STRUCTURED KERNEL-BASED LEARNING FOR THE FRAME LABELING OVER ITALIAN TEXTS

Danilo Croce, Emanuele Bastianelli and Giuseppe Castellucci

University of Rome Tor Vergata

Evalita 2011, Roma 24th January 2012

Introduction

- In the Semantic Role Labeling (SRL) task language learning systems usually generalize linguistic observations into statistical models
  - Symbolic expressions derived from the parse trees denote the position and the relationship between a predicate and its arguments
- Which are the most effective linguistic features?
  - Manual feature engineering
  - Kernel based methods

Tree Kernel methods

- With Tree Kernel based methods, Syntactic information of annotated examples can be effectively generalized in SRL
  - Tree Kernels model similarity between two training examples as a function of their shared tree fragments
  - Discriminative information is selected by the learning algorithm, e.g. SVM.

- … but the information derived from structural patterns is not always sufficient:
  - For example "*The man said . . .* " and "*The mail said . . .* " evoke the same frame
  - … but the logical subject represents 2 different roles
  - *man* is a COMMUNICATOR, while *mail* is a MEDIUM

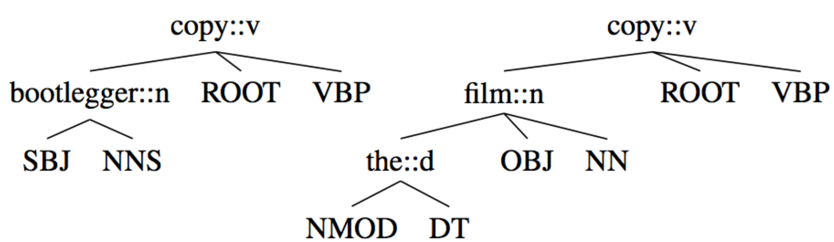
Smoothed Partial Tree Kernels (SPTK)

- In the EMNLP 2011 (Croce et al,'11) paper a family of convolution kernels for dependency structures aiming at **jointly modeling syntactic** and **lexical semantic similarity** is proposed.

- The idea is to provide a similarity score among tree nodes depending on the semantic similarity among the node labels
  - define a structured notion of similarity between trees, whereas (lexical) nodes are semantically similar
  - The lexical similarity can be acquired **automatically** through the analysis of a corpus
  - The representation space is implicit

Formal definition

[*Bootleggers*]$_{CREATOR}$, *then* **copy** [*the film*]$_{ORIGINAL}$ [*onto hundreds of VHS tapes*]$_{GOAL}$



σ(n_1,n_2) is a similarity function among the tree nodes depending on their linguistic type

- Given two trees T1,
  - If n1 and n2 are leaves then

$$\Delta_\sigma(n_1,n_2) = \mu\lambda\sigma(n_1,n_2)$$

else

$$\Delta_\sigma(n_1,n_2) = \mu\sigma(n_1,n_2) \times \left(\lambda^2 + \sum_{\vec{I}_1,\vec{I}_2,l(\vec{I}_1)=} \lambda^{d(\vec{I}_1)+d(\vec{I}_2)} \prod_{j=1}^{l(\vec{I}_1)} \Delta_\sigma(c_{n_1}(\vec{I}_{1j}), c_{n_2}(\vec{I}_{2j}))\right)$$

Syntactic information and Drawbacks

- The adoption of syntactic features can be problematic
  - The quality of the method is strongly connected to the quality of the syntactic parser
  - Moreover in (Johansson&Nugues,2008) only the 82% of roles are grammatically recognized
  - Syntactic features without a strong lexical information provide a poor domain adaptation

SRL as a sequential tagging problem

- In the AI*IA 2011 (Croce et al, '11) paper the SRL task is modeled as a sequential tagging:
  - **Adopting shallower grammatical features** (e.g. POS n-grams), i.e. no esplicit syntax
  - Making the learning process sensible to syntagmatic information within **a structured ML schema**, i.e. SVM^{HMM}
  - **Improving lexical generalization** through distributional vector space lexical semantic models

SRL and Structured Learning

[Yesterday]$_{TIME}$,[a robber]$_{KILLER}$ **killed** [a guardian]$_{VICTIM}$ [with a knife]$_{INSTR}$.

