Kernel-based Learning for Natural Language Processing tasks

Roberto Basili

DII, Università di Roma, Tor vergata,

Joint work with D. Croce, A. Moschitti, D. Pighin,

UNIVERSITA' degli STUDI di ROMA TOR VERGATA

Artificial Intelligence at Roma Tor Vergata

Overview

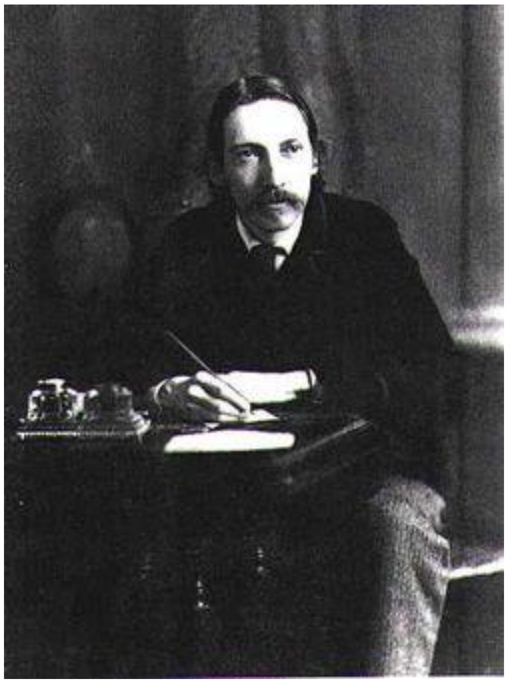
- Session I: **Machine Learning for NLP**
  - Support Vector Machines for NLP
  - Kernels for HLTs
    - Sequence and Tree Kernels
- **Session II: Semantic Role Labeling**
  - Standard Linguistic Features for SRL
  - The role of Syntax
  - Future Work: Semantic Tree Kernels (SPTK)

