

# $(LP)^2$ , an Adaptive Algorithm for IE from Web-related Texts

Using Shallow NLP in Adaptive IE from Web-related Texts

Fabio Ciravegna

Department of Computer Science

University of Sheffield

*(F.Ciravegna@dcs.shef.ac.uk)*



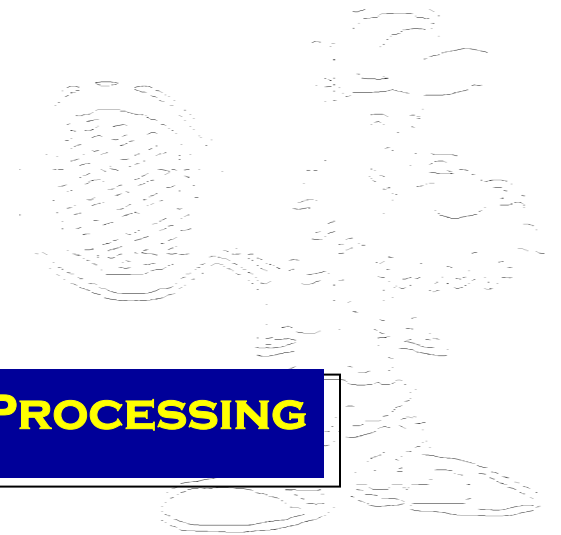


# (LP)<sup>2</sup>

Covering algorithm for adaptive IE

- Wrapper Induction-like algorithm
  - effective on structured texts
  - effective on free texts
  - Uses shallow NLP for reducing data sparseness
- (LP)<sup>2</sup> is the basis for Amilcare!

**(LP)<sup>2</sup>=LEARNING PATTERNS VIA LANGUAGE PROCESSING**





# Talk Overview

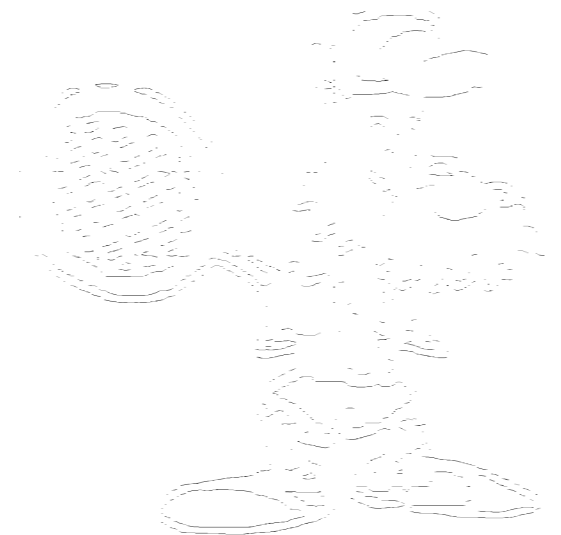
- Introduction
- Rule Induction
  - Tagging and Correction Rules
- Data Sparseness & Overfitting
  - NLP-based Rule Generalisation
  - Rule Set Pruning
- Discussion
  - Role of NLP-based Generalisation
  - Experimental Results
- Conclusion & Future Work





# Wrapper-like approaches

- Wrapper-like approaches
  - Effective on (semi-)structured texts
  - Less effective on free texts
    - Rules match words
    - Scarce or no NLP knowledge
    - Data sparseness



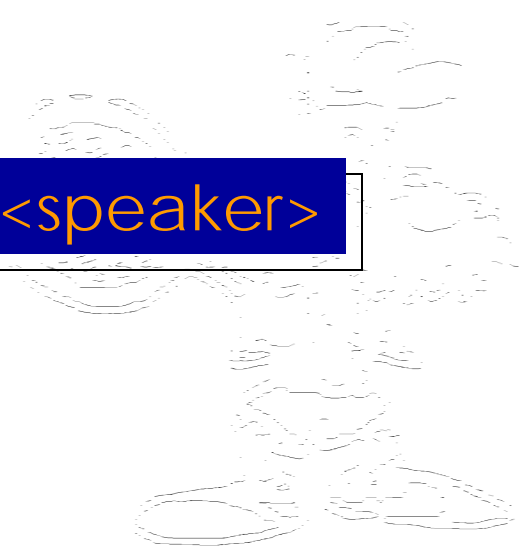


# A simple wrapper (LP)

CONDITION: a pattern of conditions on a connected sequence of words

ACTION: operate on single SGML tags (inserting or shifting tags)

`</speaker>` inserted independently from `<speaker>`

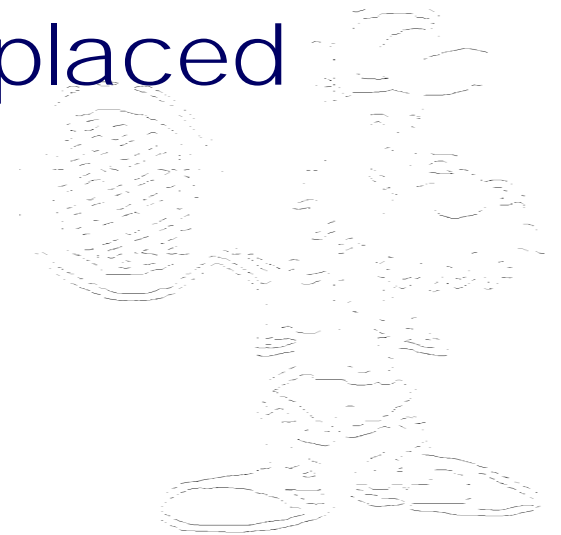




# Use of Rules for IE

## IE on **test corpus**

- Tagging Rules insert tags
- Contextual Rules add further tags:
  - inserted tags used as context
  - until no tags are added
- Correction Rules shift misplaced tags
- Tag Validation
  - removes uncoupled tags





