Intrinsic and extrinsic evaluation of a deep lexical type supertagger

João Silva

Faculty of Sciences, University of Lisbon
NLX — Natural Language and Speech Group

10th DELPH-IN Summit
Tomar, Portugal
July 14–18, 2014

Presentation outline

Introduction and recap

Extrinsic evaluation

Closing remarks

Presentation outline

Introduction and recap

Extrinsic evaluation

Closing remarks

Previously...

SVM-TK classifier: The goal

Assign deep lexical types to unknown words

- LX-Gram, an HPSG for Portuguese
  - Currently, generics for unknown word handling
    shallow processing using LX-Suite (POS $\mapsto$ default deep type)
- Make use of structured features

  syntactic constituency, grammatical dependencies, etc.
- Disambiguated lexical types, on-the-fly, off-the-shelf tools

Previously...

SVM-TK classifier: The goal

Assign deep lexical types to unknown words

- LX-Gram, an HPSG for Portuguese
  - Currently, generics for unknown word handling
    shallow processing using LX-Suite (POS $\mapsto$ default deep type)
- Make use of structured features

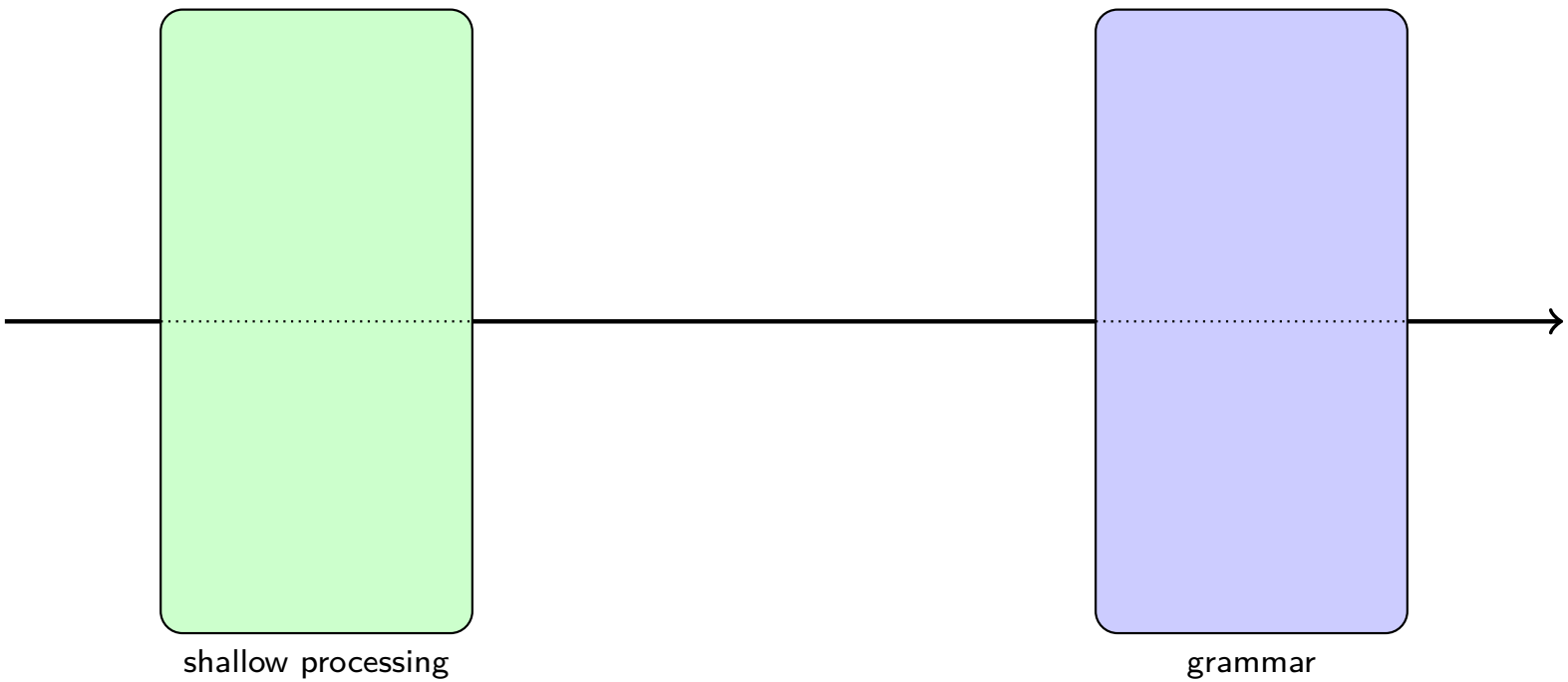
  syntactic constituency, grammatical dependencies, etc.
- Disambiguated lexical types, on-the-fly, off-the-shelf tools

