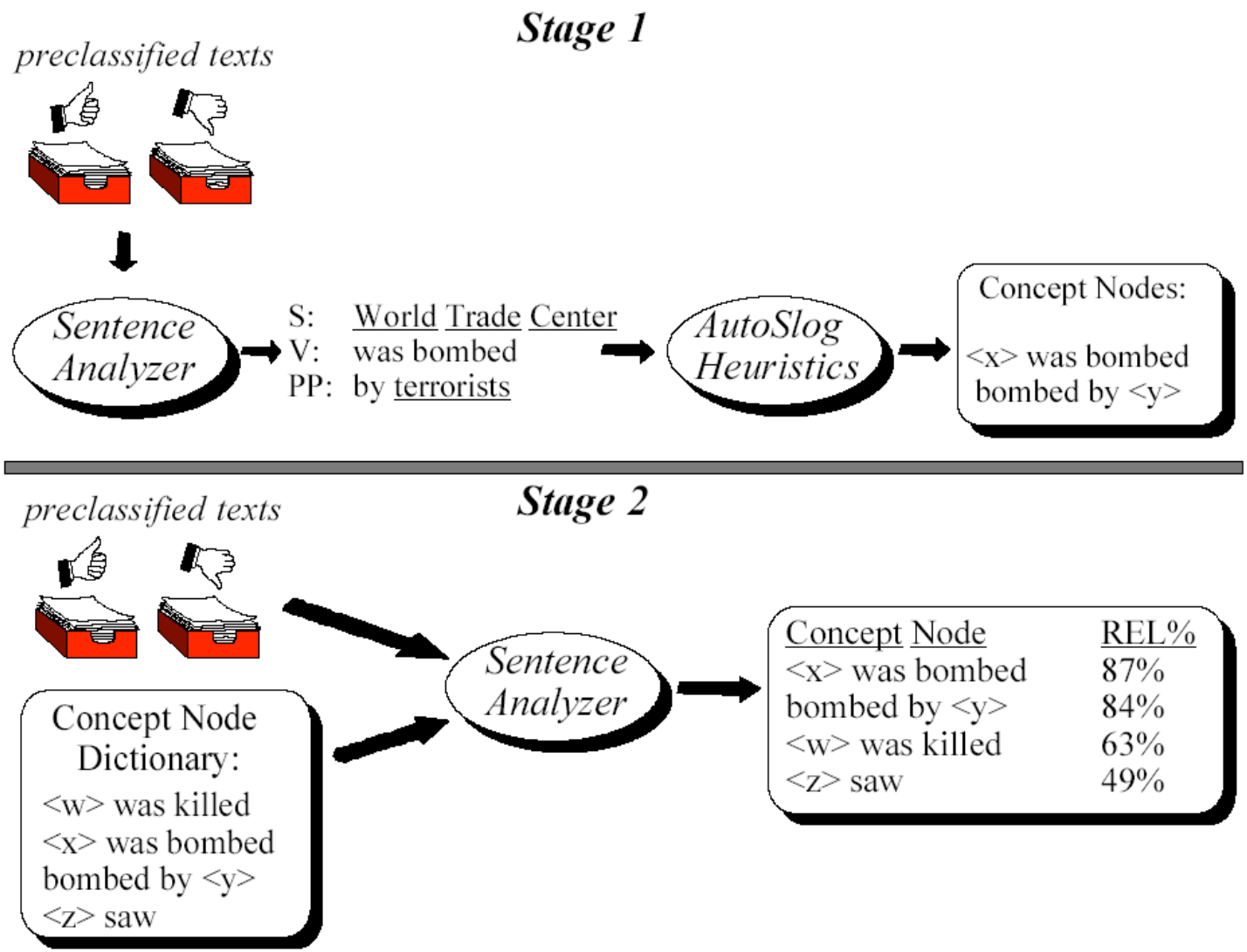
Concept type: BUILDING BOMBING

SUBJECT:	Classes include:	<PhysicalTarget>
	Terms include:	BUILDING
	Extract:	*target*
VERB:	Root:	BOMB
	Mode:	passive
PREPOS-PHRASE:	Preposition:	BY
	Classes include:	<PersonName>
	Extract:	*perpetrator name*

Semi-Supervised Approaches

AutoSlog TS [Riloff, 1996]

- Input
  - pre-classified documents (relevant vs. irrelevant)
- NLP
  - full parser for detecting linguistic structures
- Principle
  - relevant patterns are patterns occuring more often in the relevant documents
- Output
  - ranked linguistic patterns without semantic labelling
- Learning process
  - pattern generation and
  - statistical filtering
- Quality control: manual review of the results



AutoSlog-TS flowchart

Pattern Extraction

The sentence analyzer produces a syntactic analysis for each sentence and identified noun phrases. For each noun phrase, the heuristic rules generate a pattern to extract noun phrase.

<subject> bombed

Relevance Filtering

- the whole text corpus will be processed a second time using the extracted patterns obtained by stage 1.
- Then each pattern will be assigned with a relevance rate based on its occurring frequency in the relevant documents relatively to its occurrence in the total corpus.
- A preferred pattern is the one which occurs more often in the relevant documents.

Statistical Filtering

Relevance Rate:

$$Pr(\text{relevant text} \setminus \text{text contains case frame}_i) = \frac{\text{rel-freq}_i}{\text{total-freq}_i}$$

$\text{rel-freq}_{i:}$ number of instances of case-frame_i *in the relevant documents*

$\text{total-freq}_{i:}$ total number of instances of case-frame_i

Ranking Function:

$$\text{score}_i = \text{relevance rate}_i * \log_2(\text{frequency}_i)$$

$Pr < 0{,}5$ *negatively correlated with the domain*

„Top“ AutoSlog-TS Patterns

1. <subj> exploded	14. <subj> occurred
2. murder of <np>	15. <subj> was located
3. assassination of <np>	16. took_place on <np>
4. <subj> was killed	17. responsibility for <np>
5. <subj> was kidnapped	18. occurred on <np>
6. attack on <np>	19. was wounded in <np>
7. <subj> was injured	20. destroyed <dobj>
8. exploded in <np>	21. <subj> was murdered
9. death of <np>	22. one of <np>
10. <subj> took_place	23. <subj> kidnapped
11. caused <dobj>	24. exploded on <np>
12. claimed <dobj>	25. <subj> died
13. <subj> was wounded	

The Top 25 Extraction Patterns

AutoSlog Patterns

Linguistic Pattern	Example
<subject> passive-verb	<victim> was murdered
<subject> active-verb	<perpetrator> bombed
<subject> verb infinitive	<perpetrator> attempted to kill
<subject> auxiliary noun	<victim> was victim
passive-verb <dobj>[1]	killed <victim>
active-verb <dobj>	bombed <target>
infinitive <dobj>	to kill <victim>
verb infinitive <dobj>	threatened to attack <target>
gerund <dobj>	killing <victim>
noun auxiliary <dobj>	fatality was <victim>
noun prep <np>	bomb against <target>
active-verb prep <np>	killed with <instrument>

Empirical Results

- 1500 MUC-4 texts, 50% are relevant.
- In stage 1, 32,345 unique extraction patterns.
- A user reviewed the top 1970 patterns in about 85 minutes and kept the best 210 patterns.
- Evaluation
    - AutoSlog and AutoSlog-TS systems return comparable performance.

Conclusion

- Advantages
  - Pioneer approach to automatic learning of extraction patterns
  - Reduce the manual annotation
- Disadvantages
  - Ranking function is too dependent on the occurrence of a pattern, relevant patterns with low frequency can not float to the top
  - Only patterns, not classification

Minimally Supervised Approachs

ExDisco (Yangarber 2001)

- Seed
- Bootstrapping
- Duality/Density Principle for validation of each iteration

Input

- a corpus of unclassified and unannotated documents
- a seed of patterns, e.g.,

subject(company)-verb(appoint)-object(person)

NLP as Preprocessing

- full parser for detecting subject-v-object relationships
  - NE recognition
  - Functional Dependency Grammar (FDG) formalism (Tapannaien & Järvinen, 1997)

Duality/Density Principle (boostrapping)

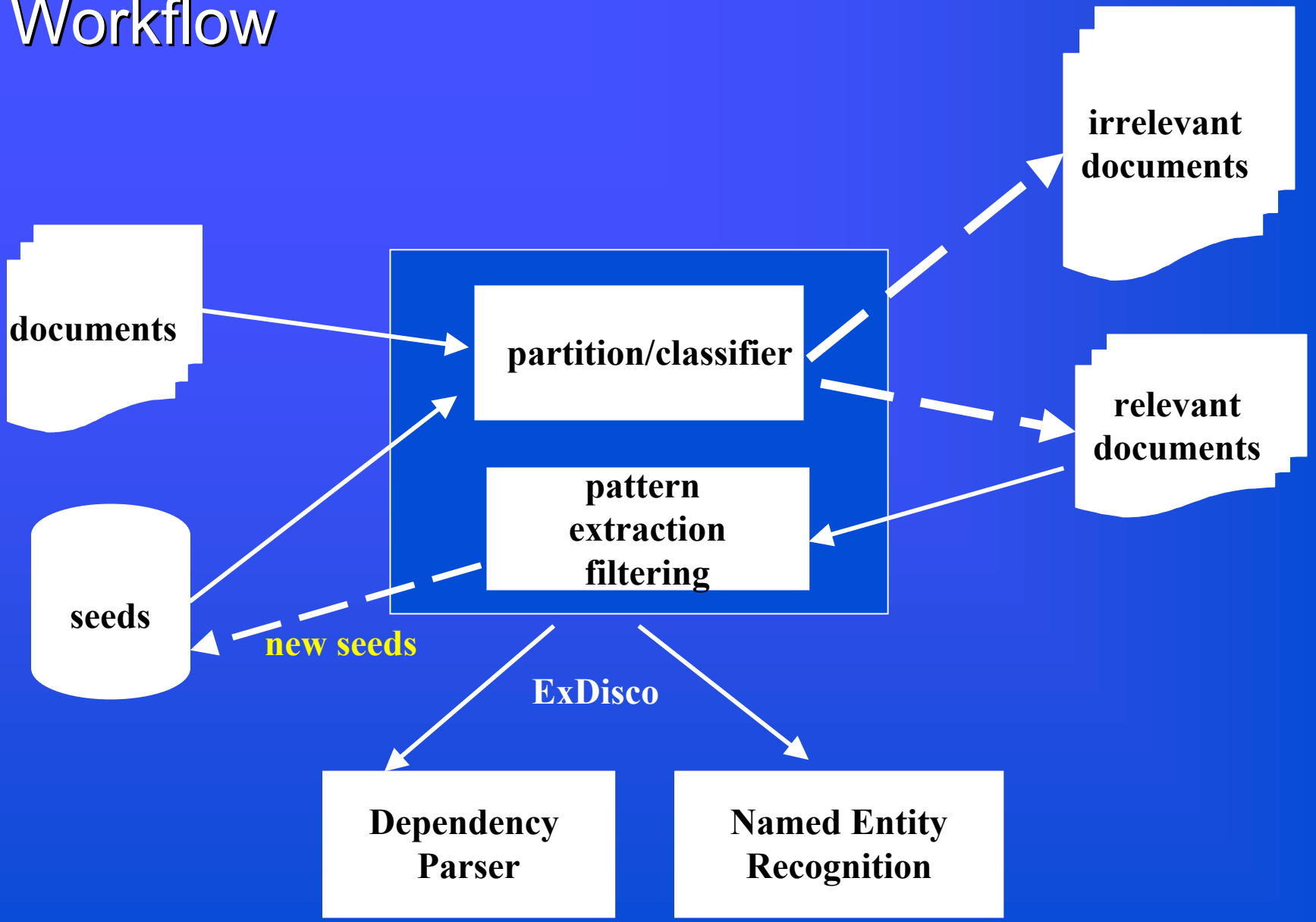
- Density:
  - Relevant documents contain more relevant patterns

- Duality:
  - documents that are relevant to the scenario are strong indicators of good patterns
  - good patterns are indicators of relevant documents

Algorithm

- Given:
  - a large corpus of un-annotated and un-classified documents
  - a trusted set of scenario patterns, initially chosen ad hoc by the user, the seed. Normally is the seed relatively small, two or three
  - (possibly empty) set of concept classes
- Partition
  - applying seed to the documents and divide them into relevant and irrelevant documents
- Search for new candidate patterns:
  - automatic convert each sentence into a set of candidate patterns.
  - choose those patterns which are strongly distributed in the relevant documents
  - Find new concepts
- User feedback
- Repeat

Workflow


irrelevant documents
documents
partition/classifier
relevant documents
pattern extraction filtering
seeds
new seeds
ExDisco
Dependency Parser
Named Entity Recognition

Pattern Ranking

$$\text{Score}(P) = \frac{|H \cap R|}{|H|} \cdot \text{LOG}(|H \cap R|)$$

Evaluation of Event Extraction

Pattern Base	*Recall*	*Precision*	*F*
Seed	27	74	39.58
ExDisco	52	72	60.16
Union	57	73	63.56
Manual-MUC	47	70	56.40
Manual-NOW	56	75	64.04

ExDisco

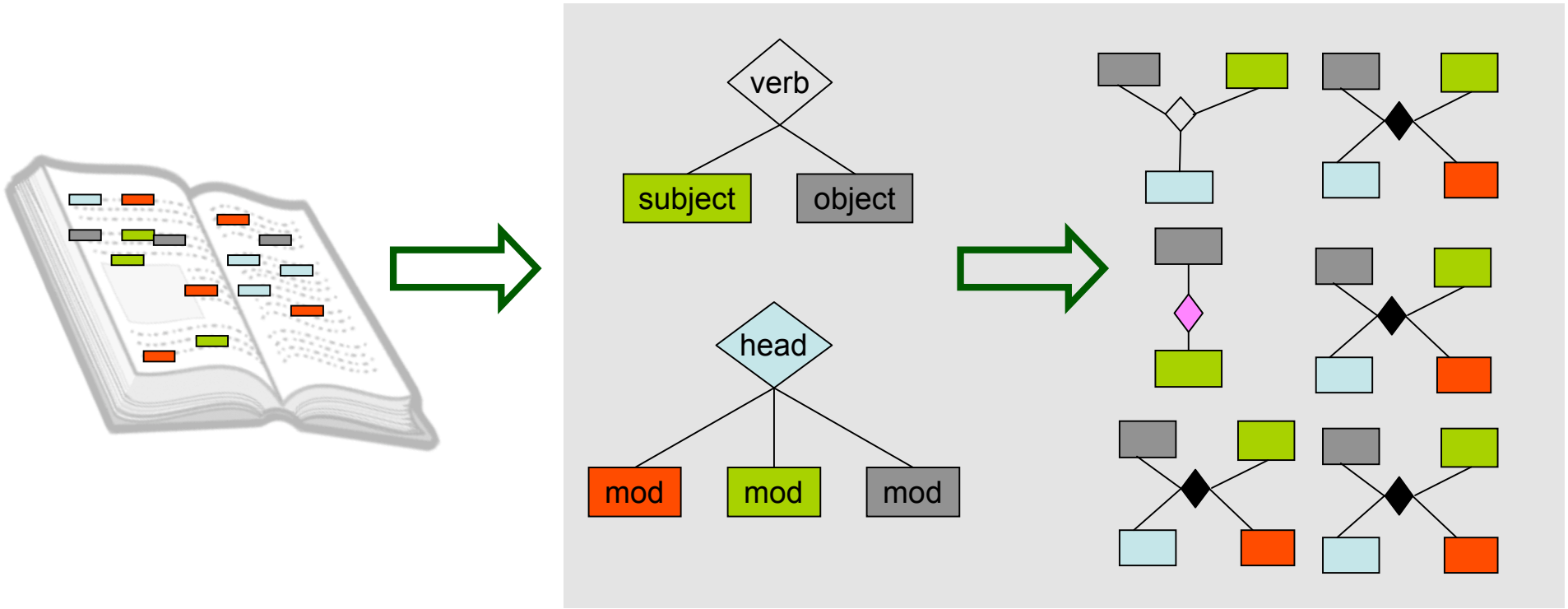
- Advantages
  - Only seed patterns are needed as initial knowledge
  - Multi-slot template filler rules
- Disadvantages
  - Only subject-verb-object patterns, local patterns are ignored
  - No generalization of pattern rules (see inductive learning)
  - Collocations are not taken into account, e.g., *PN take responsibility of Company*
- Evaluation methods
  - Event extraction: integration of patterns into IE system and test recall and precision
  - Qualitative observation: manual evaluation
  - Document filtering: using ExDisco as document classifier and document retrieval system

Bootstrapping Relation Extraction with Semantic Seeds

Feiyu Xu

Research Goal

Development of a general framework for automatically learning mappings between linguistic analyses and target semantic relations, with minimal human intervention.



Challenges

- Easy adaptation to new relation types with varied complexity
- Automatic learning without annotated corpus
- Exhaustive discovery of relevant linguistic patterns
- Integration of semantic role information into linguistic patterns

Outline

- State of the art
- **D**omain **A**daptive **R**elation **E**xtraction **F**ramework (**DARE**)
- Experiments and evaluations
- Performance analysis and discussion
- Conclusion and future work

Outline

- State of the art
- **D**omain **A**daptive **R**elation **E**xtraction **F**ramework (**DARE**)
- Experiments and evaluations
- Performance analysis and discussion
- Conclusion and future work

Example

A relation extraction task in the domain *management succession* (MUC-6)

< person_in, person_out, position, organisation>

- *person_in*: the person who obtained the position
- *person_out*: the person who left the position
- *position*: the job position that the two persons were involved in
- *organisation*: the organisation where the position was located

According to CEO Fred Bell, Acme Inc. hired Peter Smith as COO yesterday, replacing Hans Bloggs.

According to CEO Fred Bell, Acme Inc. hired Peter Smith as COO yesterday, replacing Hans Bloggs.

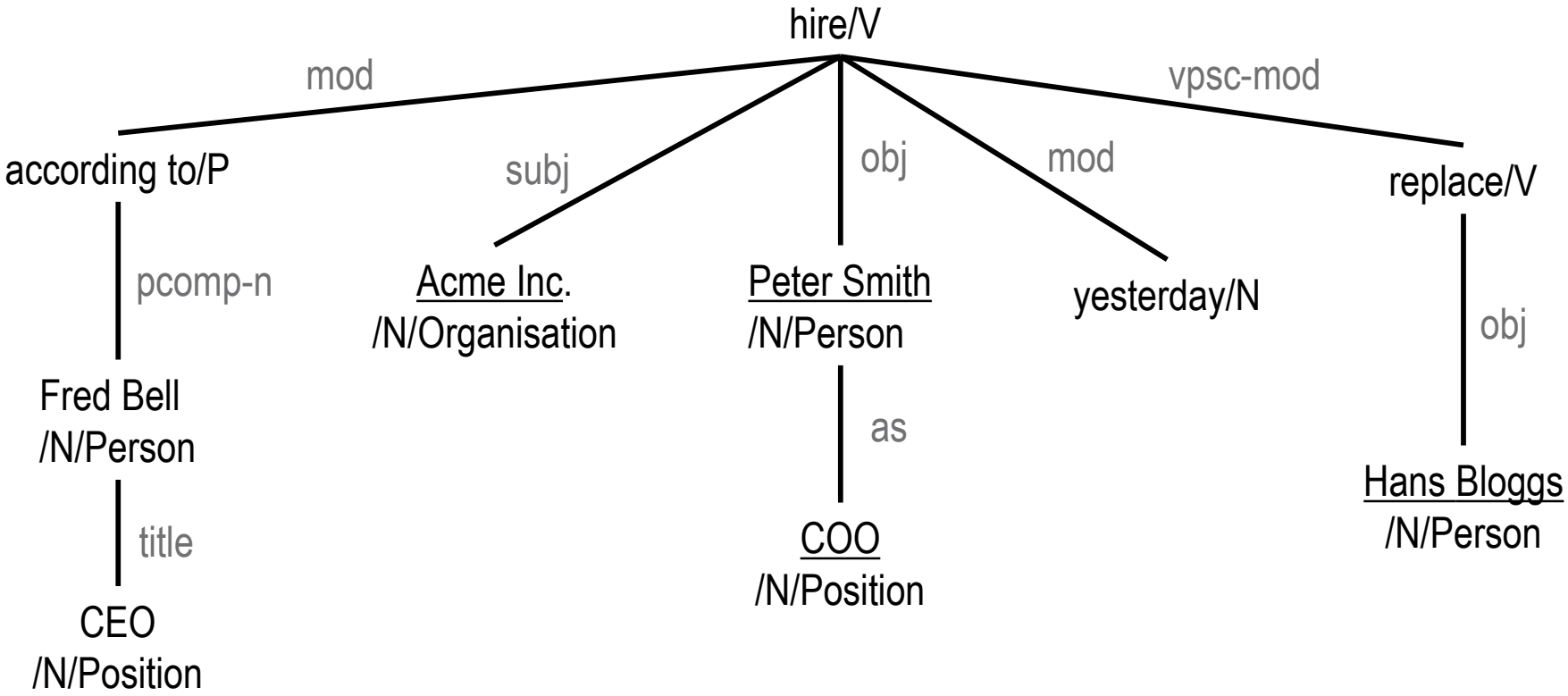
<person_in, person_out, position, organisation>

According to CEO Fred Bell, **Acme Inc.** hired **Peter Smith** as **COO** yesterday, replacing **Hans Bloggs**.