# Tagging Rule: example

the seminar at **<time>** 4 pm will

Condition on Words	Action: Insert Tag
the	
seminar	
at	<b>&lt;time&gt;</b>
4	
pm	
will	

Initial rule= window of conditions on words

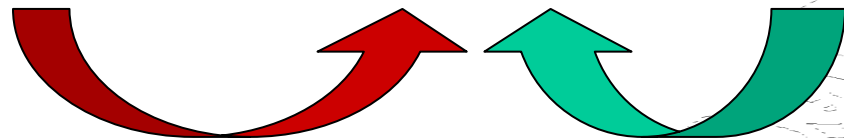




# Contextual Rules

- Add recall
- Low-precision/high-recall rules
- Used
  - when best rules not able to identify/shift some information
    - e.g., missing end or start tag
- Surrounding tags used to constrain rule application

The seminar will be given by `<speaker>` A. Padua `</speaker>` at `<time>` 12.30



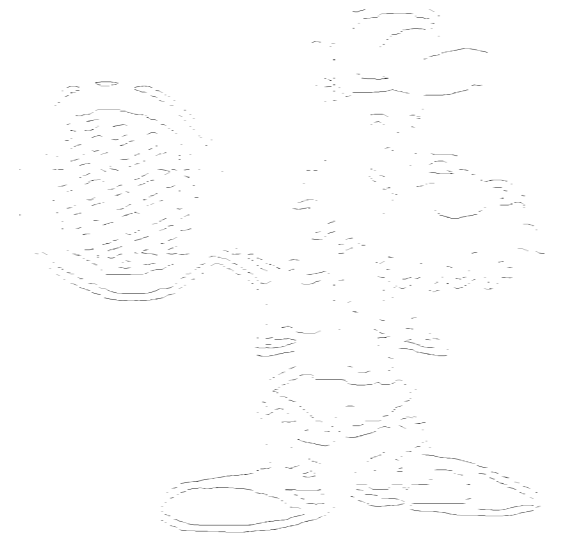


# Contextual Rule: Example

## Rule

**Word=\* </speaker> word=at**

- is not a Best Rule (high recall/low precision)
- is reliable if applied to close an open **<speaker>** only!





# Imprecision in Tagging

Border identification can be tricky

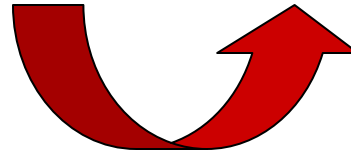
- Up to 5% imprecision for some information
- Correction rules shift tags misplaced by tagging rules
  - Positive examples: tags misplaced within a distance  $d$  from the correct position
  - Negative examples: the correct tags





# Correction Rule: Example

The seminar at 4 *</stime>* PM will be held in Room 201



Initial Rule:

Condition		Action
word	wrong tag	correct tag
at		
4	<i>&lt;/stime&gt;</i>	
pm		<i>&lt;/stime&gt;</i>

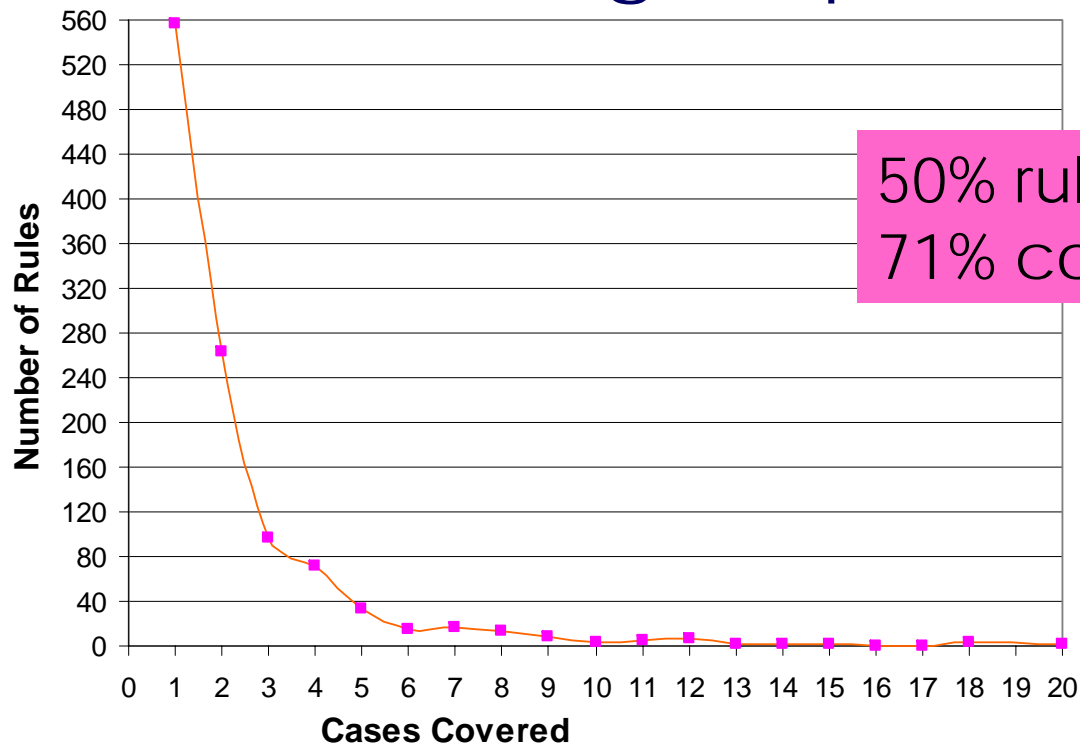




# Data Sparseness

Rules relying on word matching tend to:

- Report low coverage
  - Not able to relate cases
- Overfit training corpus



50% rules cover 1 case  
71% cover up to 2 cases





# Overfitting

- Rules with limited coverage overfit the training corpus
  - Sensitiveness to mistakes in tagging
    - Some good rules not accepted
      - Low recall at testing time
    - Some bad rules accepted
      - Low precision at testing time
- Solution:
  - Generalise rules to cover larger number of cases
  - Prune rule sets



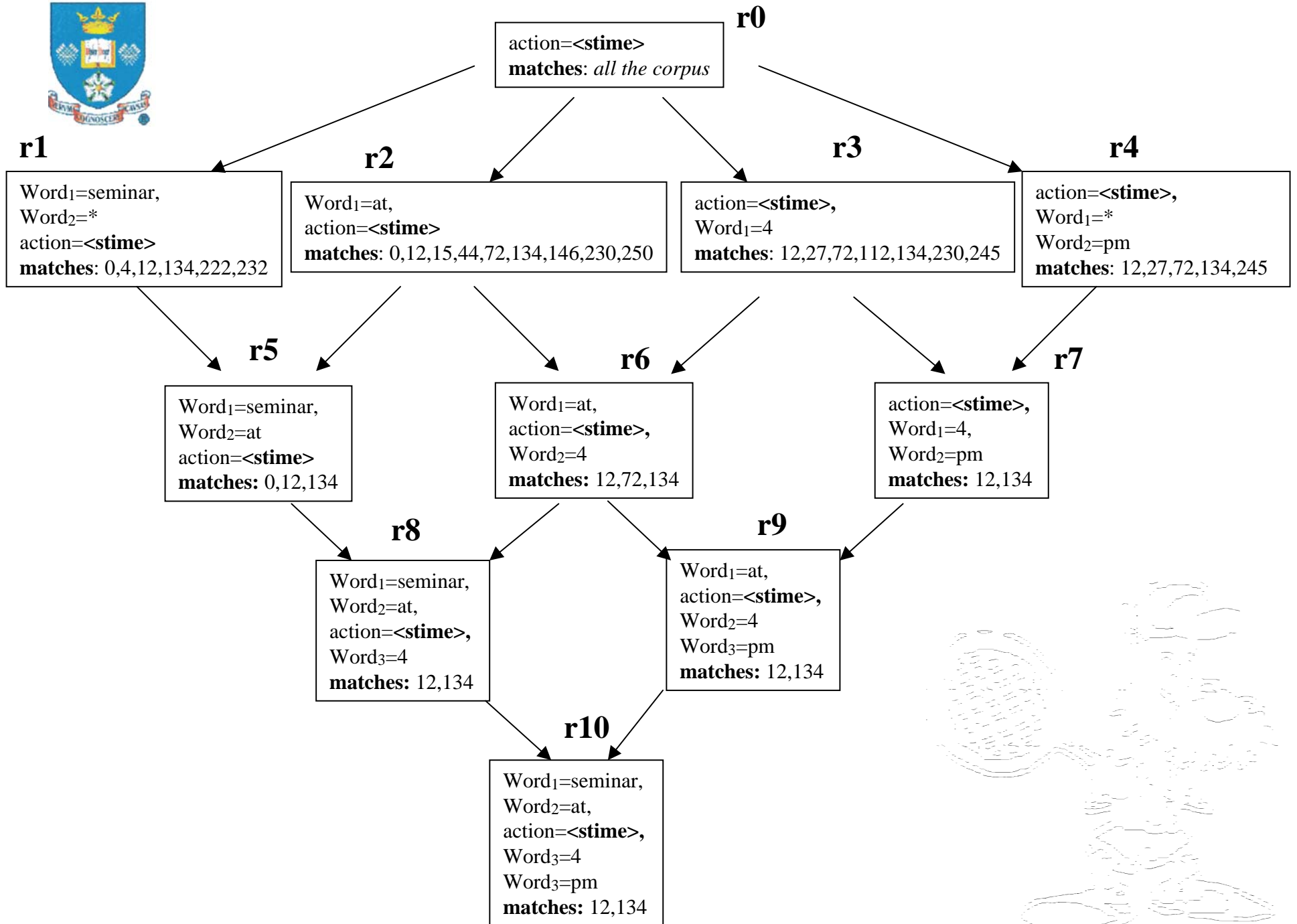


# Rule Generalisation

- Each instance is generalised by reducing its pattern in length
- Generalizations are tested on training corpus
- best  $k$  rules generated for each instance represent:
  - Smallest error
  - Greatest number of matches
  - Cover different examples

Implemented as a general to specific beam search with pruning.