The approach

- Support-vector machine (SVM) classifier
- Tree kernels (TK) to allow using structured features

Previously...

SVM-TK classifier: The central idea

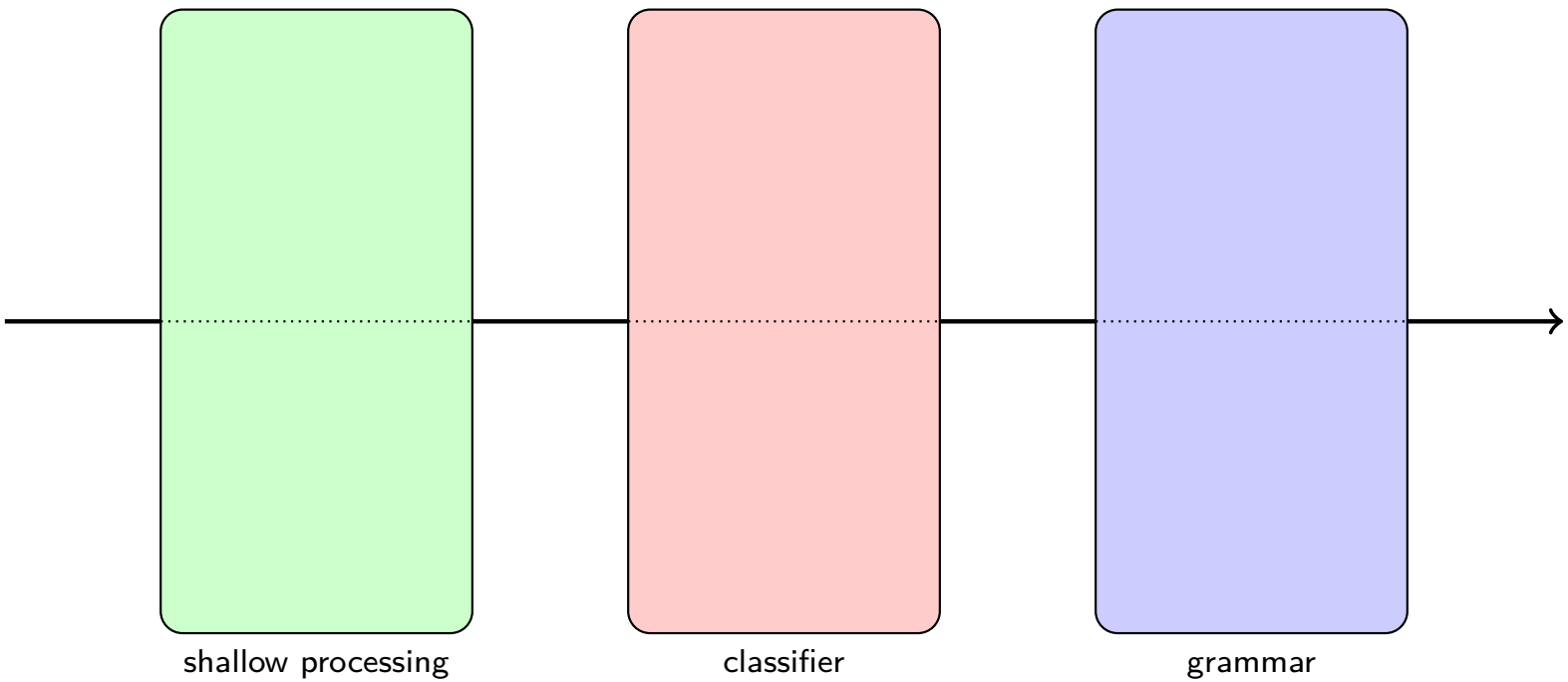


Previously...