NLP: an inductive perspective

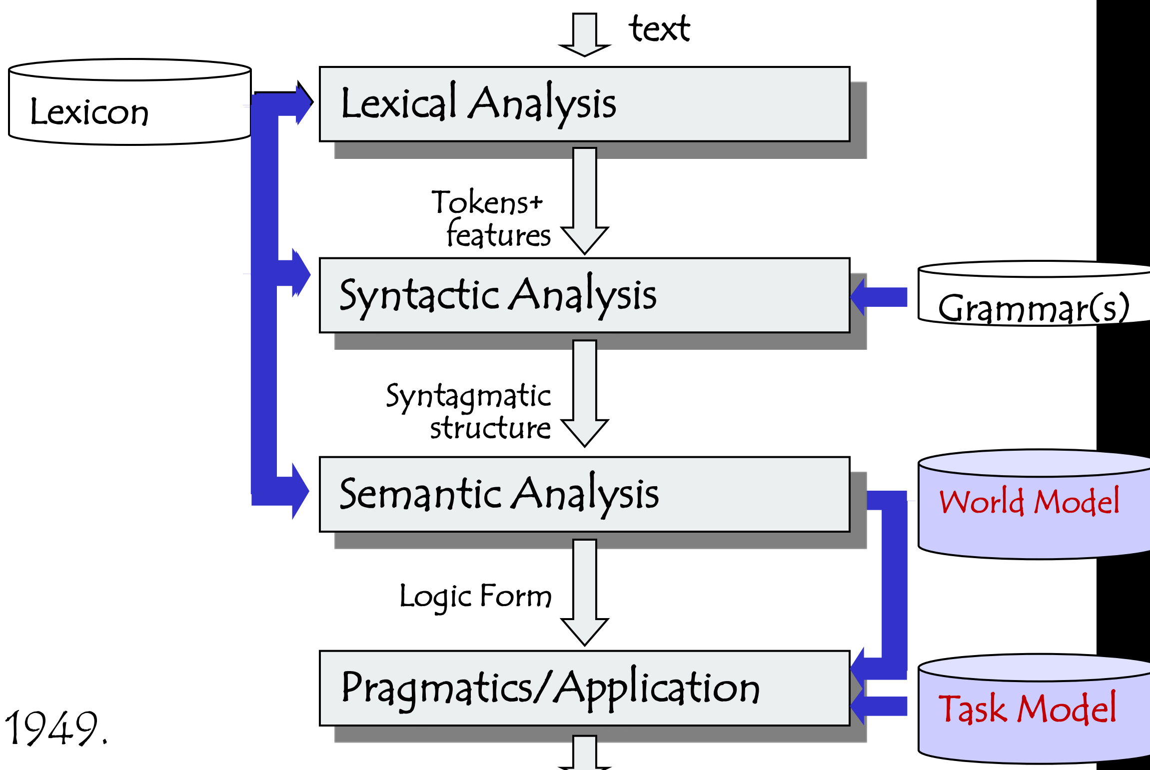
Speech and Language Processing

- What is'?
  - To develop programs able to accomplish linguistic tasks, such as:
    - To enable man-machine linguistic interaction
    - Improve communication among people (e.g. MT)
    - Manipulate linguistic objects (ad es. Web pages, documents o telephone calls)
  - Examples:
    - *Question Answering*
    - Machine Translation
    - Dialogue Agents

Language as a rule system

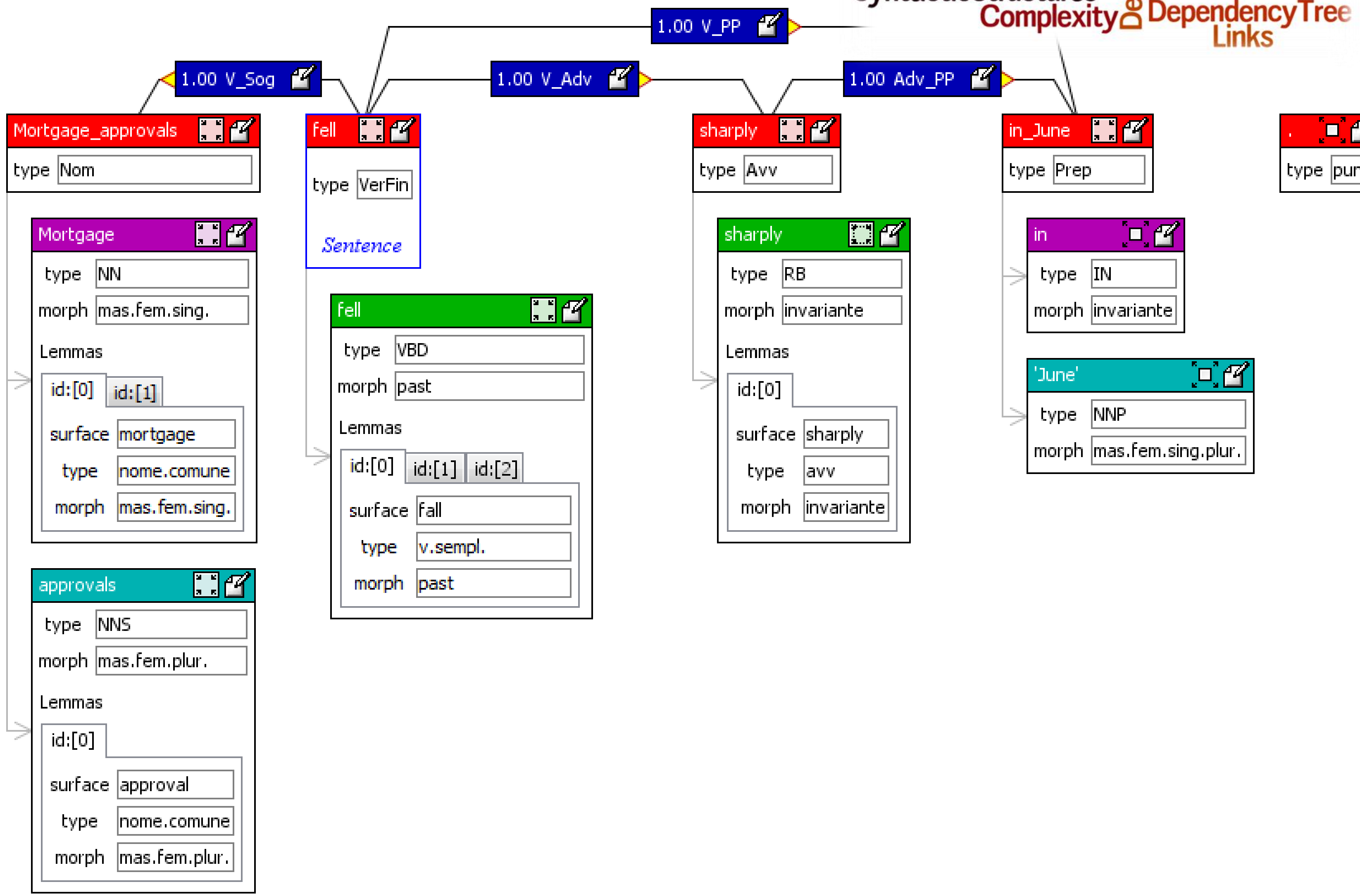


Every language is an alphabet of symbols the employment of which assumes a past **shared** by its interlocutors