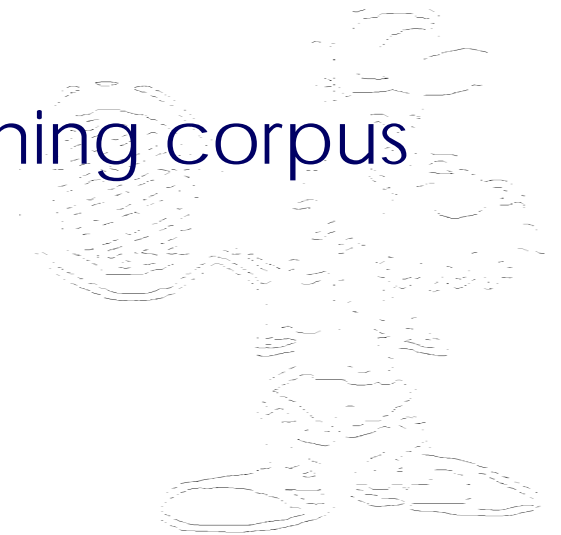
See paper for details





# Rule Pruning

- Generation time: prune rule if
  - error rate > threshold
  - coverage too small
  - coverage subsumed by more general rule with equivalent error rate
- Testing time:
  - N-cross folder cycle on training corpus
    - Unreliable rules removed
    - Error thresholds tuning





# NLP-based Adaptive IE

- NLP-based approaches
  - Effective on newspaper articles (e.g. MUC)
  - Ineffective on other types of texts
- Why?
  - NLP knowledge may be inadequate
    - Sublanguage phenomena
    - Need of adaptation of generic knowledge
      - Need an IE expert
  - Unparsability (e.g. HTML pages)





# Use of NLP in Wrappers

- Need:
  - Bridging the gap between NLP-based and Wrapper-like approaches
- Proposed solution:
  - Wrapper
    - has generic NLP knowledge
    - learns the level of NLP useful for the task at hand by measuring its effectiveness on the training corpus





# Wrapper with NLP-based Generalisation: (LP)<sup>2</sup>

Conditions on words are replaced by:

- Information from shallow NLP modules
  - Capitalisation
  - Morphological analysis
    - Generalizes over gender/number
  - POS tagging
    - Generalizes over lexical categories
  - User-defined dictionary or gazetteer
- All generalizations are tested and the best  $k$  derived from each instance are selected





# Effect of Generalization(1) Effectiveness with NLP

<i>Slot</i>	<b>(LP)<sup>2</sup><sub>G</sub></b>	<b>(LP)<sup>2</sup><sub>NG</sub></b>	} <u>Most Interesting</u>
speaker	72.1	14.5	
location	74.1	58.2	
stime	100	97.4	
etime	96.4	87.1	
<b>All slots</b>	<b>89.7</b>	<b>78.2</b>	

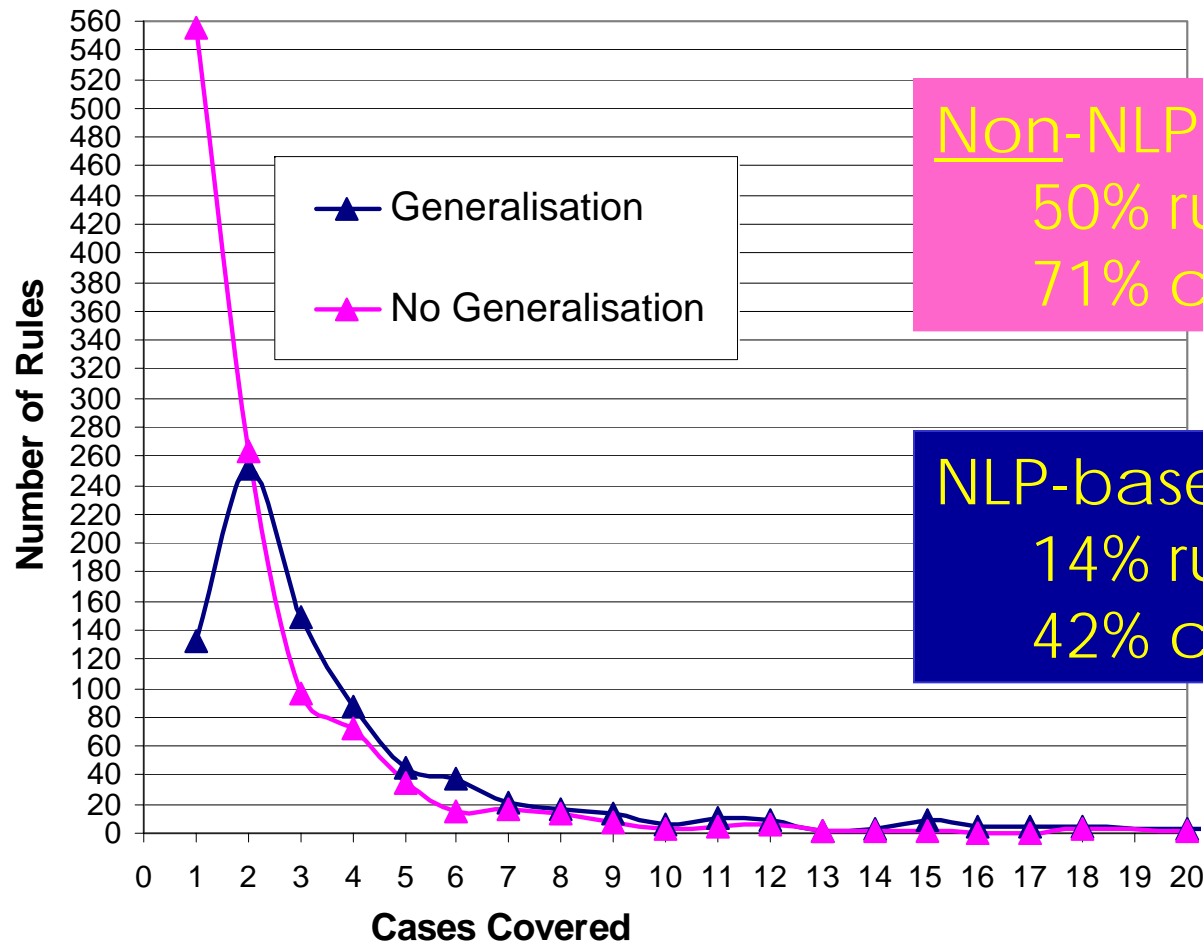
With comparable effectiveness on training corpus