SVM-TK classifier: The central idea

The classifier

- Is placed between the shallow processing and the grammar

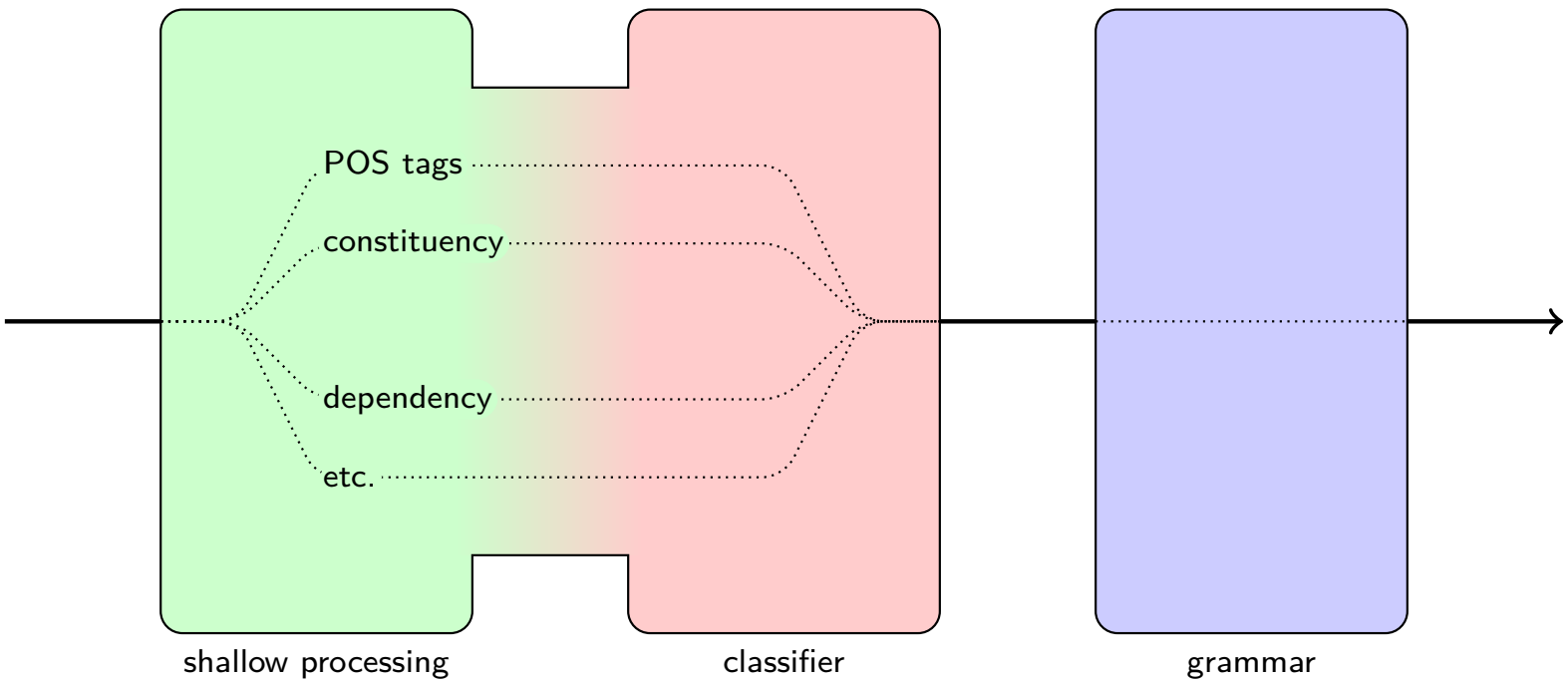


Previously...

SVM-TK classifier: The central idea

The classifier

- Is placed between the shallow processing and the grammar
- Combines shallow information and picks a single lexical type



Previously...