<person_in, person_out, position, organisation>

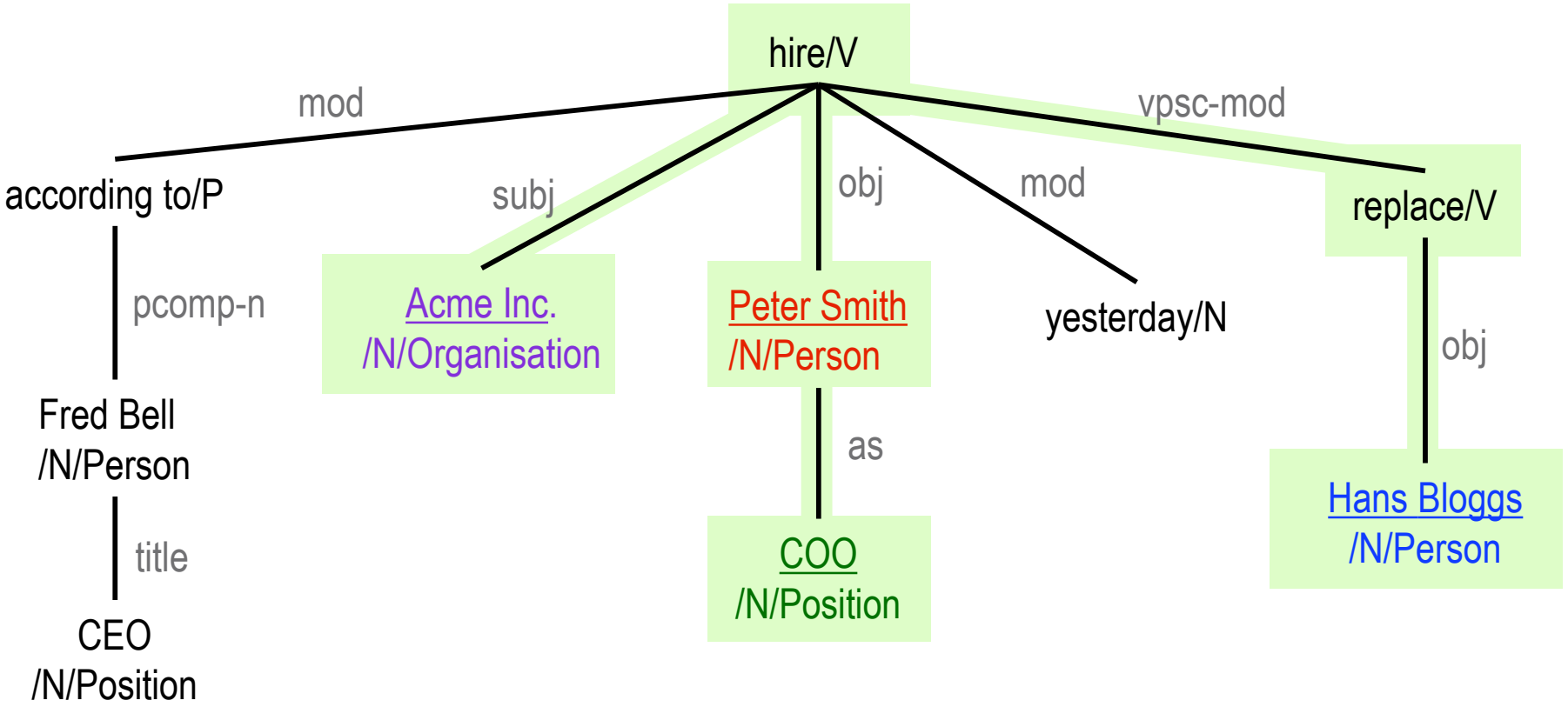
According to CEO Fred Bell, **Acme Inc.** hired **Peter Smith** as **COO** yesterday, replacing **Hans Bloggs**.

<person_in, person_out, position, organisation>

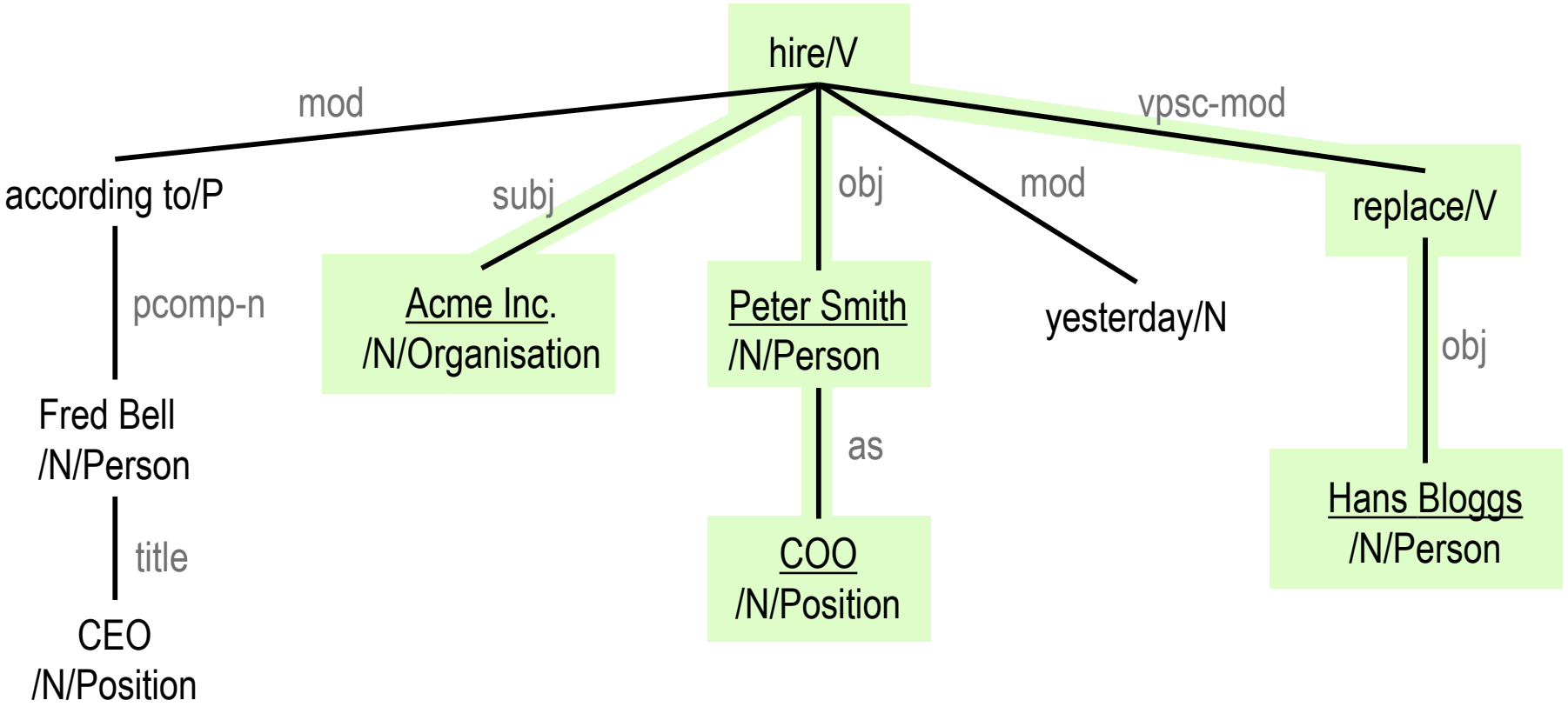


According to CEO Fred Bell, **Acme Inc.** hired **Peter Smith** as **COO** yesterday, replacing **Hans Bloggs**.

<person_in, person_out, position, organisation>



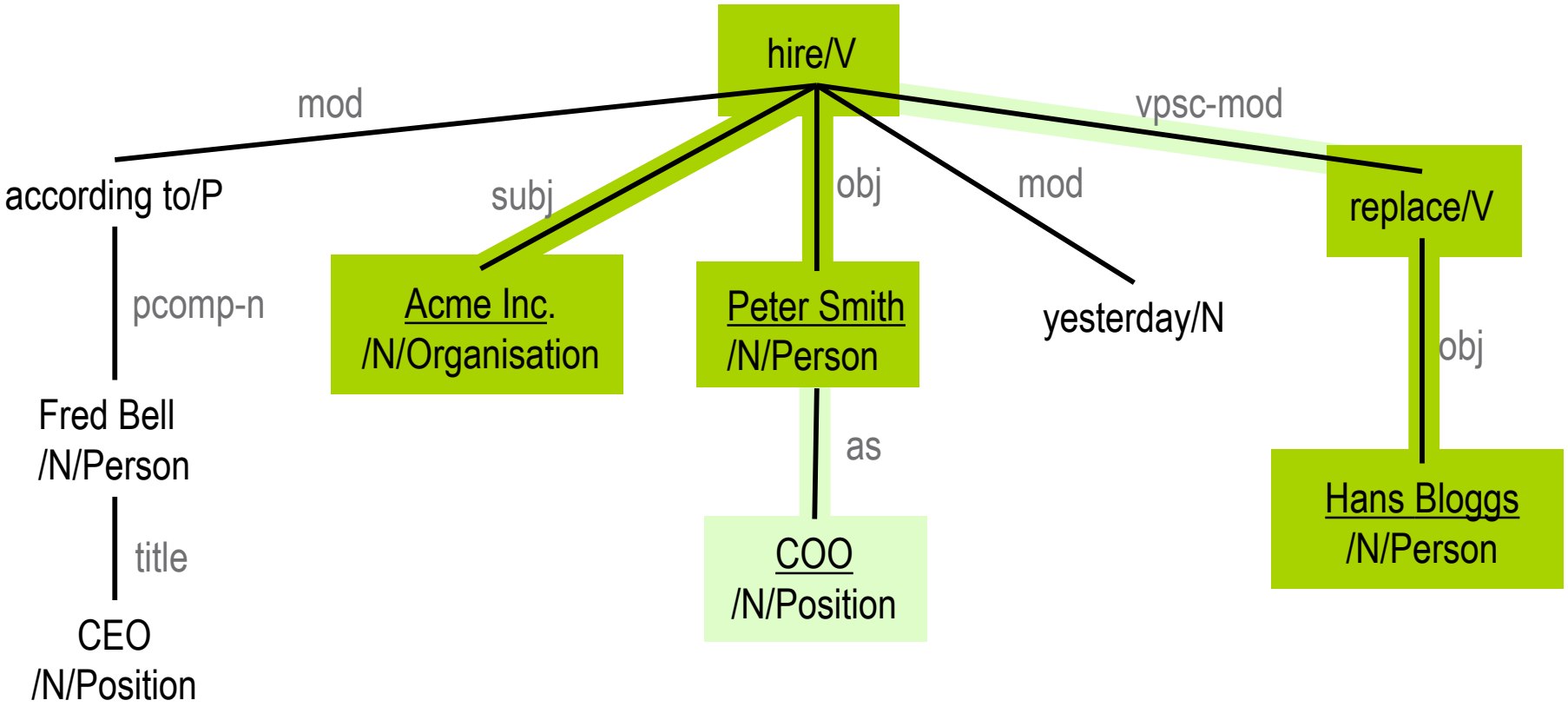
Ideal Target Pattern



Previous Work: SVO Model

Yangarber (2001)

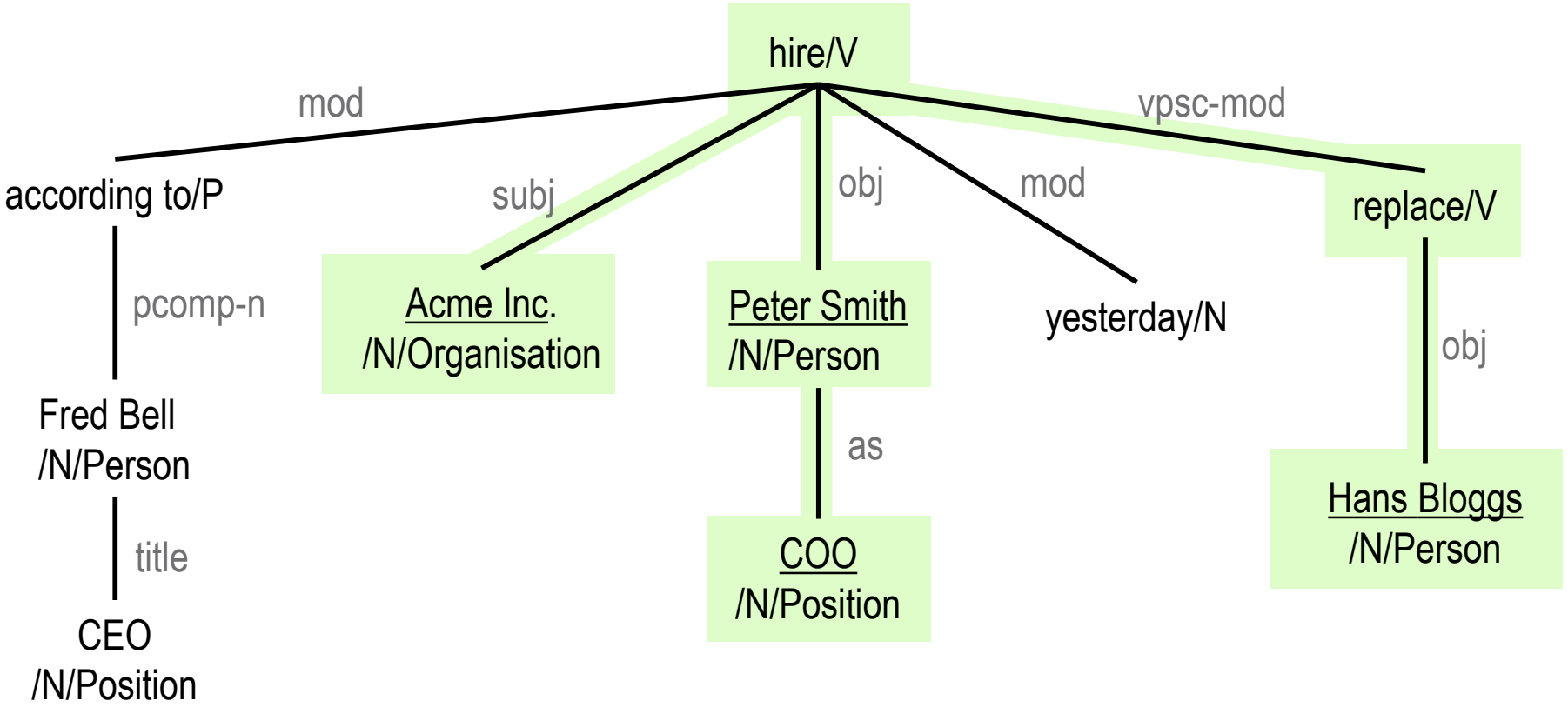
- Verb centered
- Direct relations between subject-verb-object
- Complex NP can not be extracted, e.g., the person and position relation
- The linguistic relations among patterns are not considered, e.g., hire and replace



Previous Work: Chain Model

Sudo et al. (2001)

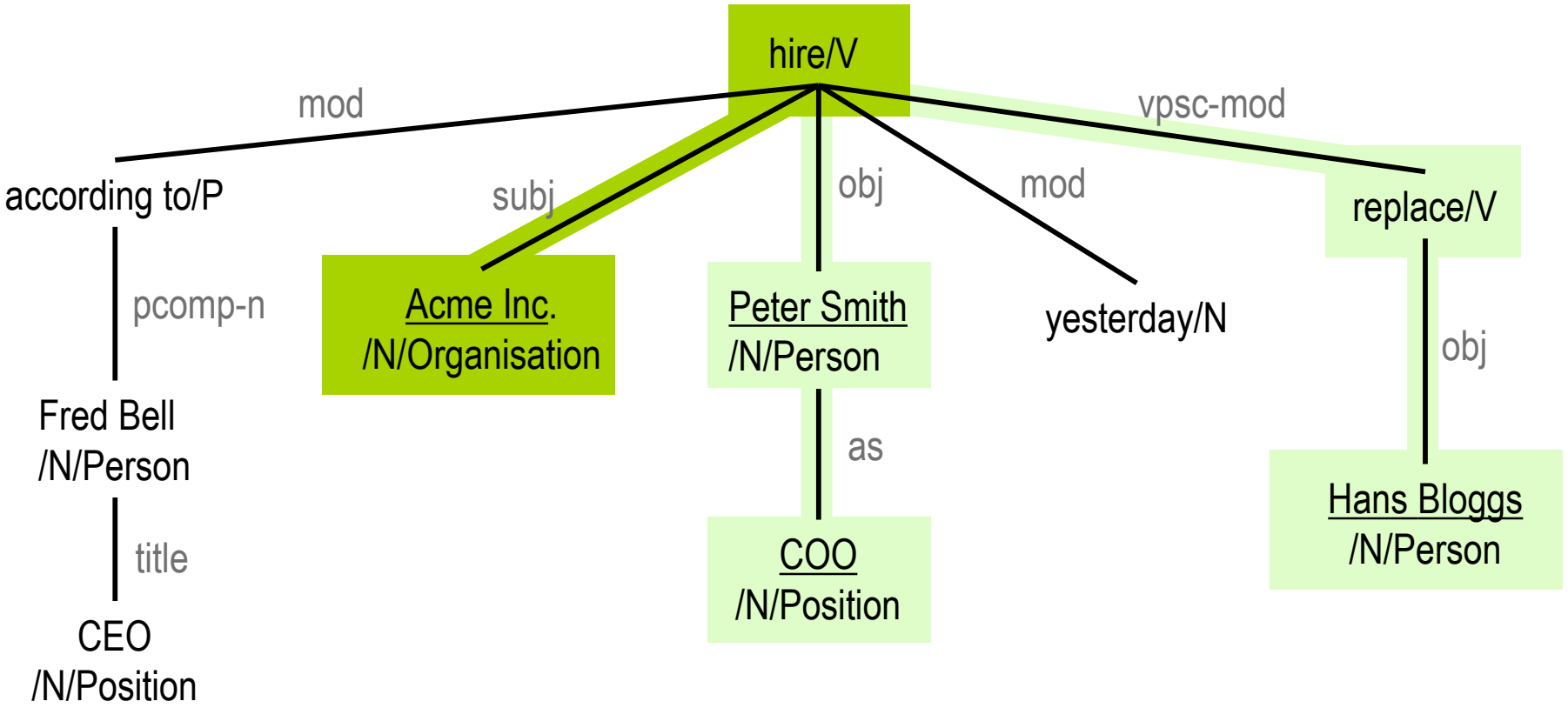
- Verb centered
- A single syntactic path dominated by a verb containing at least one relevant named entity concept



Previous Work: Chain Model

Sudo et al. (2001)

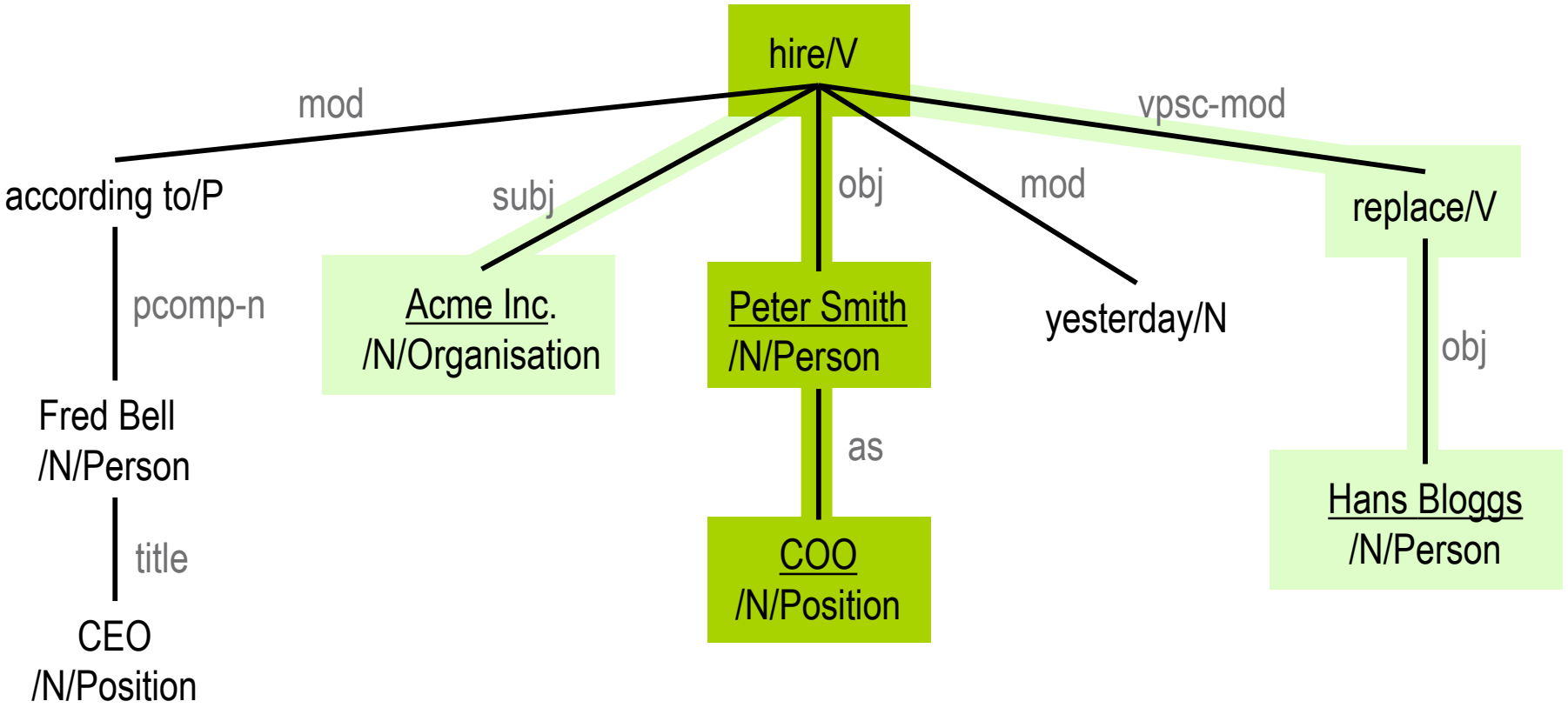
- Verb centered
- A single syntactic path dominated by a verb containing at least one relevant named entity concept