# Effect of Generalization(2)

## Reduction in Data Sparseness



### Non-NLP

50% rules cover 1 case

71% cover up to 2 cases

### NLP-based generalisation

14% rules cover 1 case

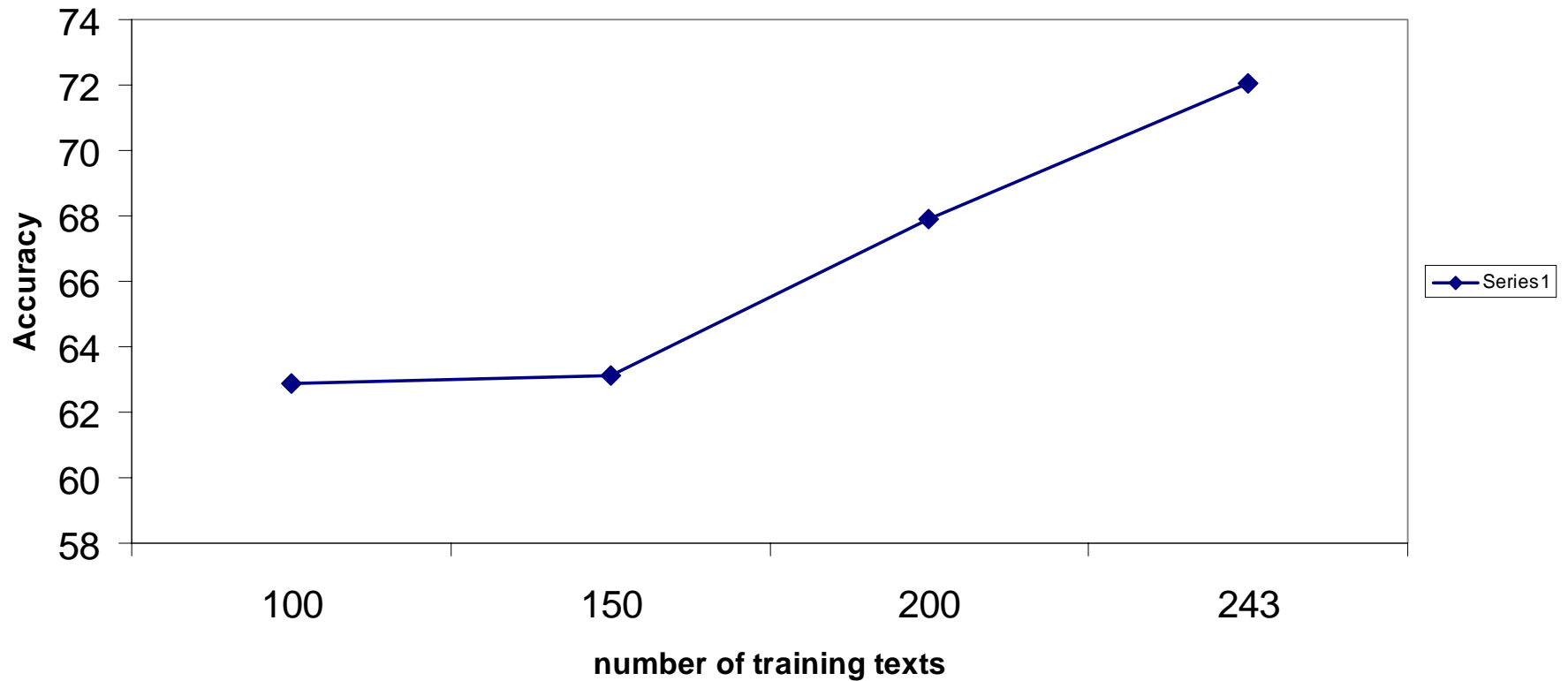