SVM-TK classifier: Evaluation

- Intrinsic evaluation
  - Top-*n* most frequent verb types
  - Comparison with other approaches
    (viz. SVMTool: state-of-the-art supertagger)
  - Running over gold dependencies vs. predicted dependencies
  - Varying the size of the training dataset
  - Changing the grammar and language (ERG for English)

Previously...

SVM-TK classifier: Evaluation

- Intrinsic evaluation
  - Top-*n* most frequent verb types
  - Comparison with other approaches
    (viz. SVMTool: state-of-the-art supertagger)
  - Running over gold dependencies vs. predicted dependencies
  - Varying the size of the training dataset
  - Changing the grammar and language (ERG for English)

	SVM-TK	SVMTool
top-10	**94.76**	94.20
top-20	90.27	**92.49**
top-30	89.04	**92.48**

LX-Gram / CINTIL

Previously...

SVM-TK classifier: Evaluation

- Intrinsic evaluation
  - Top-*n* most frequent verb types
  - Comparison with other approaches
    (viz. SVMTool: state-of-the-art supertagger)
  - Running over gold dependencies vs. predicted dependencies
  - Varying the size of the training dataset
  - Changing the grammar and language (ERG for English)

	SVM-TK	SVMTool
top-10	**94.76**	94.20
top-20	90.27	**92.49**
top-30	89.04	**92.48**

LX-Gram / CINTIL

	SVM-TK	SVMTool
top-19	**93.05**	91.53
top-41	**91.63**	89.63
top-56	**90.93**	88.80

ERG / Redwoods

Presentation outline

Extrinsic evaluation

Overview

So, SVM-TK is the best supertagger and everyone is happy. ☺
But what happens when LX-Gram is run with and without it?

Extrinsic evaluation

Overview

So, SVM-TK is the best supertagger and everyone is happy. ☺
But what happens when LX-Gram is run with and without it?

The experiment

- 5,000 previously unseen sentences
- SVM-TK for top-10 verbs, over predicted dependencies
- Manual treebanking, with and without SVM-TK
  (many thanks to our annotators for their help)

Extrinsic evaluation

Coverage results

Without/with SVM-TK,
the 4 cases:

- case $[--]$,
  failure in both situations;
- case $[-+]$,
  becomes parsable;
- case $[+-]$,
  becomes unparsable;
- and case $[++]$,
  no parsability change.