Previous Work: Chain Model

Sudo et al. (2001)

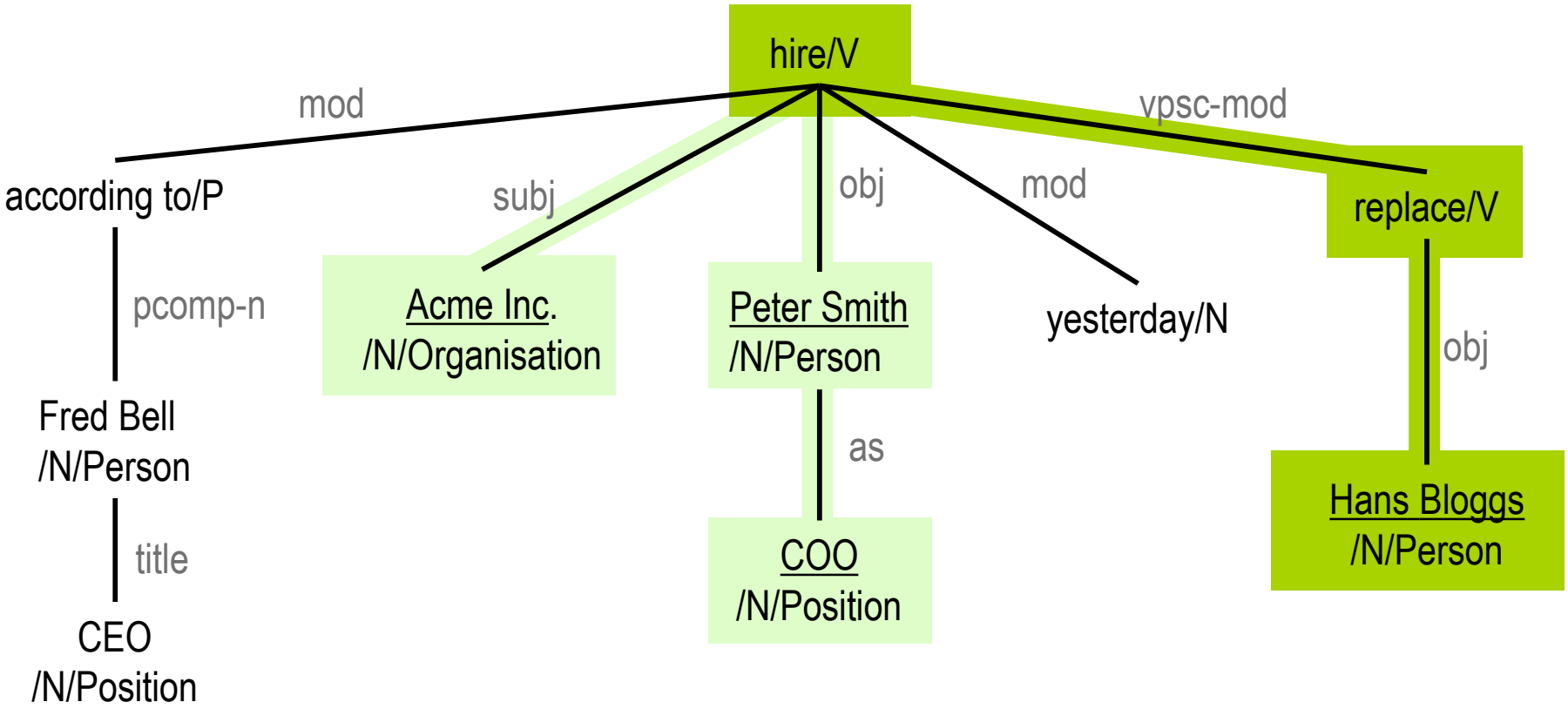
- Verb centered
- A single syntactic path dominated by a verb containing at least one relevant named entity concept



Previous Work: Chain Model

Sudo et al. (2001)

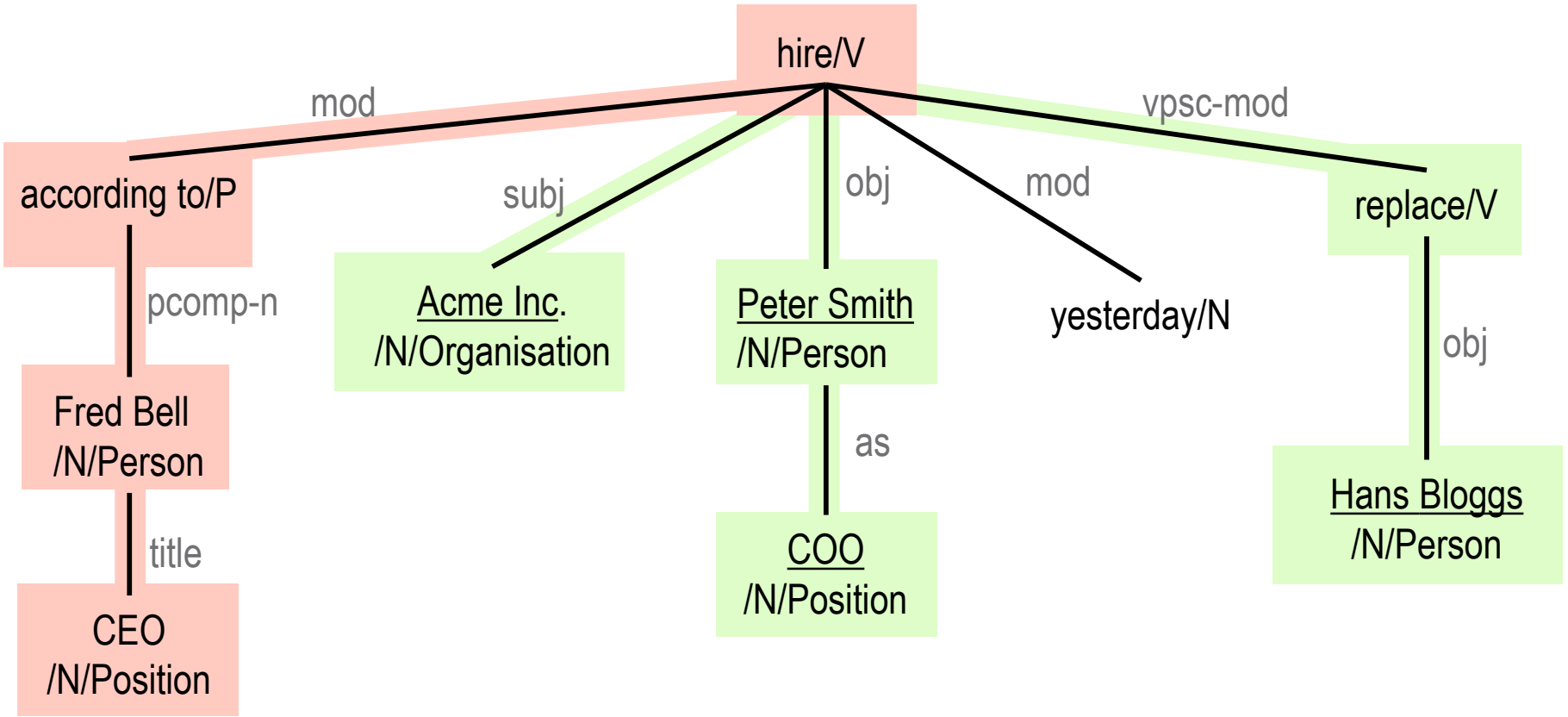
- Verb centered
- A single syntactic path dominated by a verb containing at least one relevant named entity concept



Previous Work: Chain Model

Sudo et al. (2001)

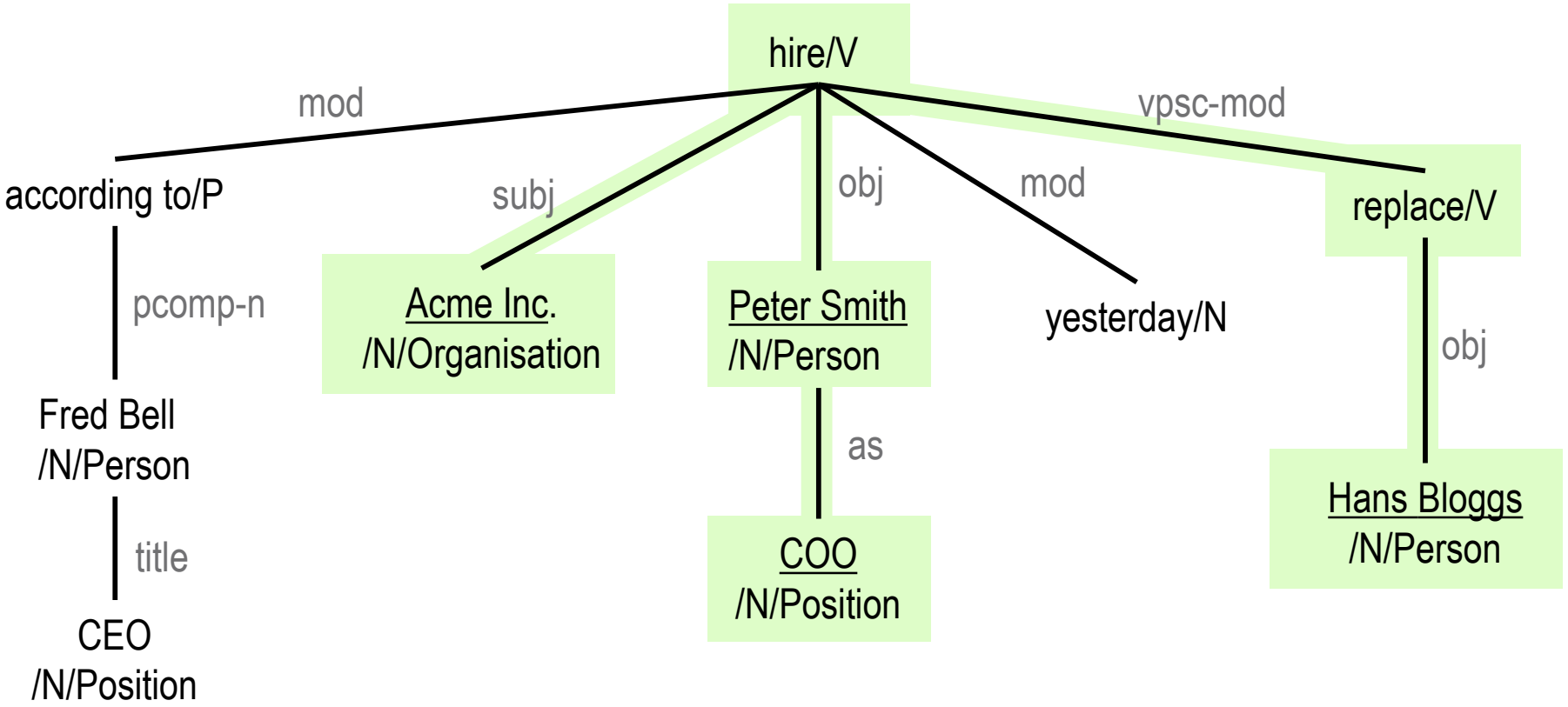
- Verb centered
- A single syntactic path dominated by a verb containing at least one relevant named entity concept



Previous Work: Linked Chain Model

Stevenson & Greenwood 2005

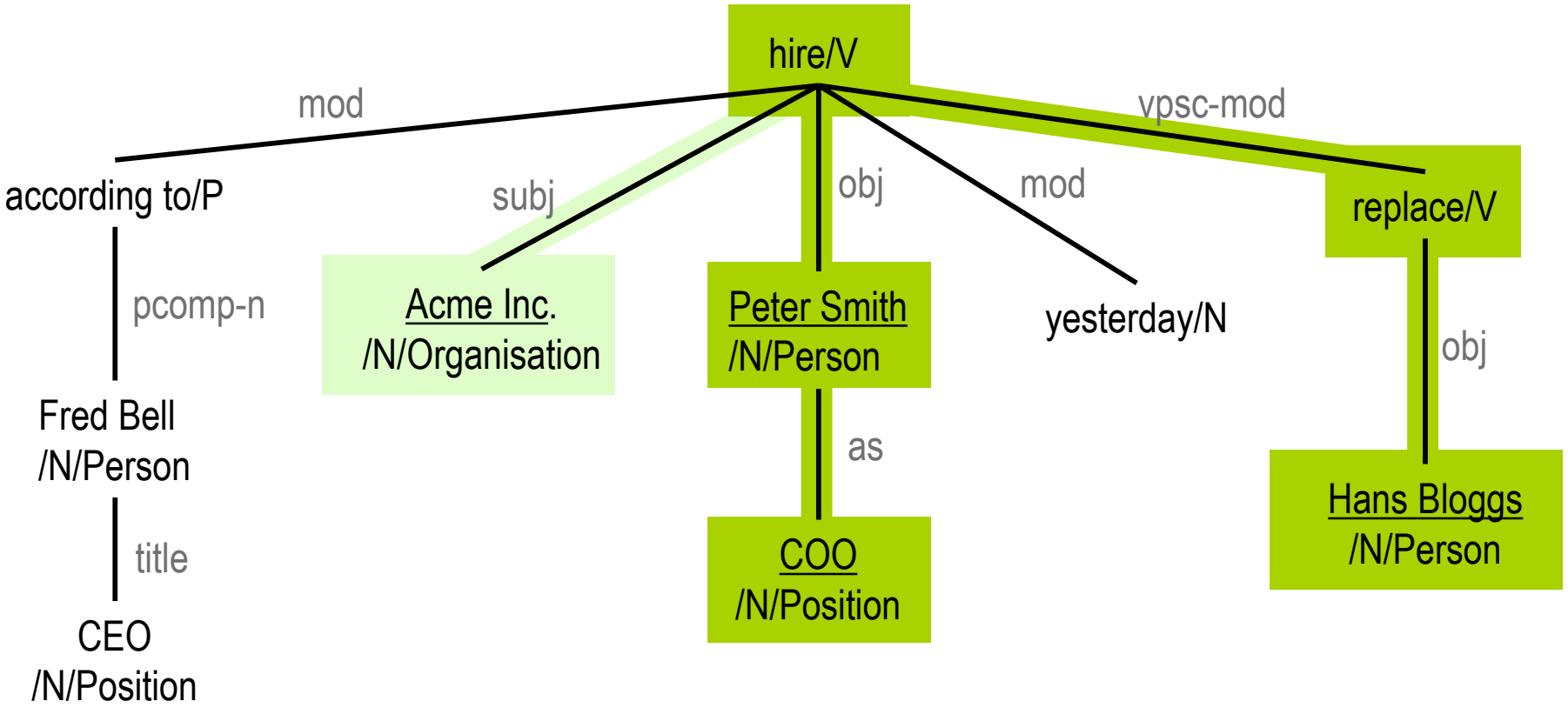
- verb centered
- pairs of chains instead of single paths



Previous Work: Linked Chain Model

Stevenson & Greenwood 2005

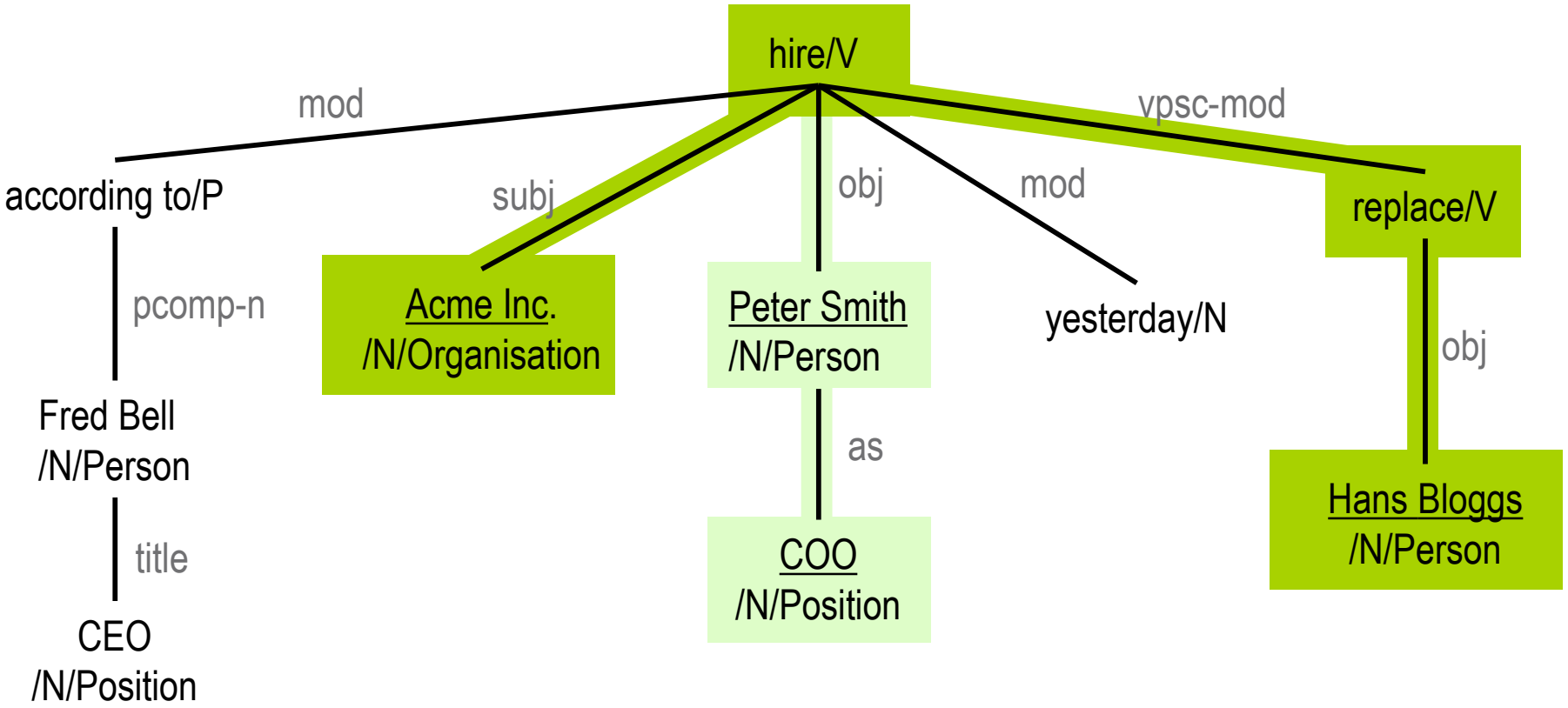
- verb centered
- pairs of chains instead of single paths