42% cover up to 2 cases





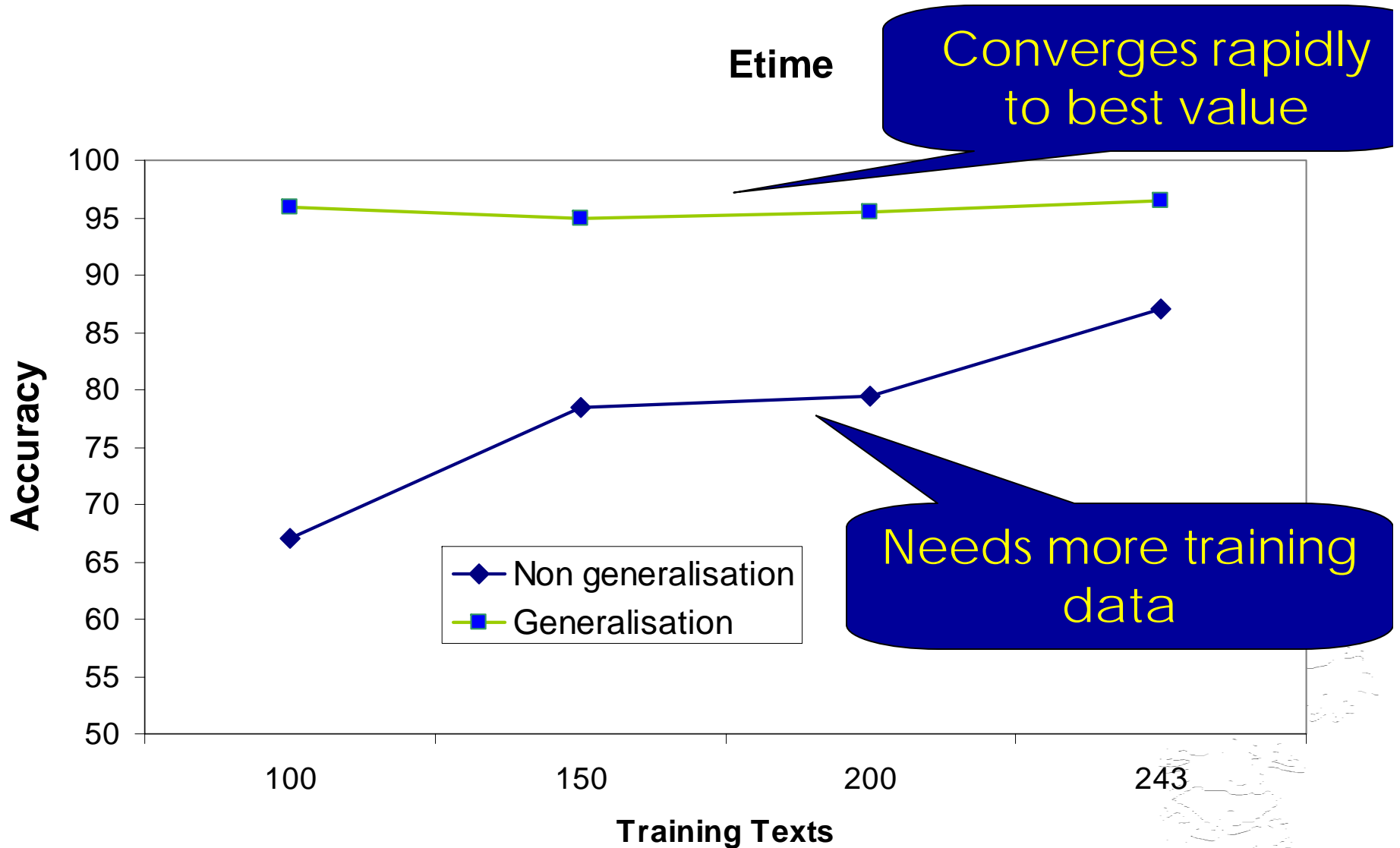
# Effect of Generalisation (3)

**<speaker> with generalisation**





# Effect of Generalisation(4)





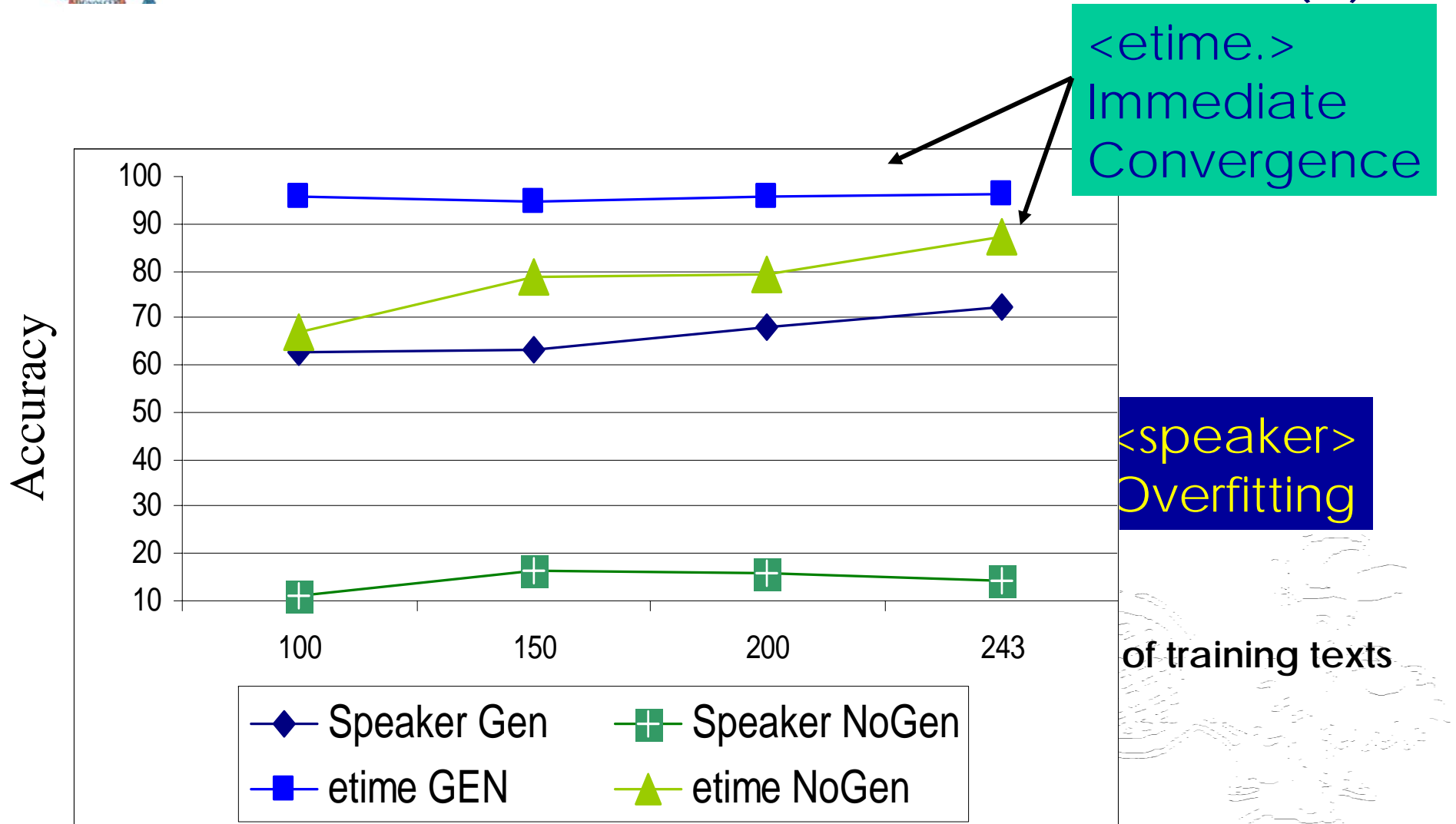
# Effect of Generalization (3)