Extrinsic evaluation

Coverage results

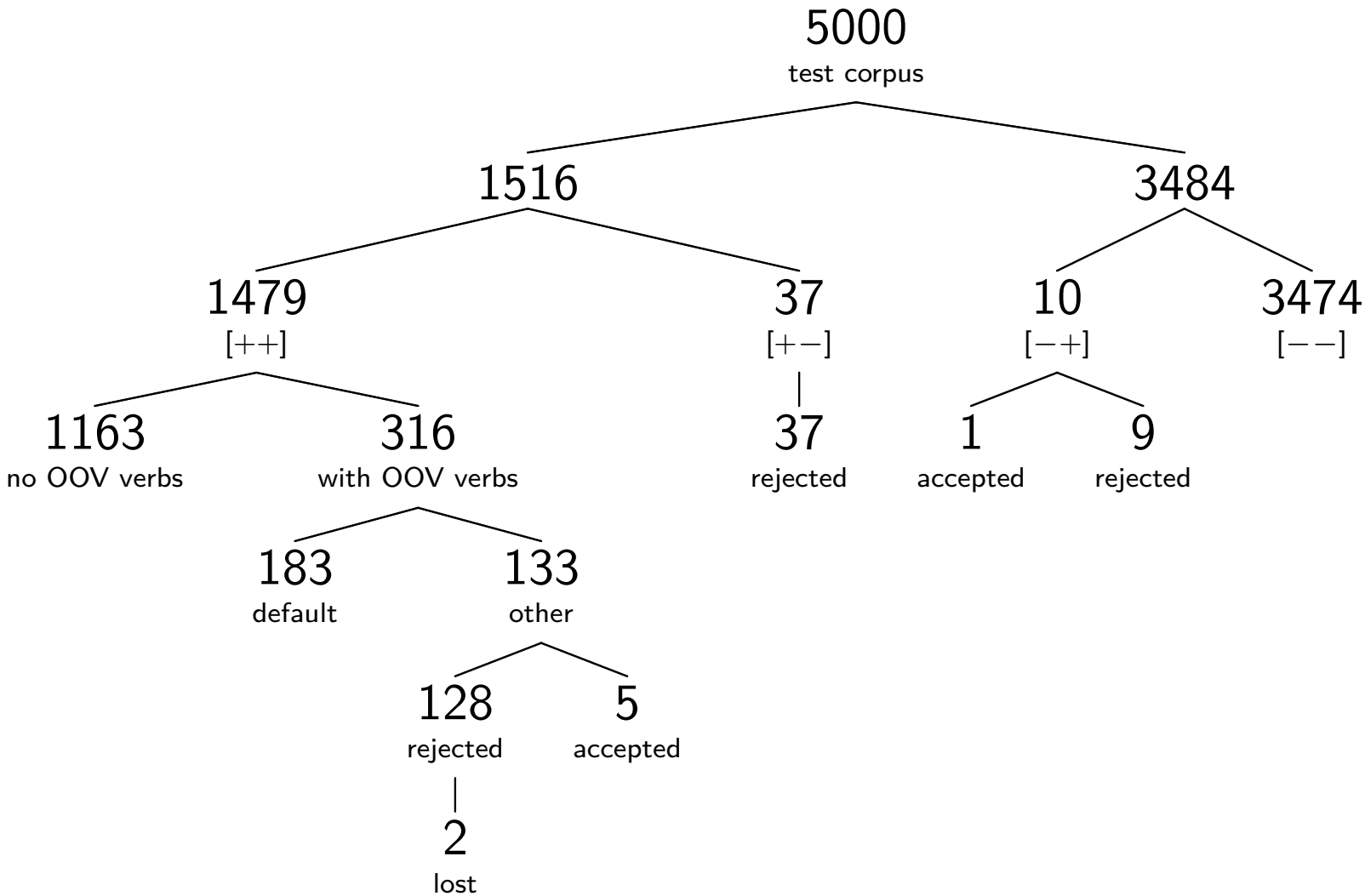
Without/with SVM-TK,
the 4 cases:

- case $[--]$,
  failure in both situations;
- case $[-+]$,
  becomes parsable;
- case $[+-]$,
  becomes unparsable;
- and case $[++]$,
  no parsability change.

case	sent.
$[--]$	3474
$[-+]$	10
$[+-]$	37
$[++]$	1479
total	5000

Extrinsic evaluation

Correctness results



Presentation outline

Comments on the extrinsic experiment

Improvements in coverage and correctness

- 10 parses gained, with 1 accepted
- 37 parses dropped, but none was acceptable
- Lost 2 accepted parses, gained 5 (+3 net)

Comments on the extrinsic experiment

Improvements in coverage and correctness

- 10 parses gained, with 1 accepted
- 37 parses dropped, but none was acceptable
- Lost 2 accepted parses, gained 5 (+3 net)

Not as good as I wanted/expected

- Top-10 most frequent types
- SVM-TK often assigns the default
- There are issues apart of OOV words
- Etc.

Thesis summary

- What is the problem?
  Assign deep lexical types to OOV words.

Thesis summary

- What is the problem?
  Assign deep lexical types to OOV words.
- What is the main insight?
  Structured features are bound to help,
  and we can get them from robust methods.

Thesis summary

- What is the problem?
  Assign deep lexical types to OOV words.
- What is the main insight?
  Structured features are bound to help,
  and we can get them from robust methods.
- What is new?
  A classifier that combines multiple shallow processes,
  with tree kernels to allow capturing structured features.

Thesis summary

- What is the problem?
  Assign deep lexical types to OOV words.
- What is the main insight?
  Structured features are bound to help,
  and we can get them from robust methods.
- What is new?
  A classifier that combines multiple shallow processes,
  with tree kernels to allow capturing structured features.
- What were the results?
  Improves on the state-of-the-art supertagging methods.
  A general harness to experiment with various shallow methods.

Thank you,
and congratulations to DELPH-IN!