Previous Work: Linked Chain Model

Stevenson & Greenwood 2005

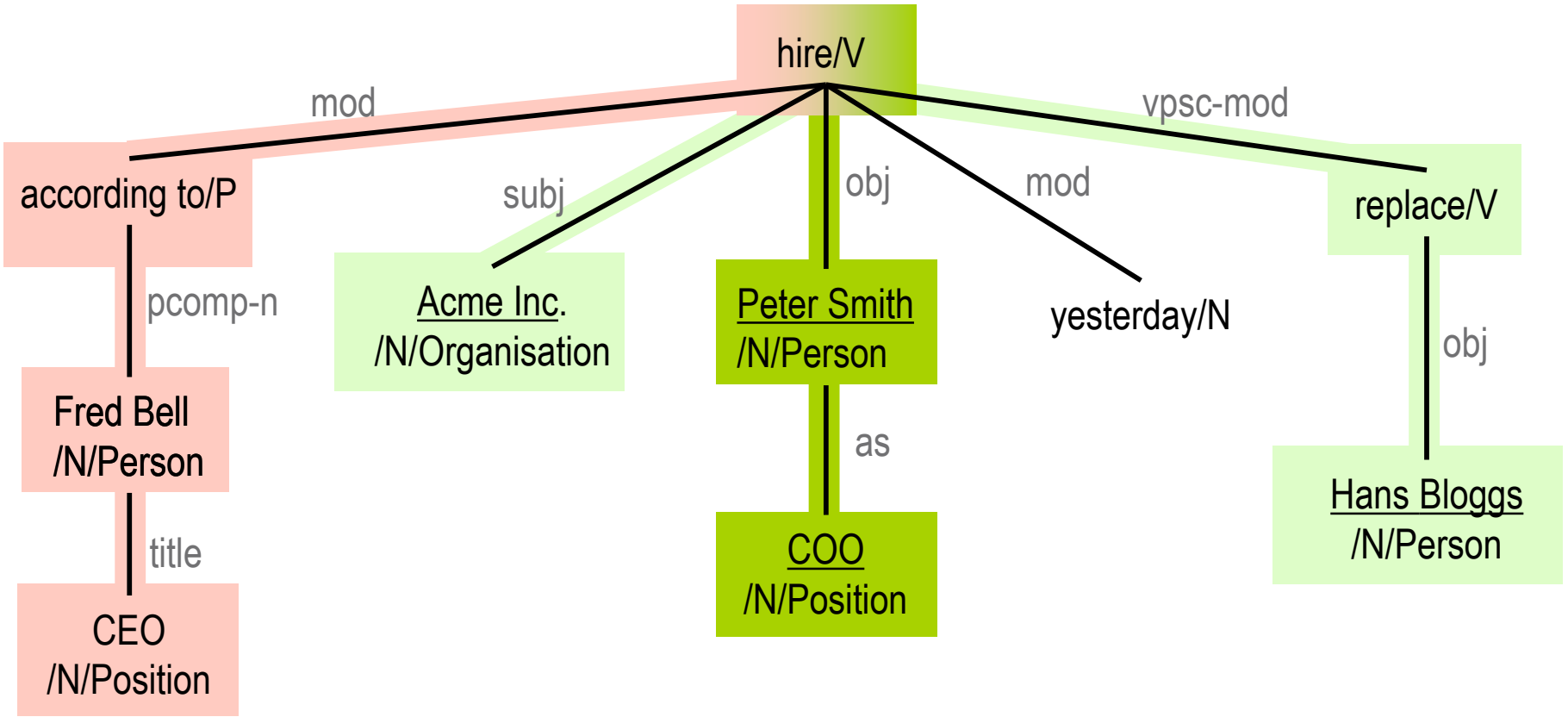
- verb centered
- pairs of chains instead of single paths



Previous Work: Linked Chain Model

Stevenson & Greenwood 2005

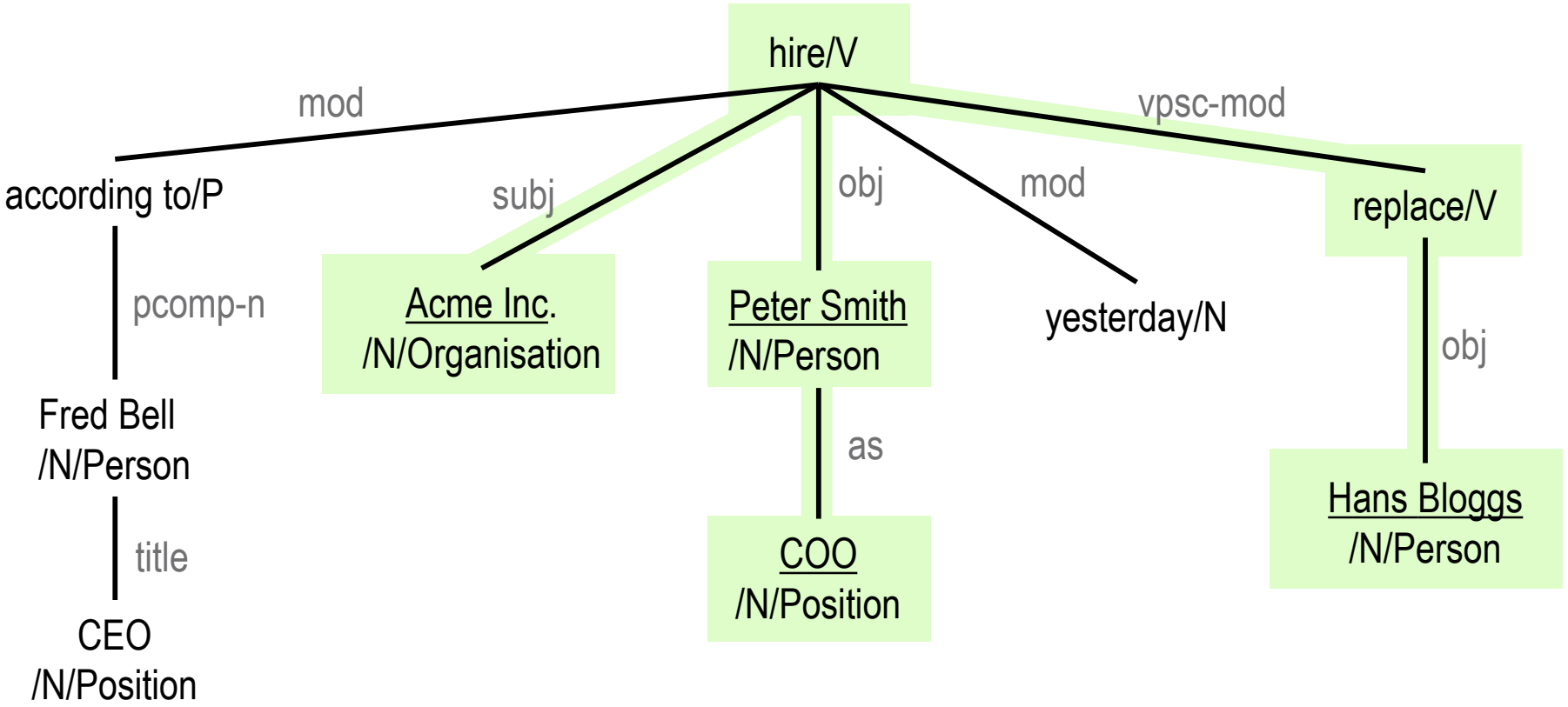
- verb centered
- pairs of chains instead of single paths



Previous Work: Subtree-Model

Sudo et al. (2003)

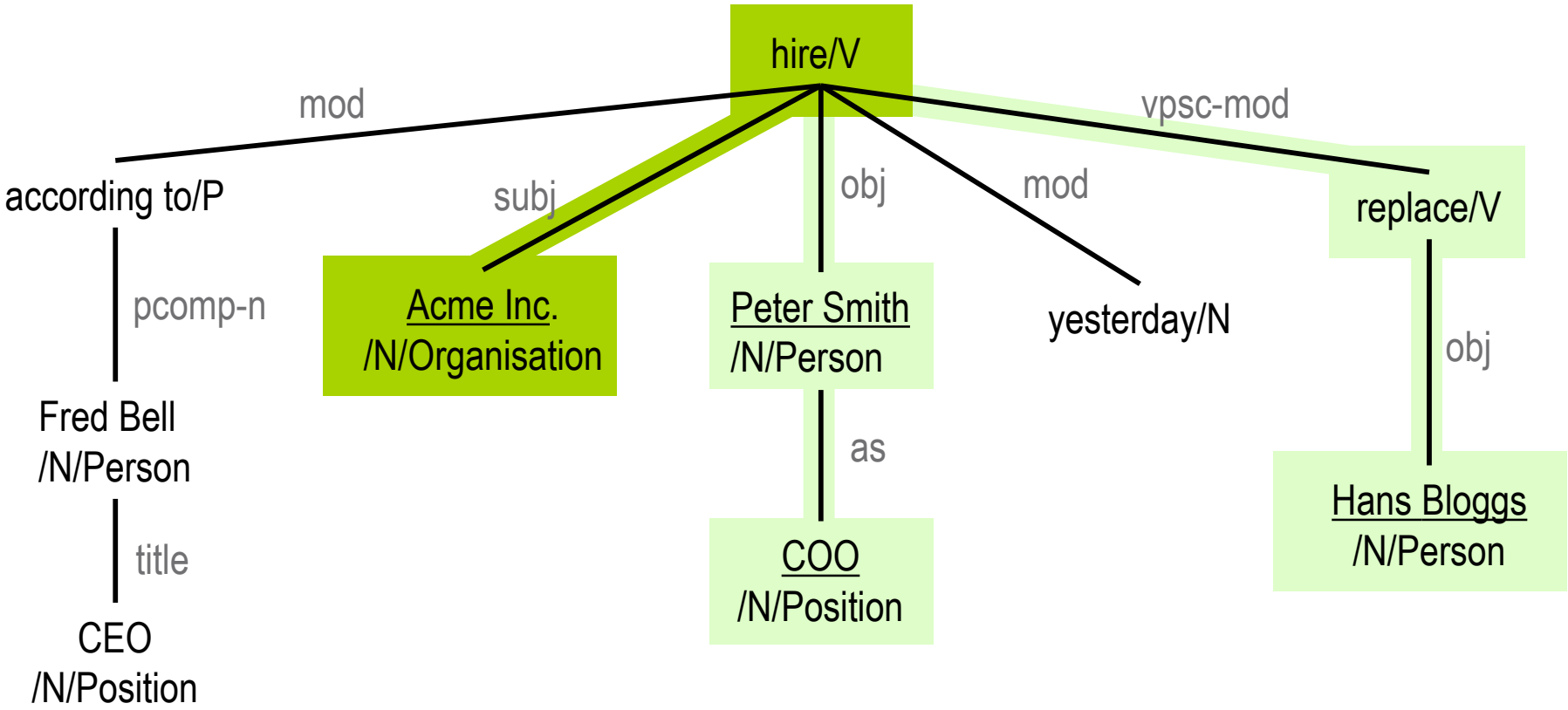
- verb centered
- All chains dominated by a verb, which contain at least one relevant named entity and their combinations



Previous Work: Subtree-Model

Sudo et al. (2003)

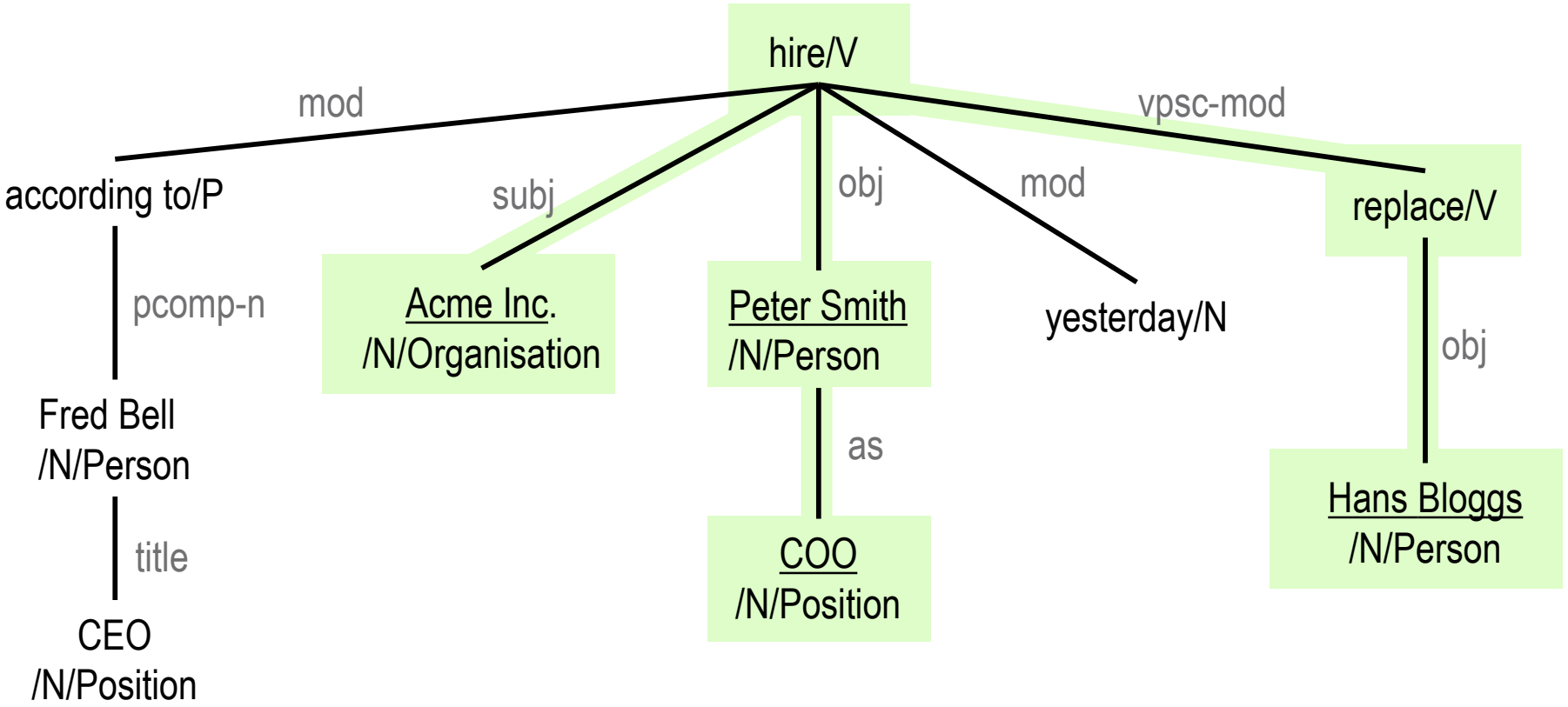
- verb centered
- All chains dominated by a verb, which contain at least one relevant named entity and their combinations



Previous Work: Subtree-Model

Sudo et al. (2003)

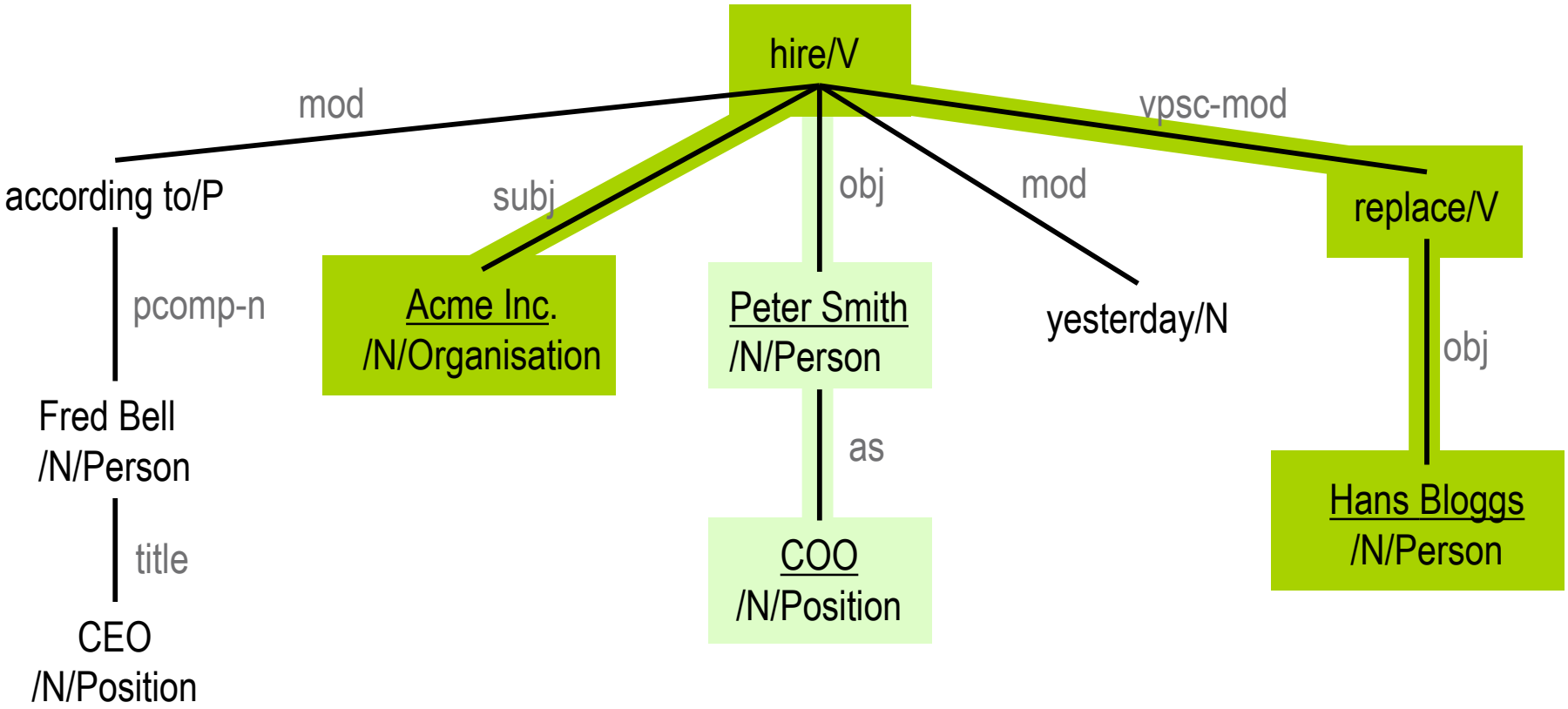
- verb centered
- All chains dominated by a verb, which contain at least one relevant named entity and their combinations



Previous Work: Subtree-Model

Sudo et al. (2003)

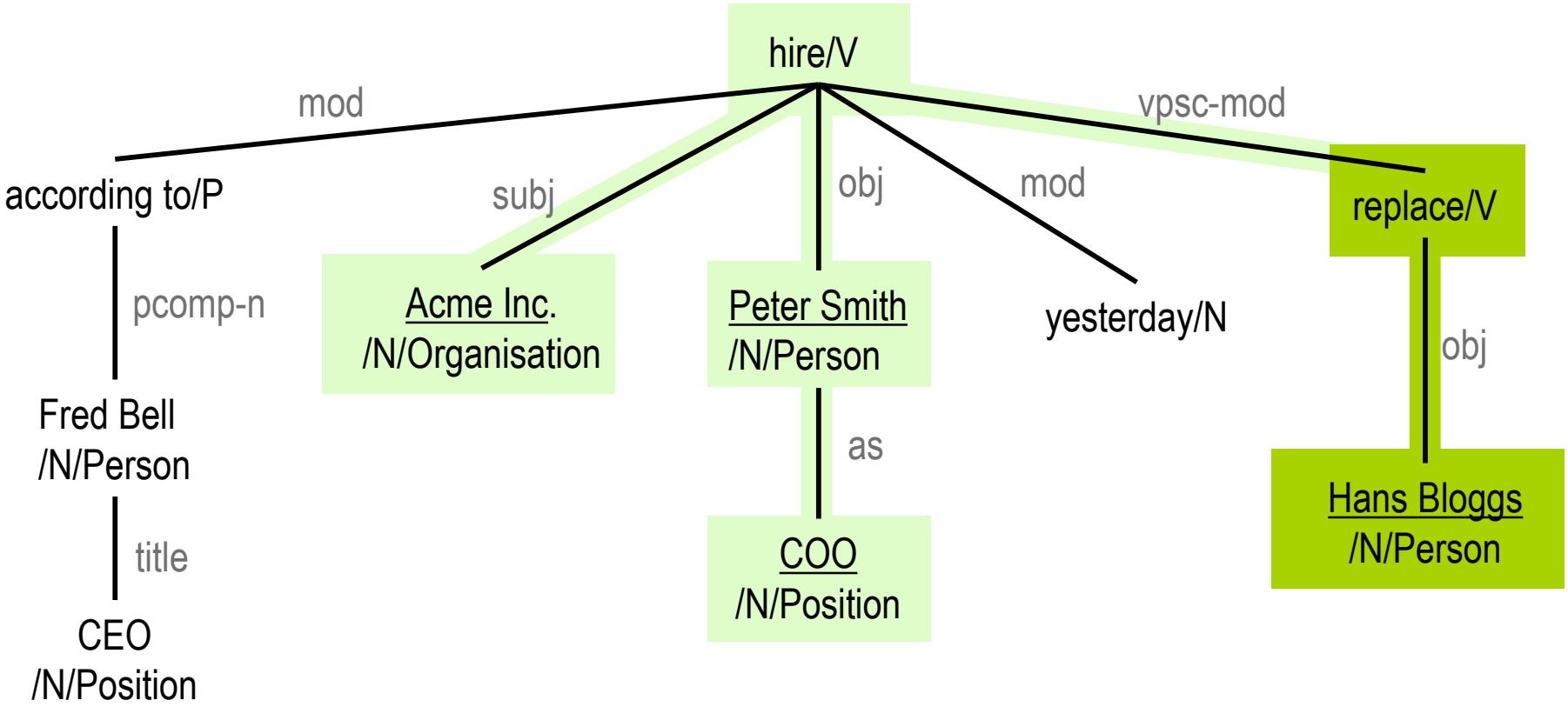
- verb centered
- All chains dominated by a verb, which contain at least one relevant named entity and their combinations



Previous Work: Subtree-Model

Sudo et al. (2003)