- SRL and classification – the BIO notation
  - Boundary detection

Yesterday/`B` , /`X` a/`B` robber/`O` killed/`X` a/`B` guardian/`O` with/`B` a/`I` knife/`O` ./`X`

  - Argument classification

Yesterday/`Time` , /`X` a/`Killer` robber/`Killer` killed/`LU` a/`Victim` guardian/`Victim` with/`Instr` a/`Instr` knife/`Instr` ./`X`

The SVM-SPTK system

- It is based on the semantically Smoothed Partial Tree Kernel
- No manual feature engineering

Task	Classification schema	Instances	Target Class
Frame Prediction	Multi-classification	The dependency parse tree of each sentence	All frames
Boundary Detection	Binary classification	The dependency parse tree nodes	The node covers/does not cover an argument span
Argument Classification	Multi-classification	The dependency parse tree nodes covering an argument	The Frame Elements of a frame

The SVM-SPTK system

- SVM-Multiclass (FP) – SVM^{HMM} (BD and AC)
- Manual feature engineering
- No explicit syntax

Task	Classification schema	Instances	Target Class
Frame Prediction	Multi classification	A sentence (words and POS n-grams)	All frames
Boundary Detection	Sequence Labeling	Manually define feature vectors (lexical, grammatical and semantic features)	BIO tags
Argument Classification			The Frame Elements

Results (1)

- Frame Prediction

	Accuracy
SVM-SPTK	**80.82%**
SVM-HMM	78.62%

- Boundary Detection

		Argument Based			Token Based		
		Precision	*Recall*	*F1*	*Precision*	*Recall*	*F1*
First Run	*SVM-SPTK*	**66,67%**	**72,50%**	**69,46%**	**81,99%**	**84,34%**	**83,15%**
	SVM-HMM	50,70%	51,43%	51,06%	68,02%	77,18%	72,31%
Second Run	*SVM-SPTK*	66,67%	**72,50%**	**69,46%**	81,99%	84,34%	83,15%
	SVM-HMM	49,91%	50,36%	50,13%	68,14%	76,69%	72,16%

Results (2)

- Argument Classification

		Argument Based			Token Based		
		Precision	*Recall*	*F1*	*Precision*	*Recall*	*F1*
First Run	*SVM-SPTK*	**48,44%**	**52,68%**	**50,47%**	**62,58%**	**64,38%**	**63,47%**
	SVM-HMM	33,10%	33,57%	33,33%	46,77%	53,06%	49,72%
Second Run	*SVM-SPTK*	51,23%	**55,71%**	**53,38%**	**69,01%**	**70,99%**	**69,99%**
	SVM-HMM	37,52%	37,86%	37,69%	54,63%	61,48%	57,86%
Third Run	*SVM-SPTK*	70,36%	**70,36%**	**70,36%**	**78,35%**	**78,35%**	**78,35%**
	SVM-HMM	66,67%	65,36%	66,01%	77,71%	77,46%	77,59%

Conclusions

- The SVM-SPTK system is based on the Smoothed Partial Tree Kernel
  - It Implicitly combines syntactic and lexical information
  - No manual feature engineering
  - State-of-the-art results are achieved in almost all the challenge tasks.

- The SVM-HMM system is based on the Markovian formulation of the Structural SVM learning algorithm.
  - It represents a very flexible approach for SRL
  - Results are lower with respect to the SVM-SPTK, but in line with the other systems in most runs.
  - It does not rely on a full syntactic parsing of sentences.

Thanks for the attention

Many thanks to Roberto Basili and Alessandro Moschitti for their advices

…and thanks to Emanuele Bastianelli and Giuseppe Castellucci for

BABEL

BIO **A**nnotation **B**ased **E**ngine for sr**L**