	<b>(LP)<sup>2</sup><sub>G</sub></b>	<b>(LP)<sup>2</sup><sub>NG</sub></b>
Average rule coverage	10.2	6.2
Selected rules	887	1136
Rules covering 1 case	133 (14%)	560 (50%)
Rules covering >50 cases	37	15
Rules covering >100 cases	19	3





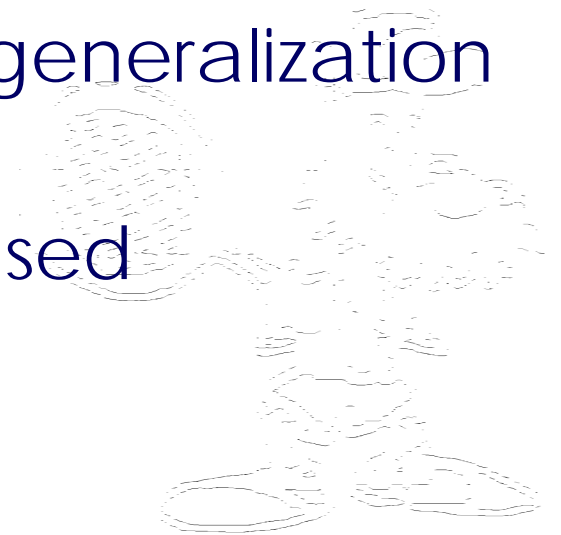
# Effect of Generalisation (4)





# Best Level of Generalization

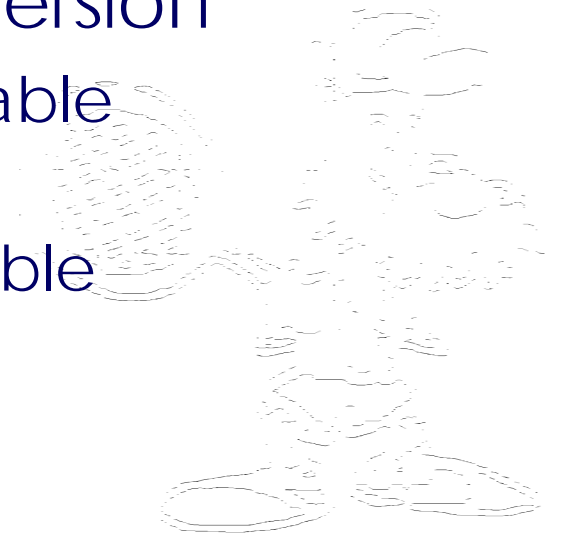
- Rules undergo selection during Learning
  - NLP-based generalization is accepted only if useful and does not introduce errors
  - Best level of NLP generalization is automatically selected
- Heuristics
  - If rule matches  $> n$  cases then generalization is used
  - Otherwise generalization is refused





# Best level of Generalization(2)

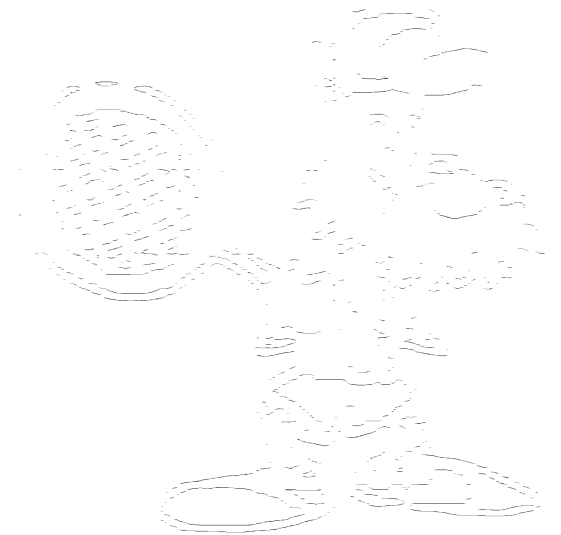
- Example
  - ITC seminar announcements (mixed I/E)
    - Date, time, location in Italian
    - Speaker, title and abstract in English
  - English POS also for the Italian part
  - NLP-based outperforms other version
    - Same accuracy when NLP unreliable
      - Date, time, location
    - Better accuracy when NLP is reliable
      - Speaker, title





# Conclusion & Future Work

- (LP)<sup>2</sup> selects the correct level of NLP processing needed for the task at hand
- Generalisation to be extended to:
  - Parsing: syntactic relations as alternative adjacency relations
  - Discourse processing