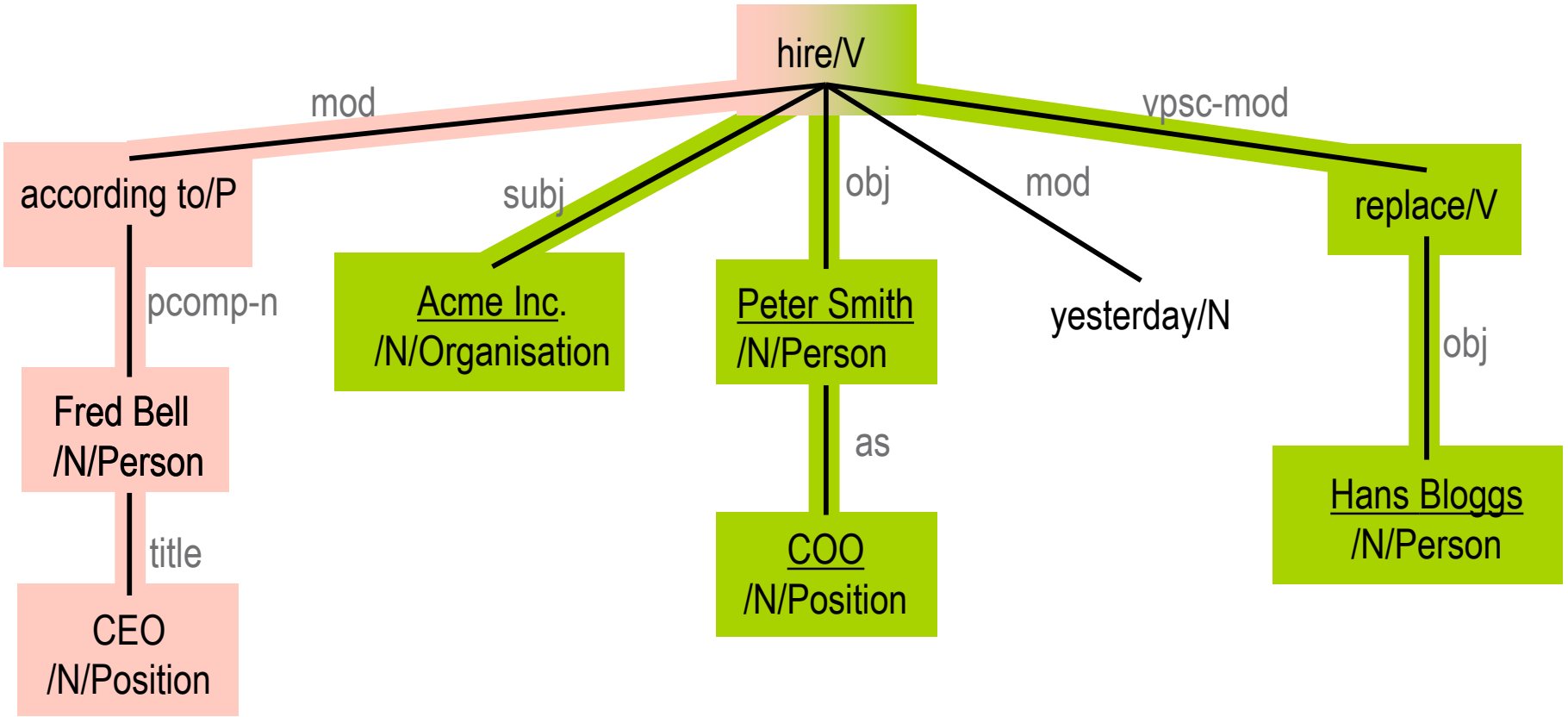
- verb centered
- All chains dominated by a verb, which contain at least one relevant named entity and their combinations



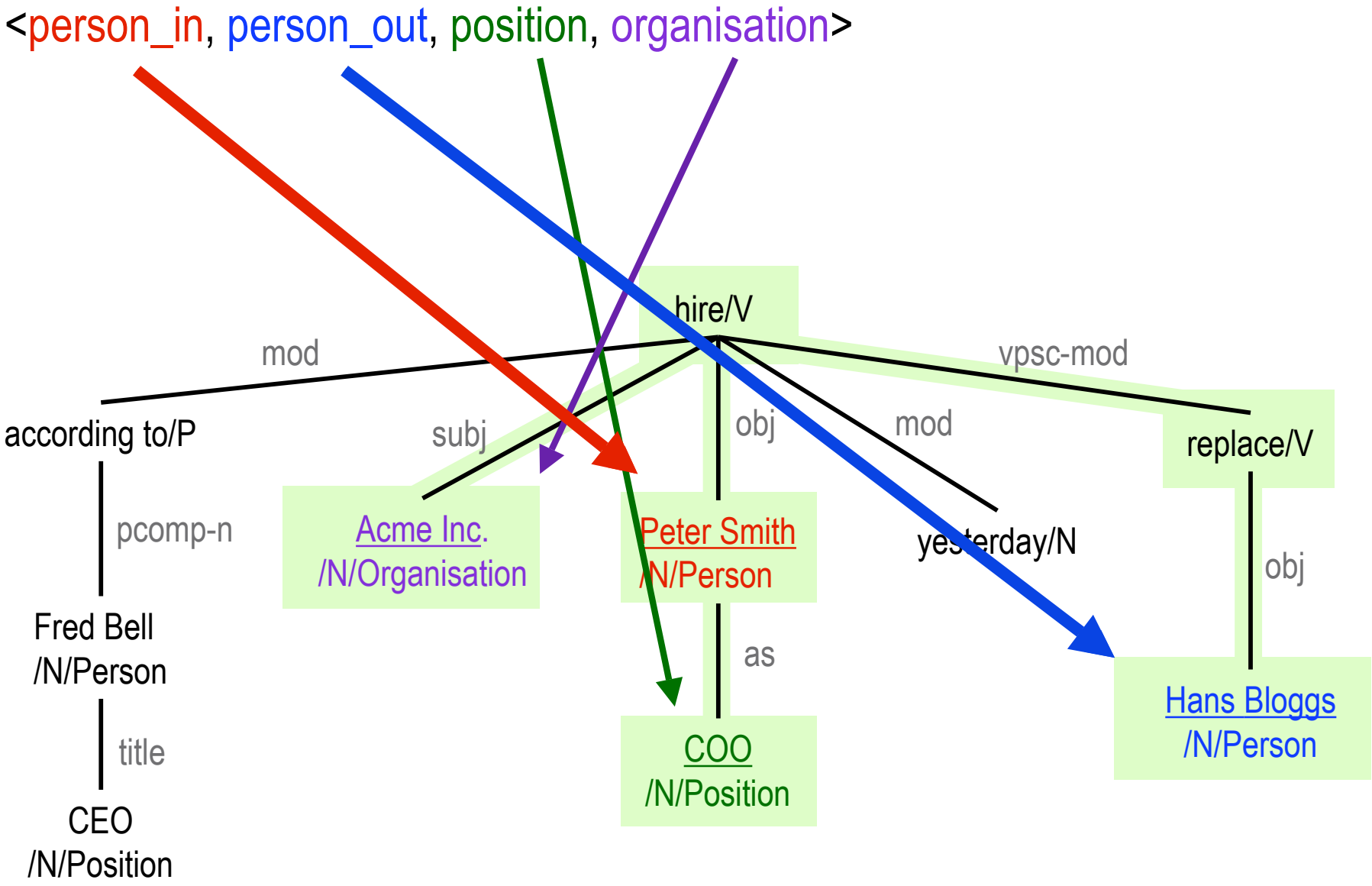
Previous Work: Subtree-Model

Sudo et al. (2003)

- verb centered
- All chains dominated by a verb, which contain at least one relevant named entity and their combinations



None of the existing models links the detected slot-filling candidates with their respective semantic roles



- State of the art
- **D**omain **A**daptive **R**elation **E**xtraction **F**ramework (**DARE**)
- Experiments and evaluations
- Performance analysis and discussion
- Conclusion and future work

Properties of DARE

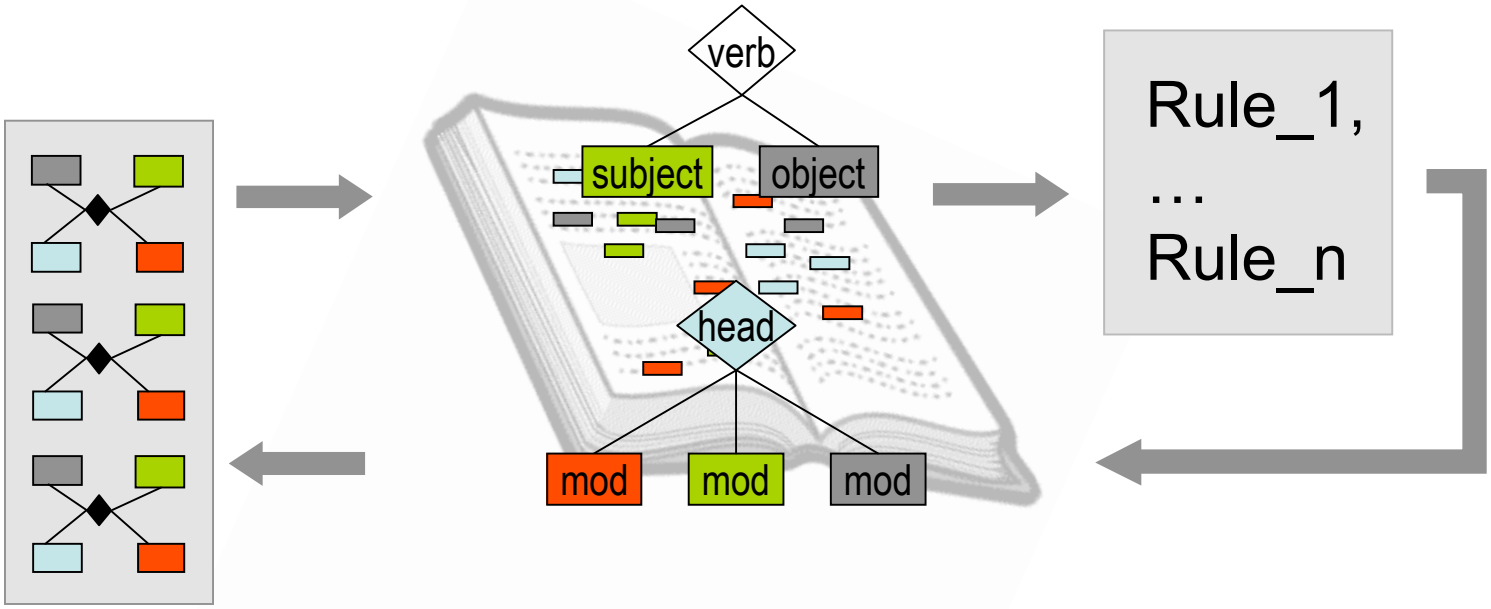
- Samples of target relation instances serve as semantic seed
- Systematic treatment of n-ary relations and their projections
- Exploitation of relation projections for pattern discovery
- Bottom-up compositional pattern discovery
- A recursive linguistic rule representation
- Rules contain semantic roles w.r.t. to target relation
- Bottom-up compression method to generalize rules
- Filtering of rule candidates by “domain relevance”

Novel Properties of DARE

- Samples of target relation instances serve as semantic seed
- Systematic treatment of n-ary relations and their projections
- Exploitation of relation projections for pattern discovery
- Bottom-up compositional pattern discovery
- A recursive linguistic rule representation
- Rules contain semantic roles w.r.t. to target relation
- Bottom-up compression method to generalize rules
- Filtering of rule candidates by “domain relevance”

Bootstrapping Relation Extraction with Semantic Seed

Adapted from
DIPRE (Brin, 1998) and Snowball (Agichtein & Gravano, 2000)
but extended and enriched with linguistic analysis

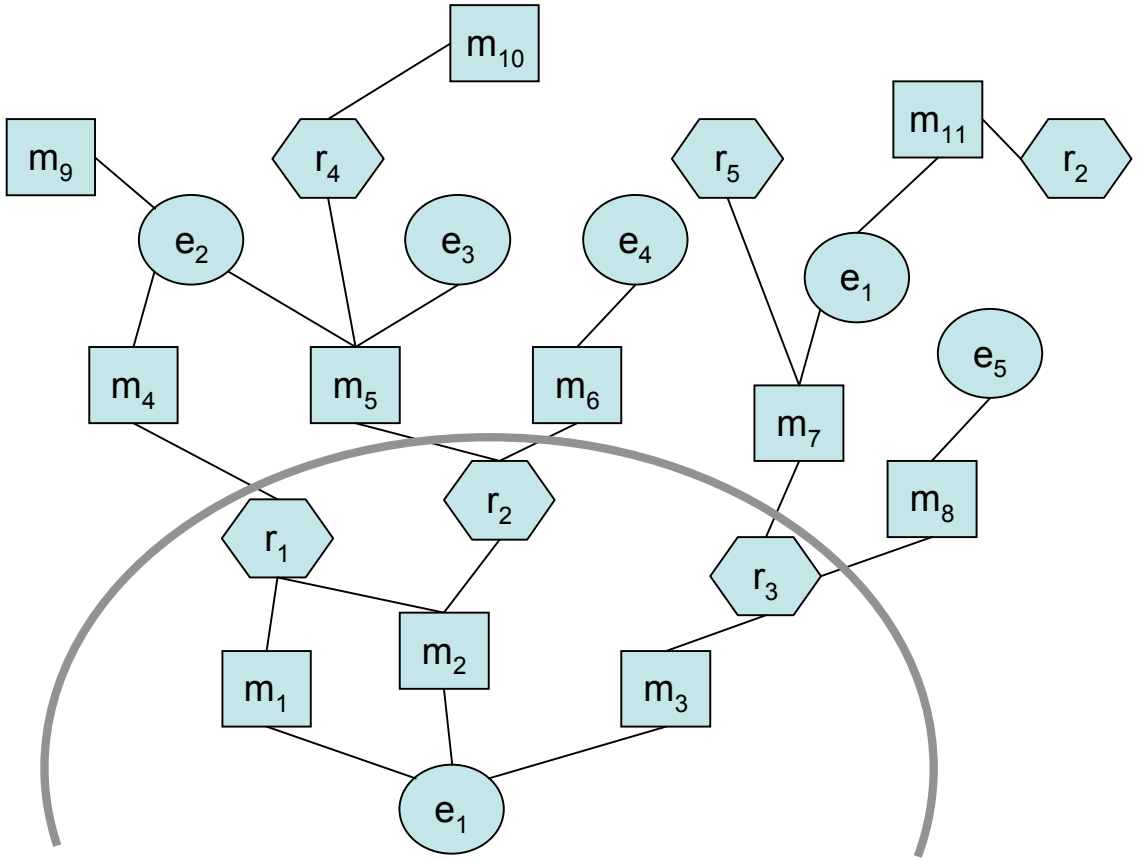


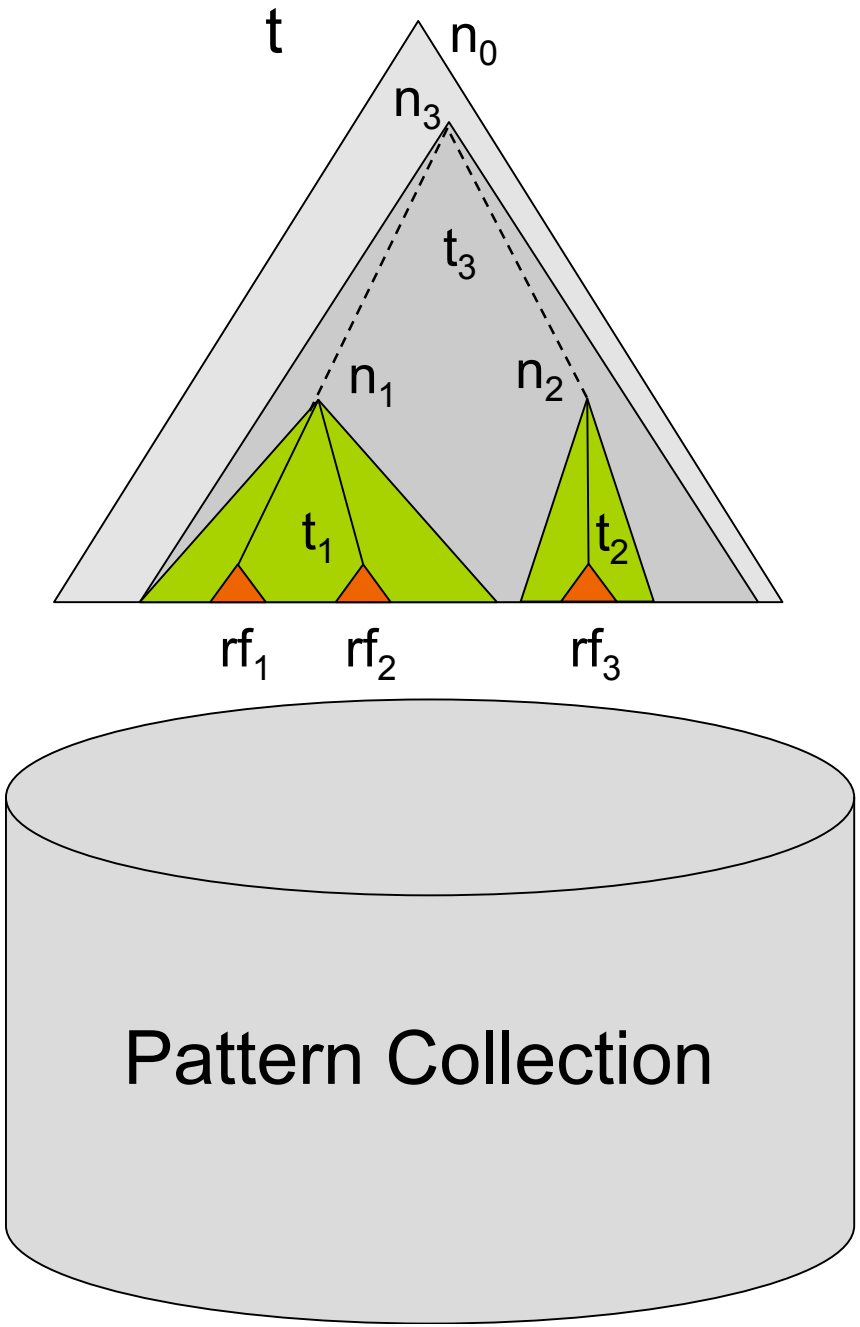
Bootstrapping Relation Extraction with Semantic Seed

- DIPRE and Snowball
  - binary relations only, no projections, no linguistic analysis
- DARE
  - n-ary relations and their projections, deep linguistic analysis

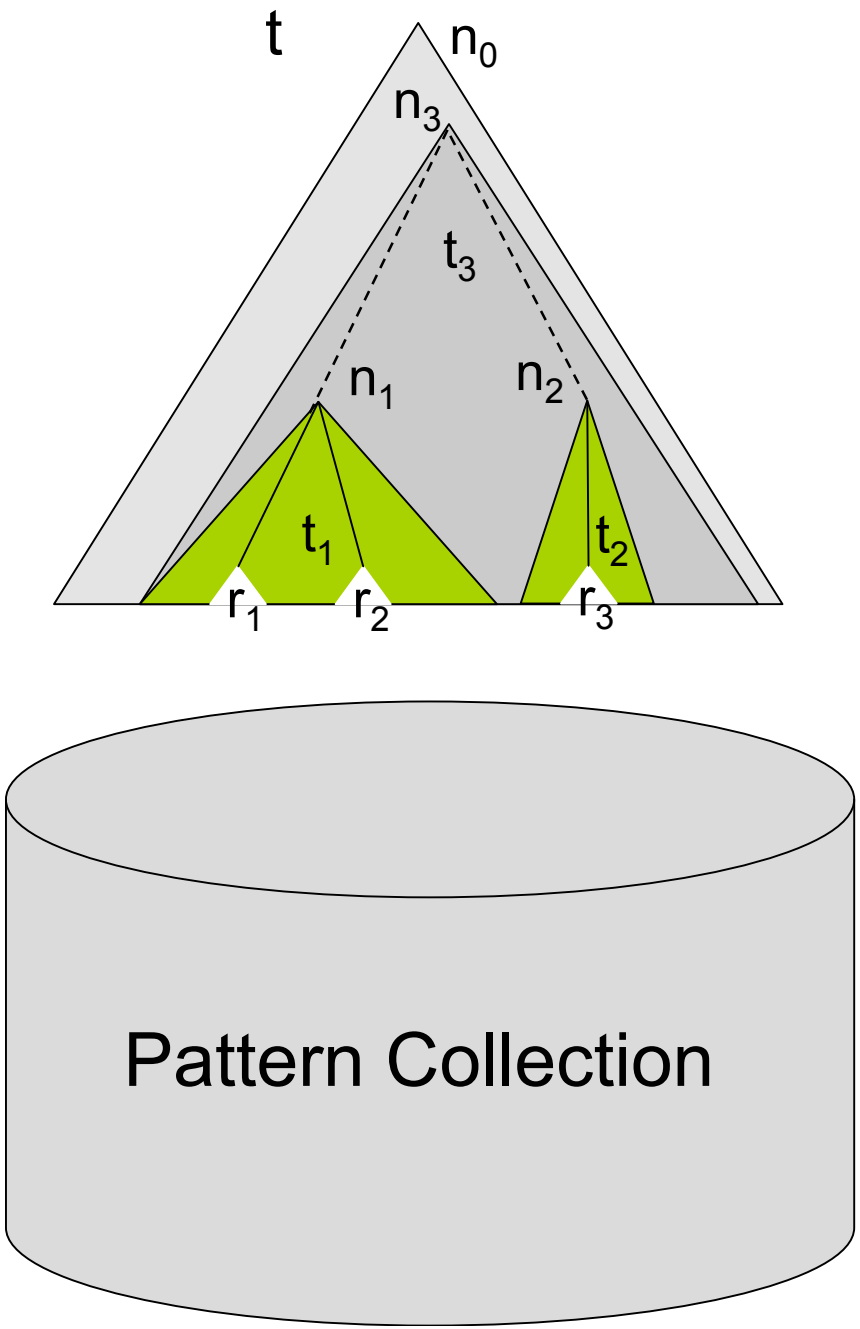
(in the experiments I use MINIPAR by Dekan Lin 1999)

Start of Bootstrapping (simplified)

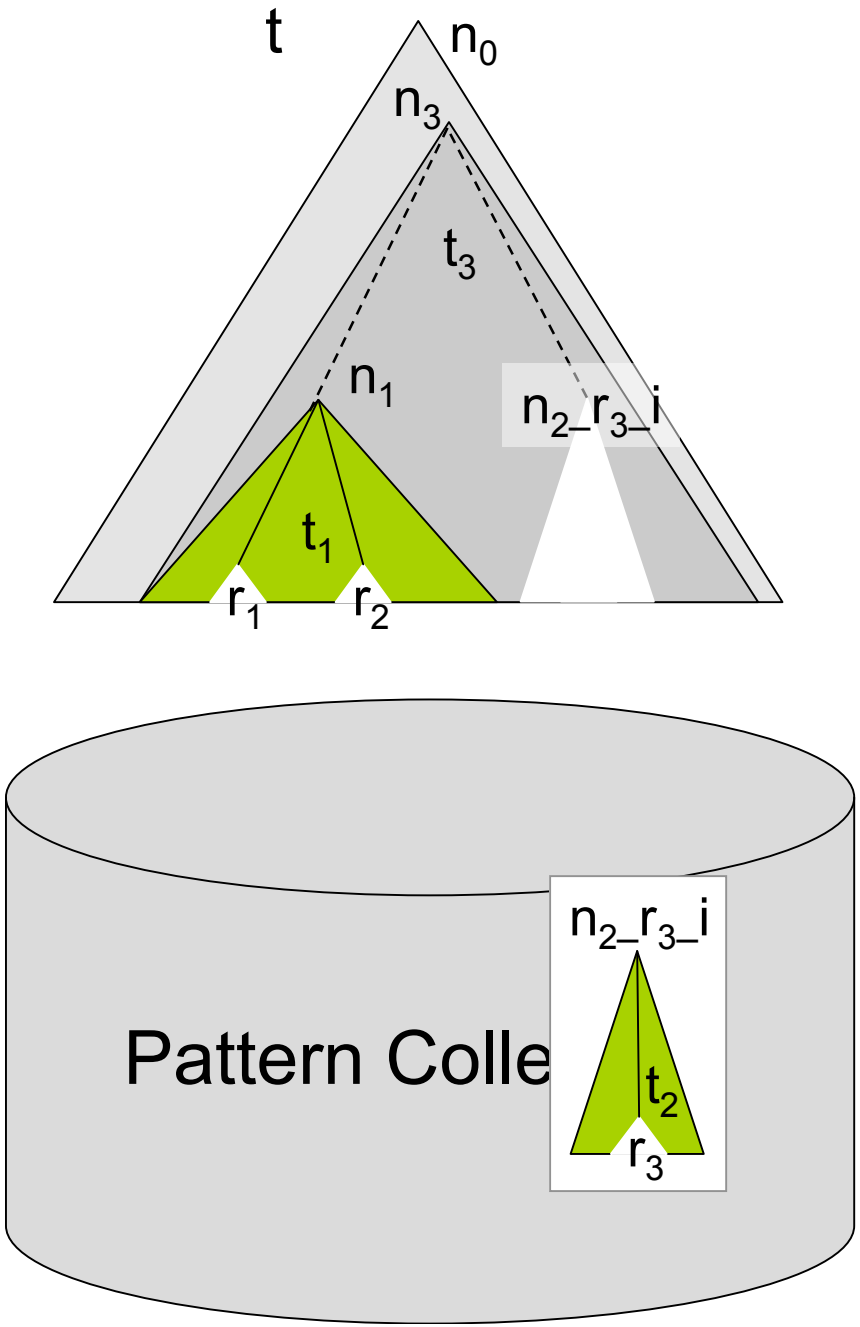




0. replace all nodes that are instantiated with the seed arguments by new nodes. Label these new nodes with the seed argument roles and their entity classes;



0. replace all nodes that are instantiated with the seed arguments by new nodes. Label these new nodes with the seed argument roles and their entity classes;



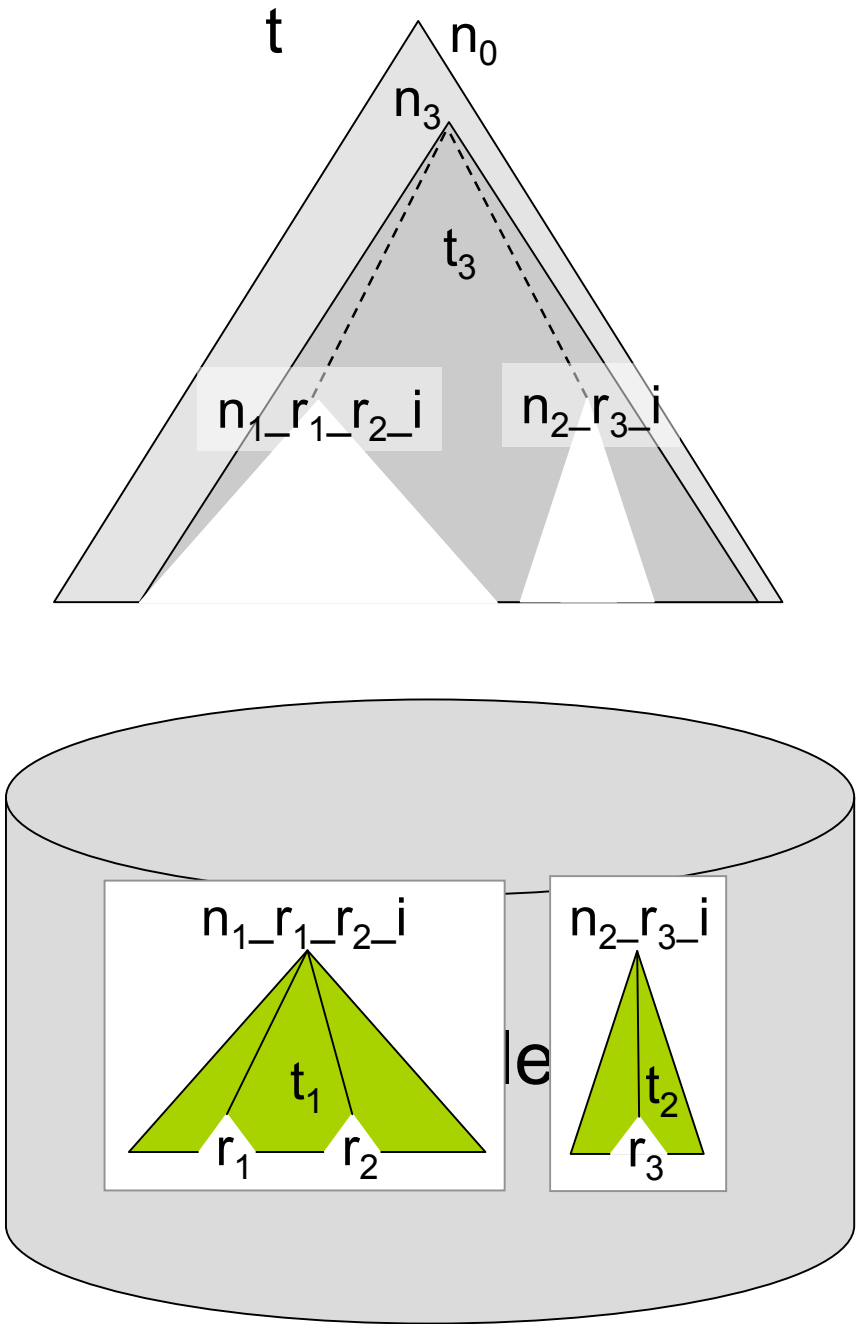
0. replace all nodes that are instantiated with the seed arguments by new nodes. Label these new nodes with the seed argument roles and their entity classes;

for i=1 to n

1. identify the set of the lowest non-terminal nodes N_1 in t that dominate i arguments (possibly among other nodes).

2. substitute N_1 by nodes labelled with the seed argument roles and their entity classes

3. prune the subtrees dominated by N_1 from t and add these subtrees into the pattern collection. These subtrees are assigned the argument role information and a unique id.



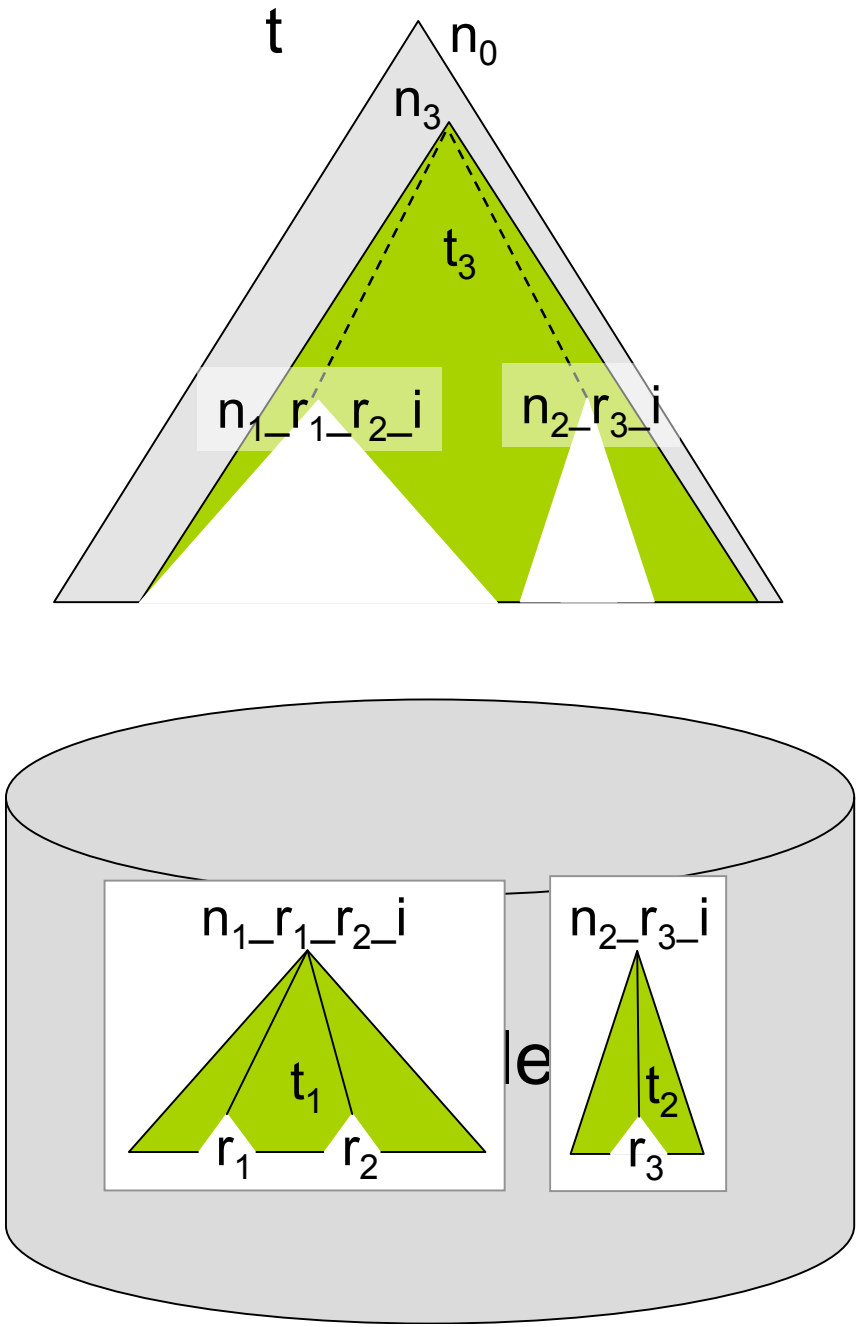
0. replace all nodes that are instantiated with the seed arguments by new nodes. Label these new nodes with the seed argument roles and their entity classes;

for i=1 to n

1. identify the set of the lowest non-terminal nodes N_1 in t that dominate i arguments (possibly among other nodes).

2. substitute N_1 by nodes labelled with the seed argument roles and their entity classes

3. prune the subtrees dominated by N_1 from t and add these subtrees into the pattern collection. These subtrees are assigned the argument role information and a unique id.



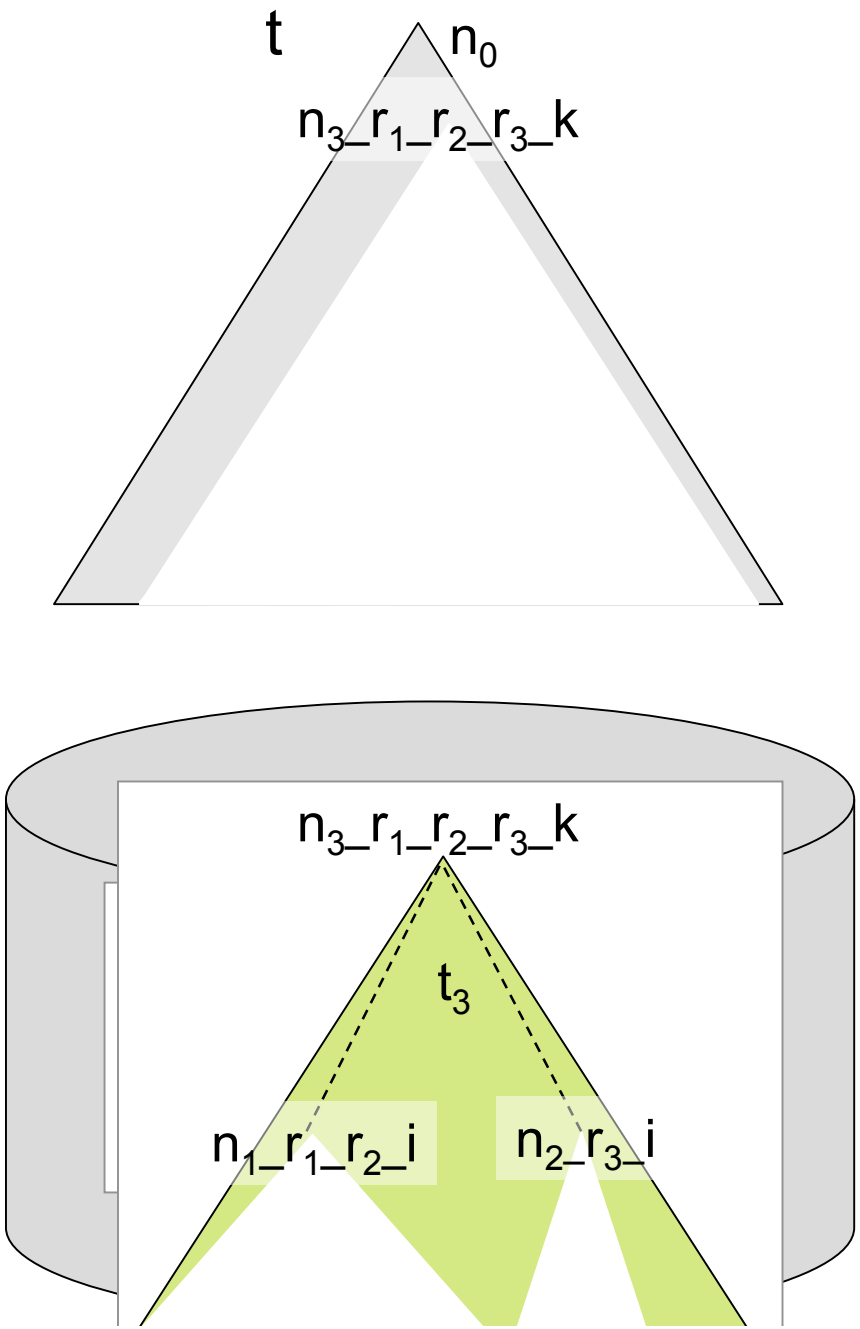
0. replace all nodes that are instantiated with the seed arguments by new nodes. Label these new nodes with the seed argument roles and their entity classes;

for i=1 to n

1. identify the set of the lowest non-terminal nodes N_1 in t that dominate i arguments (possibly among other nodes).

2. substitute N_1 by nodes labelled with the seed argument roles and their entity classes

3. prune the subtrees dominated by N_1 from t and add these subtrees into the pattern collection. These subtrees are assigned the argument role information and a unique id.



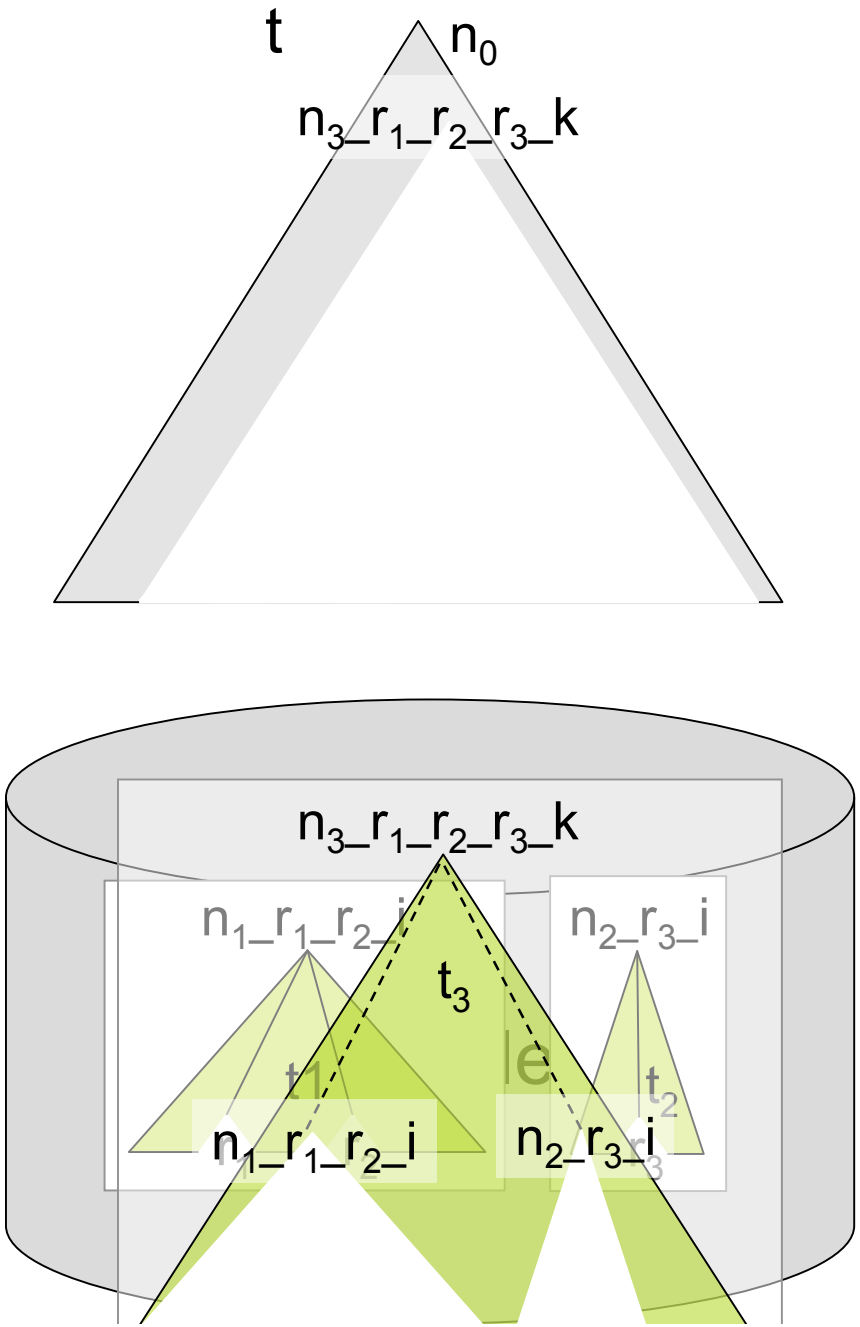
0. replace all nodes that are instantiated with the seed arguments by new nodes. Label these new nodes with the seed argument roles and their entity classes;

for i=1 to n

1. identify the set of the lowest non-terminal nodes N_1 in t that dominate i arguments (possibly among other nodes).

2. substitute N_1 by nodes labelled with the seed argument roles and their entity classes

3. prune the subtrees dominated by N_1 from t and add these subtrees into the pattern collection. These subtrees are assigned the argument role information and a unique id.