# (LP)<sup>2</sup>: Experimental results

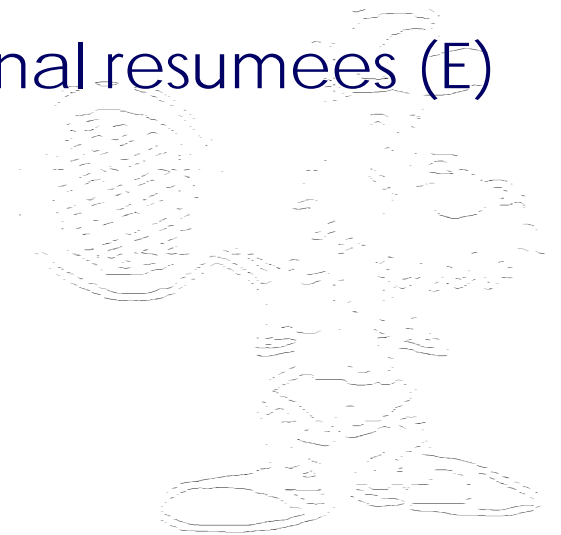
- Ported to
  - 6 scenarios
  - 2 languages
  - 2 text types (free and semi-structured)
- Naïve version used in Learning Pinocchio
  - a commercial system for adaptive IE
- The basis for Amilcare





# Experiments(2)

- Excellent results wrt state of art
  - CMU seminar announcements (E)
  - TX-Austin Job Announcements (E)
- Application to other corpora
  - ITC seminar announcements (mixed I/E)
- Commercial Applications
  - “Tombstone” data from professional resumes (E)
  - IE from financial news (I)
  - IE from classified ads (I)
  - (...)

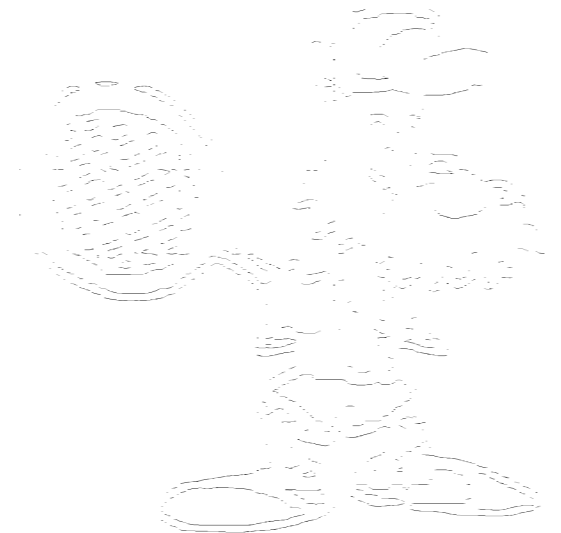




# CMU: detailed results

	<b>(LP)<sup>2</sup></b>	<b>BWI</b>	<b>HMM</b>	<b>SRV</b>	<b>Rapier</b>	<b>Whisk</b>
speaker	<b>77.6</b>	67.7	76.6	56.3	53.0	18.3
location	75.0	76.7	<b>78.6</b>	72.3	72.7	66.4
stime	99.0	<b>99.6</b>	98.5	98.5	93.4	92.6
etime	95.5	93.9	62.1	77.9	<b>96.2</b>	86.0
All Slots	<b>86.0</b>	<b>83.9</b>	<b>82.0</b>	<b>77.1</b>	<b>77.3</b>	<b>64.9</b>

1. Best overall accuracy
2. Best result on speaker field
3. No results below 75%





# Jobs: Detailed Results

	LearningPinocchio			Rapiet		
	Rec	Prec	F(1)	Rec	Prec	F(1)
id	100	100	100.00	98.0	97.0	97.50
title	37	54	43.91	67.0	29.0	40.48
salary	53	77	62.78	89.2	54.2	67.43
company	66	79	71.92	76.0	64.8	69.49
recruiter	75	87	80.56	87.7	56.0	68.35
state	90	80	84.71	93.5	87.1	90.19
city	94	92	92.99	97.4	84.3	90.38
country	96	70	80.96	92.2	94.2	93.19
language	90	92	90.99	95.3	71.6	80.85
platform	80	81	80.50	92.2	59.7	72.47
application	72	86	78.38	87.5	57.4	69.32
area	64	70	66.87	66.6	31.1	42.40
req-years-e	61	79	68.84	80.7	57.5	67.15
desired-yrn	55	67	60.41	94.6	81.4	87.50
req-degree	80	90	84.71	88.0	75.9	81.50
desired-deg	51	90	65.11	86.7	61.9	72.23
post-date	100	99	99.50	99.3	99.7	99.50
<b>ALL SLOTS</b>	<b>82</b>	<b>86</b>	<b>84.14</b>	<b>89.4</b>	<b>64.8</b>	<b>75.14</b>

84.14

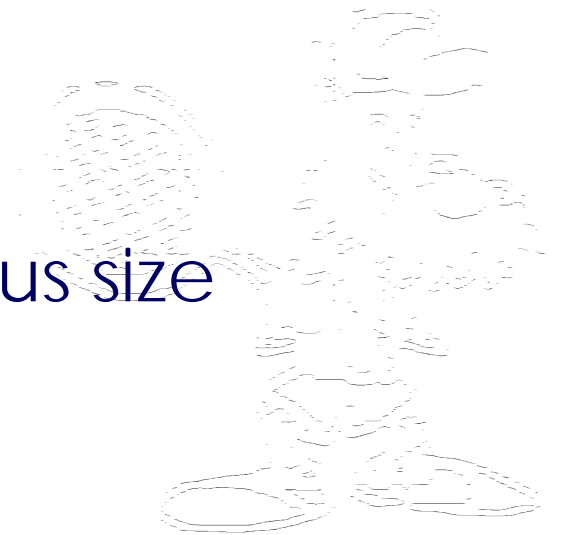
75.14





# Algorithm's Pros

- Single tag rules
- Contextual rules
- Correction rules
  
- NLP-based generalization:
  - Reduces data sparseness
  - Reduces needed training corpus size





Amilcare

Thank You!