0. replace all nodes that are instantiated with the seed arguments by new nodes. Label these new nodes with the seed argument roles and their entity classes;

for i=1 to n

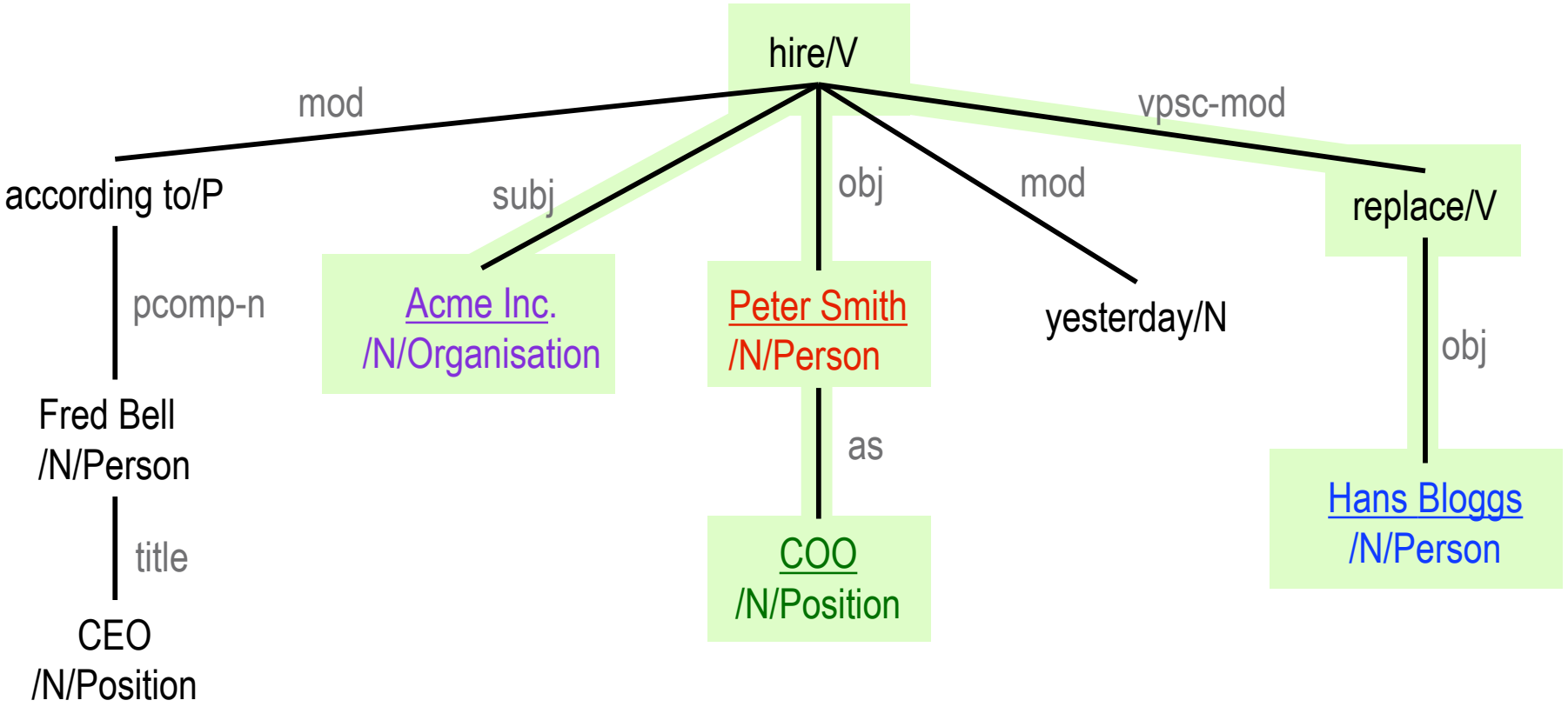
1. identify the set of the lowest non-terminal nodes N_1 in t that dominate i arguments (possibly among other nodes).

2. substitute N_1 by nodes labelled with the seed argument roles and their entity classes

3. prune the subtrees dominated by N_1 from t and add these subtrees into the pattern collection. These subtrees are assigned the argument role information and a unique id.

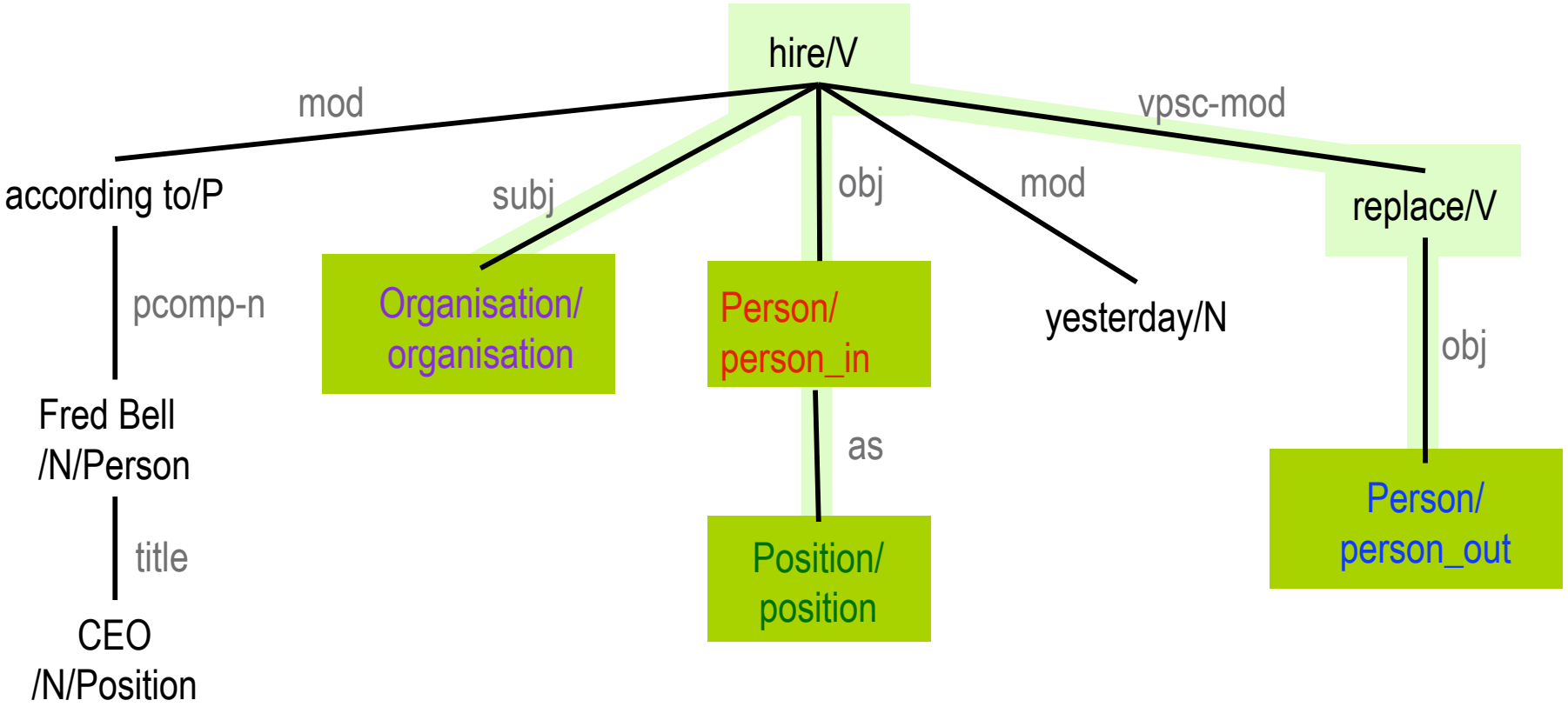
According to CEO Fred Bell, Acme Inc. hired Peter Smith as COO yesterday, replacing Hans Bloggs.

<Peter Smith/person_in, Hans Bloggs/person_out, COO /position, Acme Inc. /organisation>



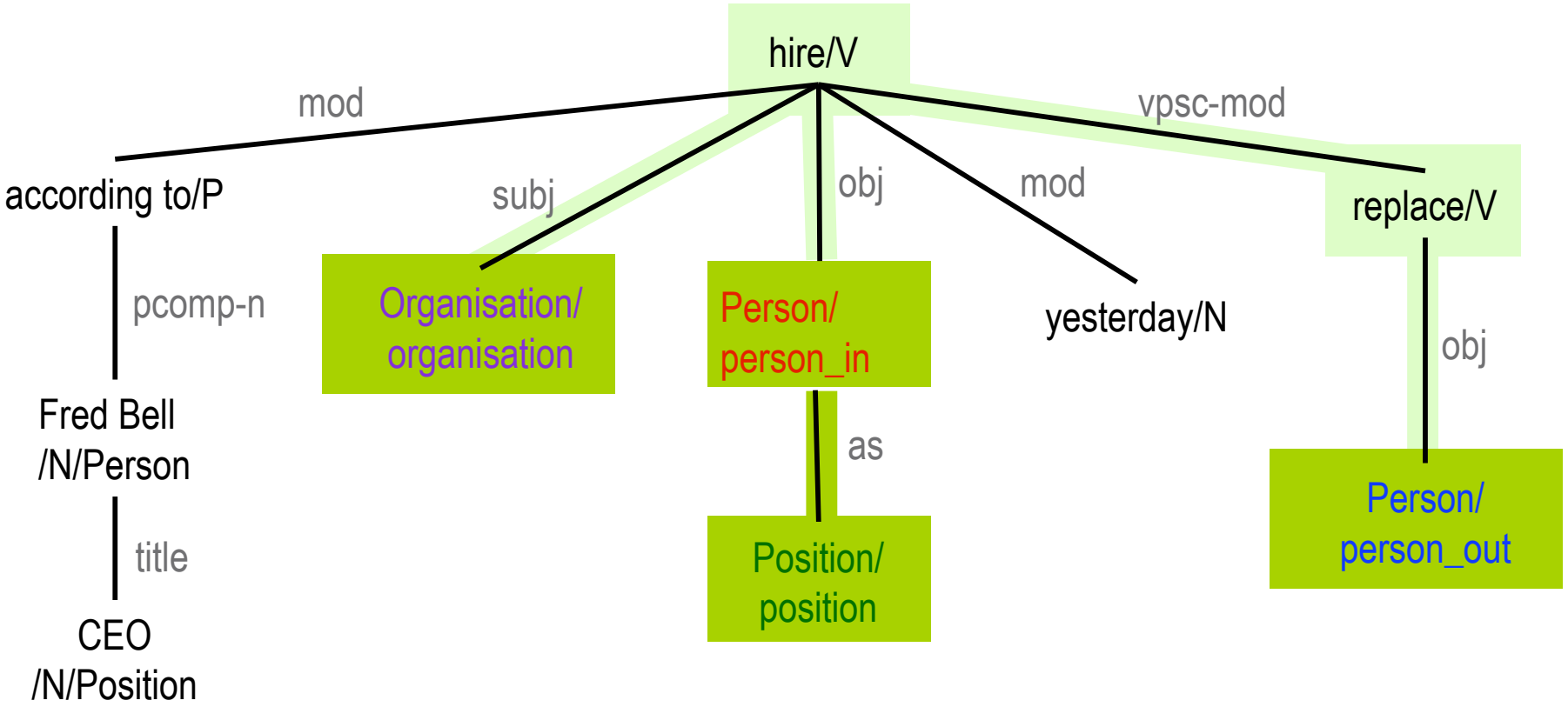
According to CEO Fred Bell, Acme Inc. hired Peter Smith as COO yesterday, replacing Hans Bloggs.

<Peter Smith/person_in, Hans Bloggs/person_out, COO /position, Acme Inc. /organisation>



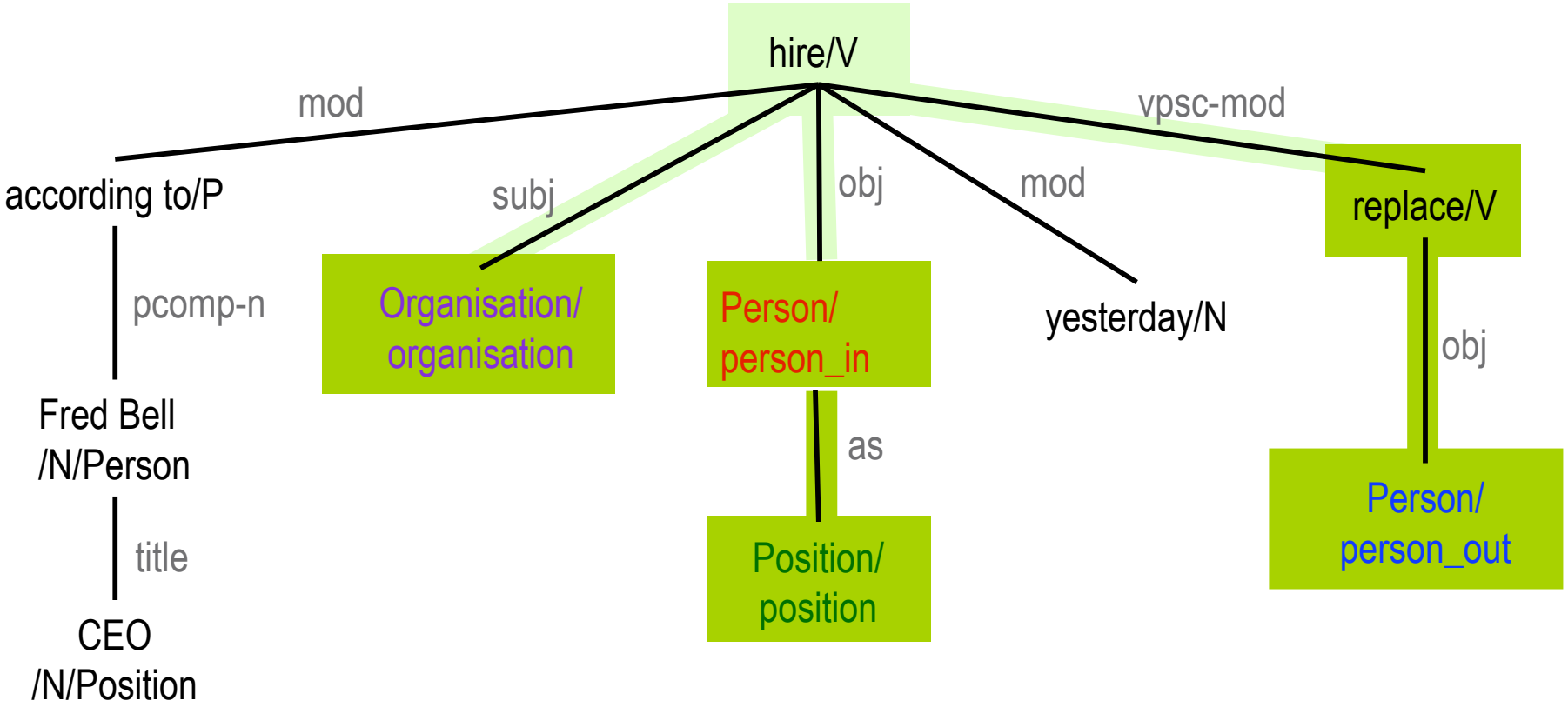
According to CEO Fred Bell, Acme Inc. hired Peter Smith as COO yesterday, replacing Hans Bloggs.

<Peter Smith/person_in, Hans Bloggs/person_out, COO /position, Acme Inc. /organisation>



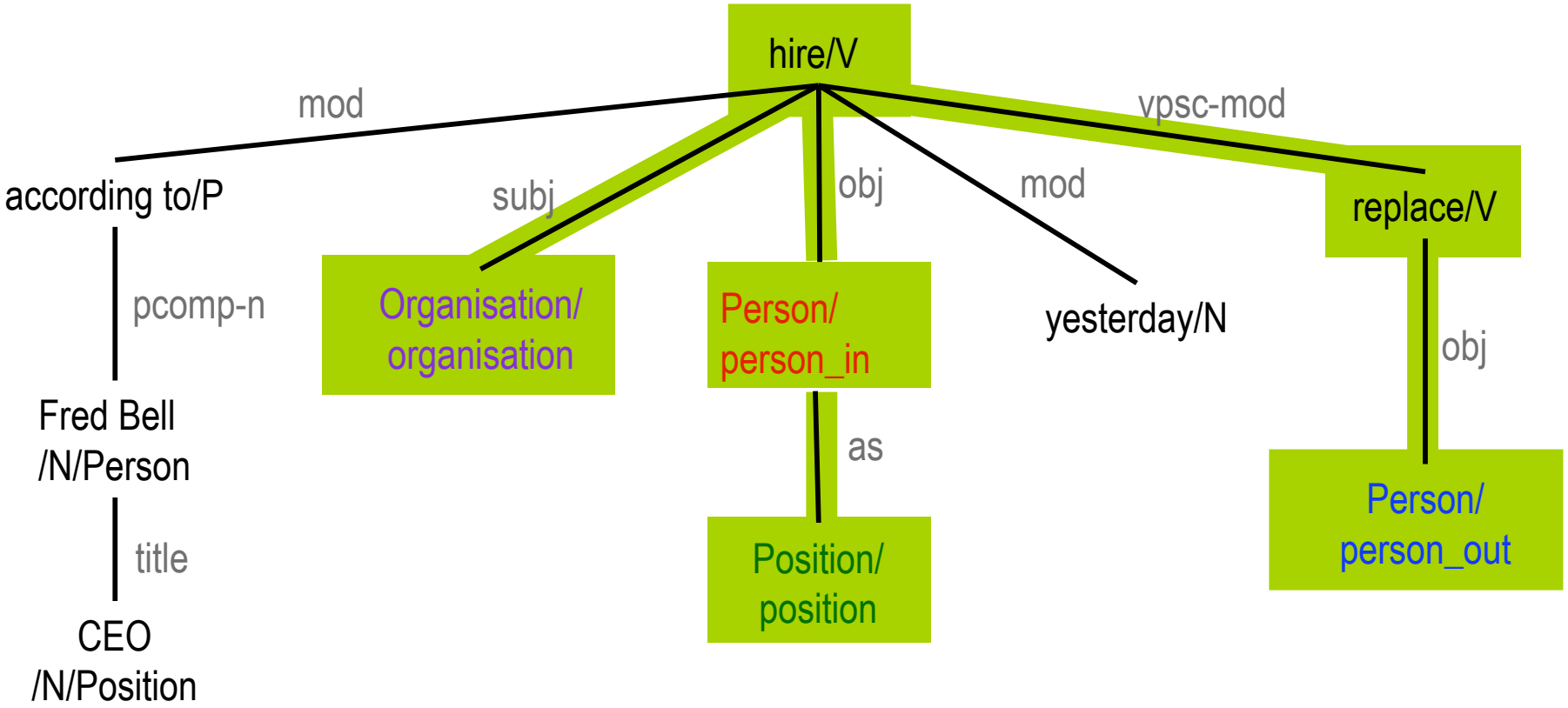
According to CEO Fred Bell, Acme Inc. hired Peter Smith as COO yesterday, replacing Hans Bloggs.

<Peter Smith/person_in, Hans Bloggs/person_out, COO /position, Acme Inc. /organisation>



According to CEO Fred Bell, Acme Inc. hired Peter Smith as COO yesterday, replacing Hans Bloggs.

<Peter Smith/person_in, Hans Bloggs/person_out, COO /position, Acme Inc. /organisation>



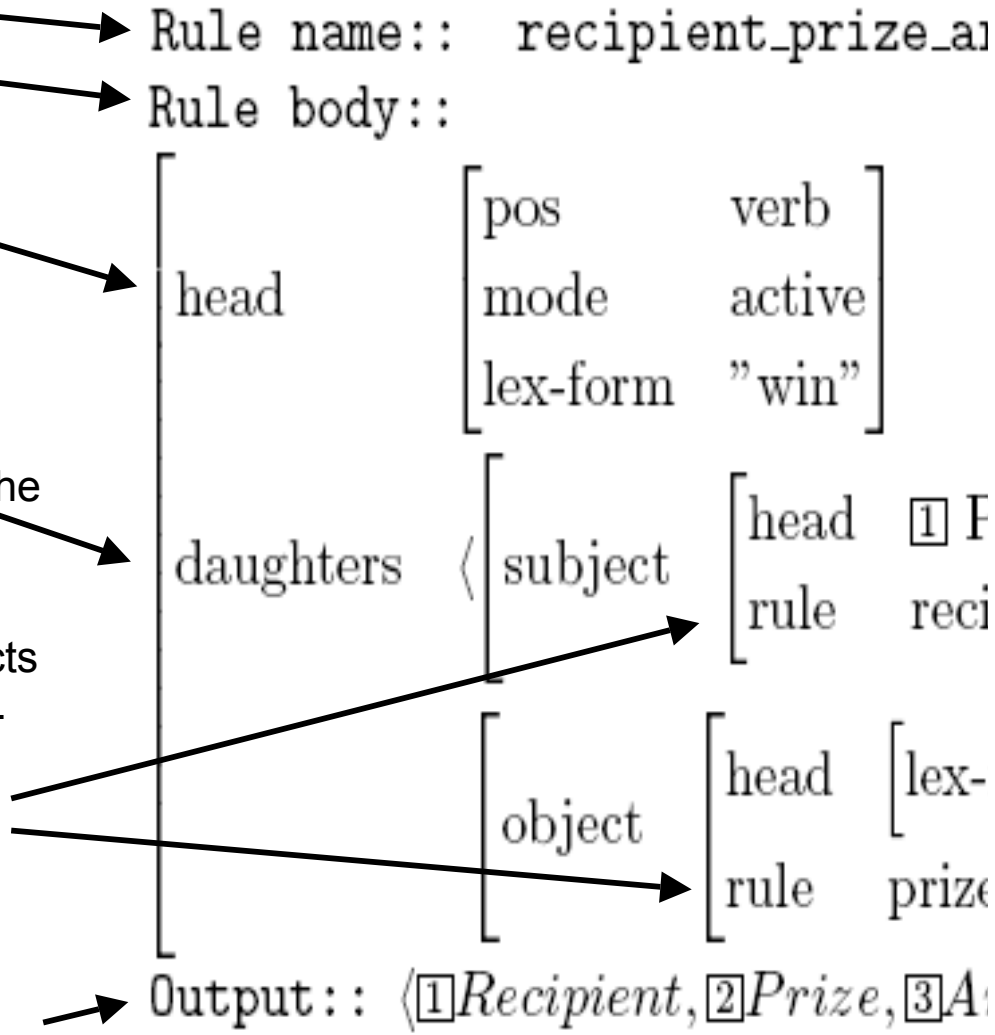
DARE Rule Components

1. rule name: r_i;

2. rule body: in AVM format containing:

- **head**: the linguistic annotation of the top node of the linguistic structure;
- **daughters**: its value is a list of specific linguistic structures (e.g., subject, object, head, mod), derived from the linguistic analysis, e.g., dependency structures and the named entity information;
- **rules**: its value is a DARE rule which extracts a subset of arguments of the target relation.

3. Output: n-tupel of arguments with their roles



DARE Rule Components

Rule name:: recipient_prize_area_year_1

Rule body::

$$\left[\begin{array}{ll} \text{head} & \left[\begin{array}{ll} \text{pos} & \text{verb} \\ \text{mode} & \text{active} \\ \text{lex-form} & \text{"win"} \end{array}\right] \\ \text{daughters} & \left\langle \left[\begin{array}{ll} \text{subject} & \left[\begin{array}{ll} \text{head} & \boxed{1}\ \text{Person} \\ \text{rule} & \text{recipient\_1::}\ \langle \boxed{1} Person \rangle \end{array}\right] \end{array}\right], \right. \\ & \left. \left[\begin{array}{ll} \text{object} & \left[\begin{array}{ll} \text{head} & \left[\text{lex-form} \quad \text{"prize"}\right] \\ \text{rule} & \text{prize\_area\_year\_1::}\ \langle \boxed{2} Prize, \boxed{3} Area, \boxed{4} Year \rangle \end{array}\right] \end{array}\right] \right\rangle \end{array}\right]$$

Output:: $\langle \boxed{1} Recipient, \boxed{2} Prize, \boxed{3} Area, \boxed{4} Year \rangle$

prize_area_year_1

Rule name:: prize_area_year_1

Rule body::

$$\begin{bmatrix} \text{head} & \begin{bmatrix} \text{pos} & \text{noun} \\ \text{lex-form} & \text{"prize"} \end{bmatrix} \\ \text{daughters} & \left\langle \begin{bmatrix} \text{lex-mod} & \begin{bmatrix} \text{head} & \boxed{3}\ \text{Year} \end{bmatrix} \end{bmatrix}, \right. \\ & \begin{bmatrix} \text{lex-mod} & \begin{bmatrix} \text{head} & \boxed{1}\ \text{Prize} \end{bmatrix} \end{bmatrix}, \\ & \left. \begin{bmatrix} \text{lex-mod} & \begin{bmatrix} \text{head} & \boxed{2}\ \text{Area} \end{bmatrix} \end{bmatrix} \right\rangle \end{bmatrix}$$

Output:: $\langle \boxed{1}Prize, \boxed{2}Area, \boxed{3}Year \rangle$

- State of the art
- **D**omain **A**daptive **R**elation **E**xtraction **F**ramework (**DARE**)
- Experiments and evaluations
- Performance analysis and discussion
- Conclusion and future work

Two Domains

- Award Events (start with subdomain Nobel Prizes)

  reasons: good news coverage
  complete list of all award events
  good starting point for other award domains

- Management Succession Events

  reason: comparison with previous work

Experiments

- Two domains
  - Nobel Prize Awards: <recipient, prize, area, year>
  - Management Succession: <person_in, person_out, position, organisation>
- Test data sets

Data Set Name	Doc Number	Data Amount
Nobel Prize A (1981-1998)	1032	5.8 MB
Nobel Prize B (1999-2005)	2296	12.6 MB
Nobel Prize A+B	3328	18.4 MB
MUC-6	199	1MB

Evaluation Against Ideal Tables

Data Set	Seed	Precision	Recall
Nobel Prize A	<[Sen, Amartya], nobel, economics, 1998>	**87.3%**	**31.0%**
Nobel Prize A	<[Arias, Oscar], nobel, peace, 1987>	**83.8%**	**32.0%**
Nobel Prize B	<[Zewail, Ahmed H], nobel, chemistry,1999>	**71.6%**	**50.7%**
A+B	<[Zewail, Ahmed H], nobel, chemistry,1999>	**80.6%**	**62.9%**

Management Succession Domain

Initial Seed #	Precision	Recall
1	**12.6%**	**7.0%**
1	**15.1%**	**21.8%**
20	**48.4%**	**34.2%**
55	**62.0%**	**48.0%**

Comparison

Our result with 20 seeds (after 4 iterations)

- precision: 48.4%
- recall: 34.2%

compares well with the best result reported so far by (Greenwood and Stevenson, 2006) with the linked chain model starting with 7 hand-crafted patterns (after 190 iterations)

- precision: 43.4%
- recall: 26.5%

Reusability of Rules

- Prize award patterns
  - Detection of other Prizes such as *Pulitzer Prize*, *Turner Prize*
  - Precision: 86.2%
- Management succession
  - Domain independent binary pattern rules:
    Person-Organisation, *Person-Position*
  - Evaluation of top 100 relation instances
    Precision: 98%

- State of the art
- **D**omain **A**daptive **R**elation **E**xtraction **F**ramework (**DARE**)
- Experiments and evaluations
- Performance analysis and discussion
- Conclusion and future work

The Dream

- Wouldn‘t it be wonderful if we could always automatically learn most or all relevant patterns of some relation from one single semantic instance!
- Or at least find all event instances.
- This sounds too good to be true!

Research Questions

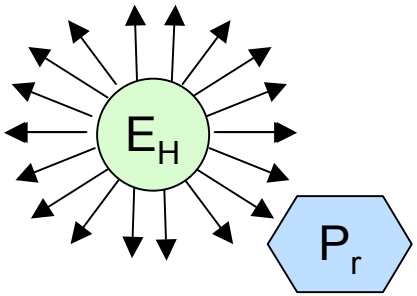
As scientists we want to know

- Why does it work for some tasks?
- Why doesn‘t it work for all tasks?
- How can we estimate the suitability of domains?
- How can we deal with less suitable domains?

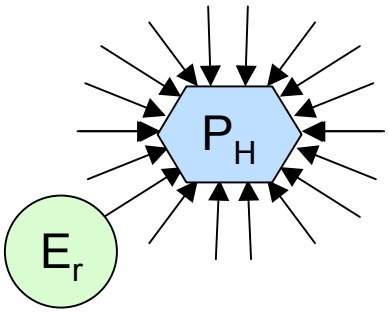
Careful analysis confirmed the following assumption:

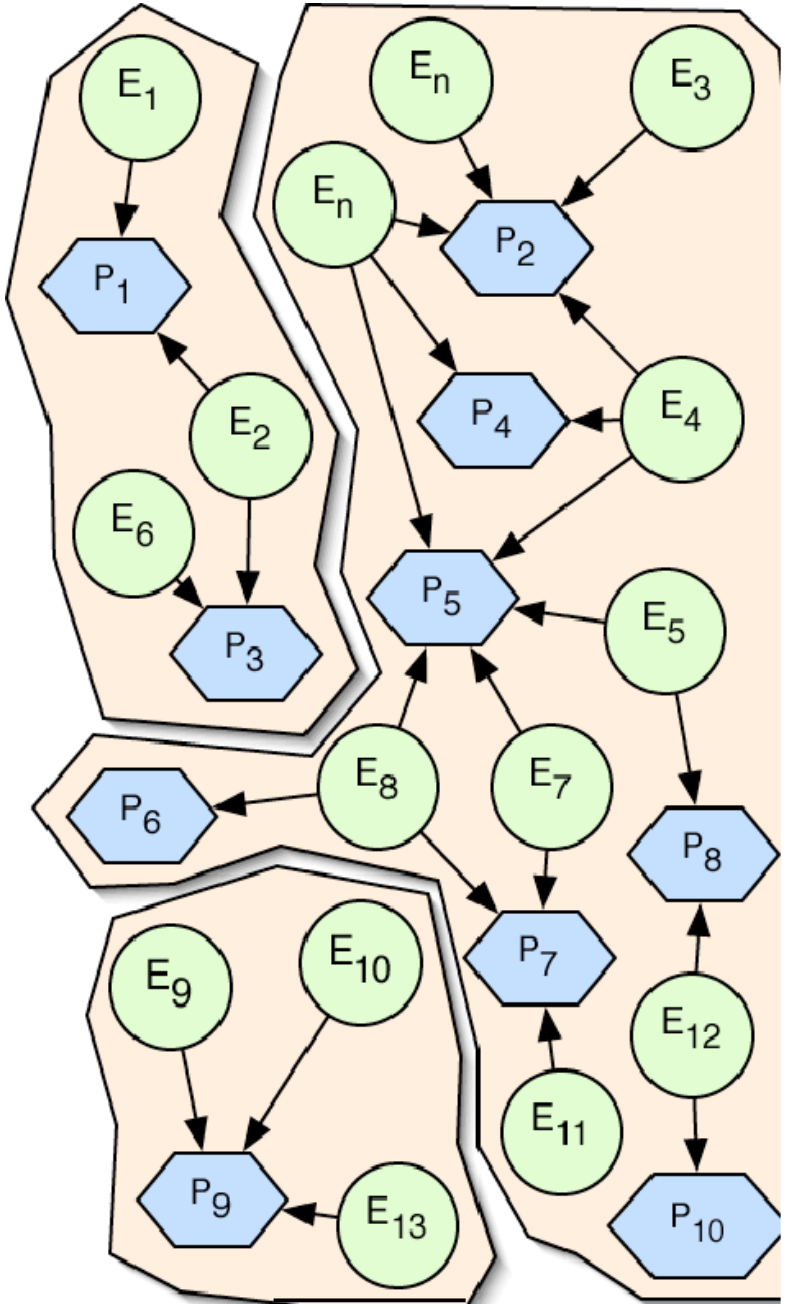
redundancy, both on patterns and event mentions, helps.

Frequently reported events make rare patterns reachable

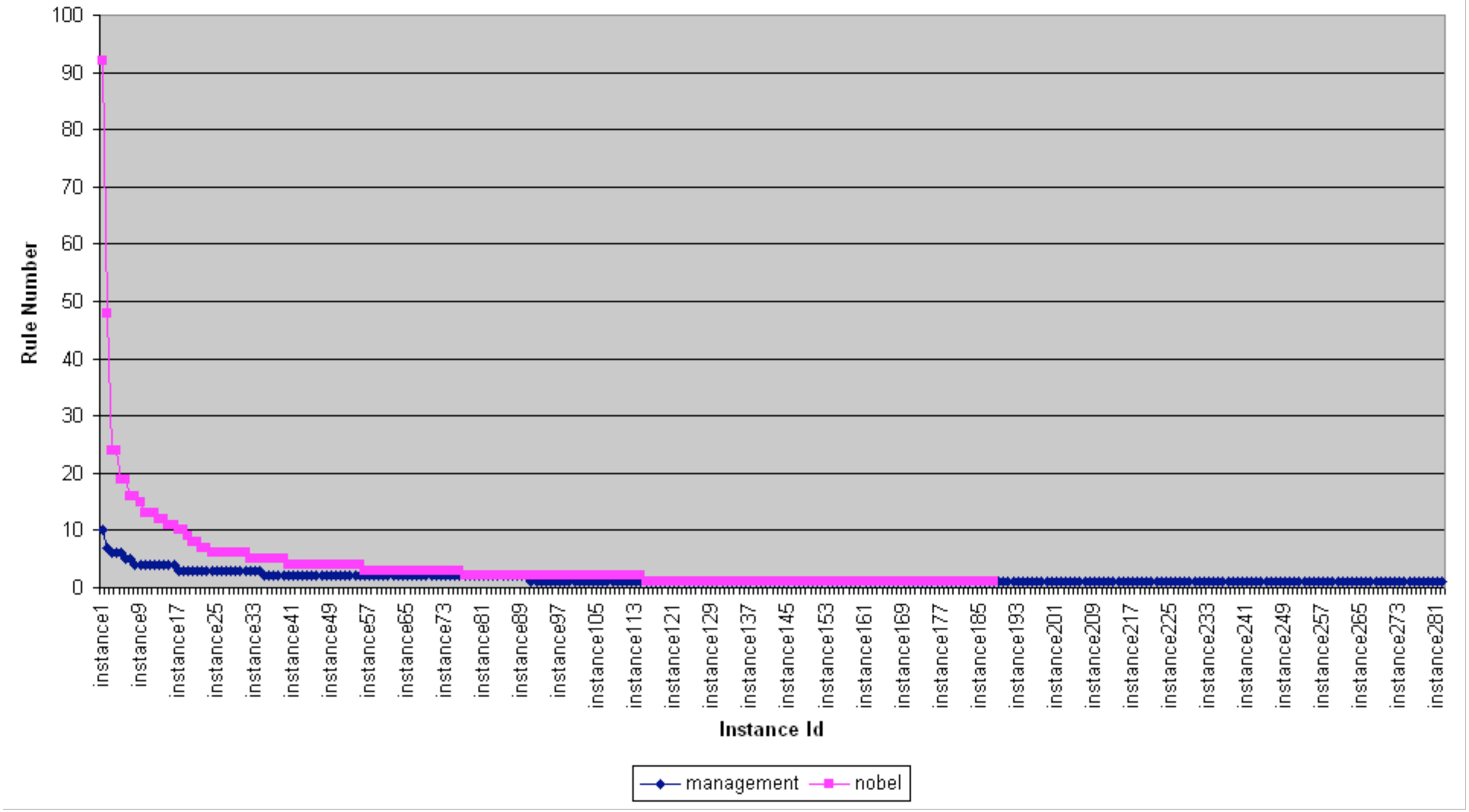


Popular patterns help to reach rarely mentioned events

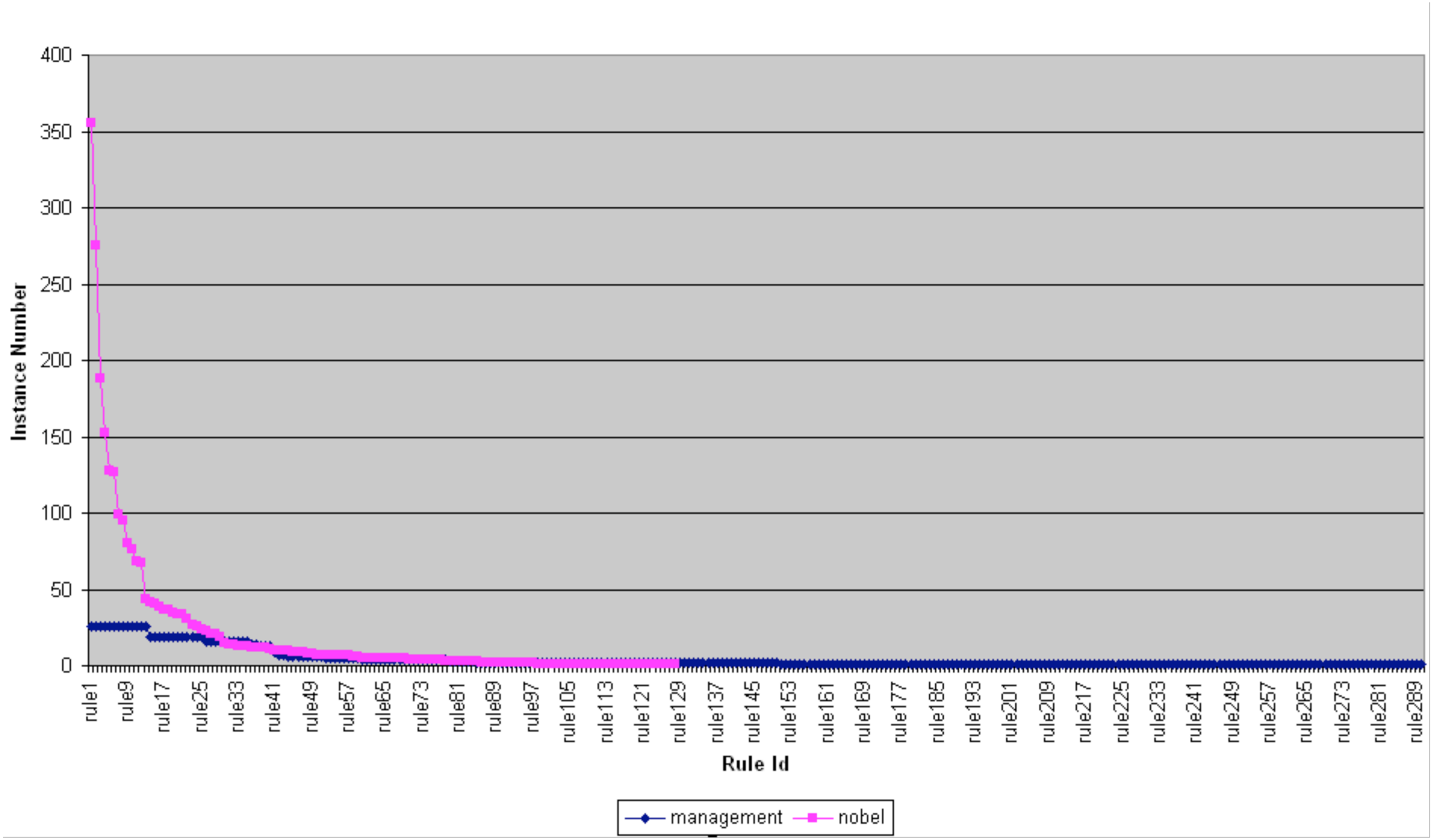




Instance to Pattern
Nobel Prize vs. Management Succession



Rule to Instances
(Nobel Prize vs. Management Succession)



Insights

- Results from graph theory help to understand the requirements on data.

  Example: small world property

- For data sets with continents and islands, we can sometimes exploit additional data or auxiliary domains to bridge the islands by learning rare patterns.

  Example: use of Nobel prize domain for learning patterns for events concerning less popular prizes (many other prizes could be detected)

- State of the art
- **D**omain **A**daptive **R**elation **E**xtraction **F**ramework (**DARE**)
- Experiments and evaluations
- Performance analysis and discussion
- Conclusion and future work

Conclusion

- DARE is the first approach to combine the idea of bootstrapping IE systems with a linguistic grammar

- This can be illustrated by a simple formula:

  reusable generic linguistic knowledge
  \+ raw data
  \+ a few examples (seed)
  = domain specific relation extraction grammar

- In addition to the obvious practical advantages, the approach offers theoretical benefits: It supports a view of IE as a systematic gradual approximation of language understanding.

Future Work

- Improvement of recall
  - Extension of learning data
    - Bridging the islands by new additional data
    - Use of a related domain, e.g, Nobel Prize for other prizes
  - Improvement of rule generalization
  - Intersentential extraction
- Improvement of precision
  - Negative rules (domain indepedent and domain specific)
  - Integration of high-precision NLP analysis (HPSG)

References

1. N. Kushmerick. Wrapper induction: Efficiency and Expressiveness, Artificial Intelligence, 2000.
2. I. Muslea. Extraction Patterns for Information Extraction. AAAI-99 Workshop on Machine Learning for Information Extraction.
3. Riloff, E. and R. Jones. Learning Dictionaries for Information Extraction by Multi-Level Bootstrapping. In Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI-99) , 1999, pp. 474-479.
4. R. Yangarber, R. Grishman, P. Tapanainen and S. Huttunen. Automatic Acquisition of Domain Knowledge for Information Extraction. In Proceedings of the 18th International Conference on Computational Linguistics: COLING-2000, Saarbrücken.
5. F. Xu, H. Uszkoreit and Hong Li. Automatic Event and Relation Detection with Seeds of Varying Complexity. In Proceedings of AAAI 2006 Workshop Event Extraction and Synthesis, Boston, July, 2006.
6. F. Xu, D Kurz, J Piskorski, S Schmeier. A Domain Adaptive Approach to Automatic Acquisition of Domain Relevant Terms and their Relations with Bootstrapping. In Proceedings of LREC 2002.
7. W. Drozdzyski, H.U. Krieger, J. Piskorski, U. Schäfer and F. Xu. Shallow Processing with Unification and Typed Feature Structures -- Foundations and Applications. In KI (Artifical Intelligence) journal 2004.
8. Feiyu Xu, Hans Uszkoreit, Hong Li. A Seed-driven Bottom-up Machine Learning Framework for Extracting Relations of Various Complexity. In Proceeedings of ACL 2007, Prague
9. Feiyu Xu. (2007). Bootstrapping Relation Extraction from Semantic Seeds. PhD. thesis, Saarland University.

http://www.dfki.de/~neumann/ie-esslli04.html
http://en.wikipedia.org/wiki/Information_extraction
http://de.wikipedia.org/wiki/Informationsextraktion