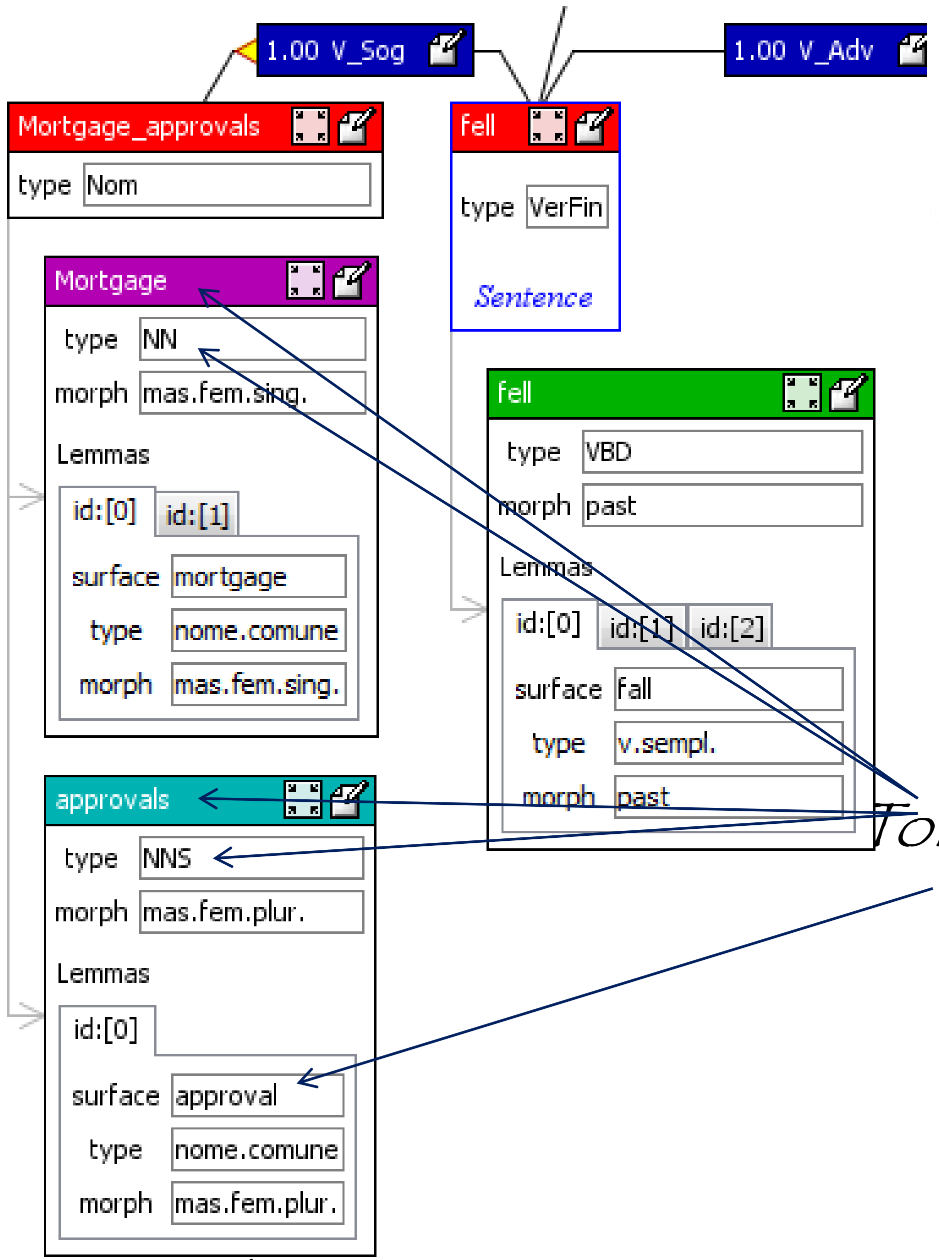
(*) J.L.Borges, "L'aleph", 1949.

What's in a Parse Tree?



FT (July, 29): *Mortgage approvals fell sharply in June*

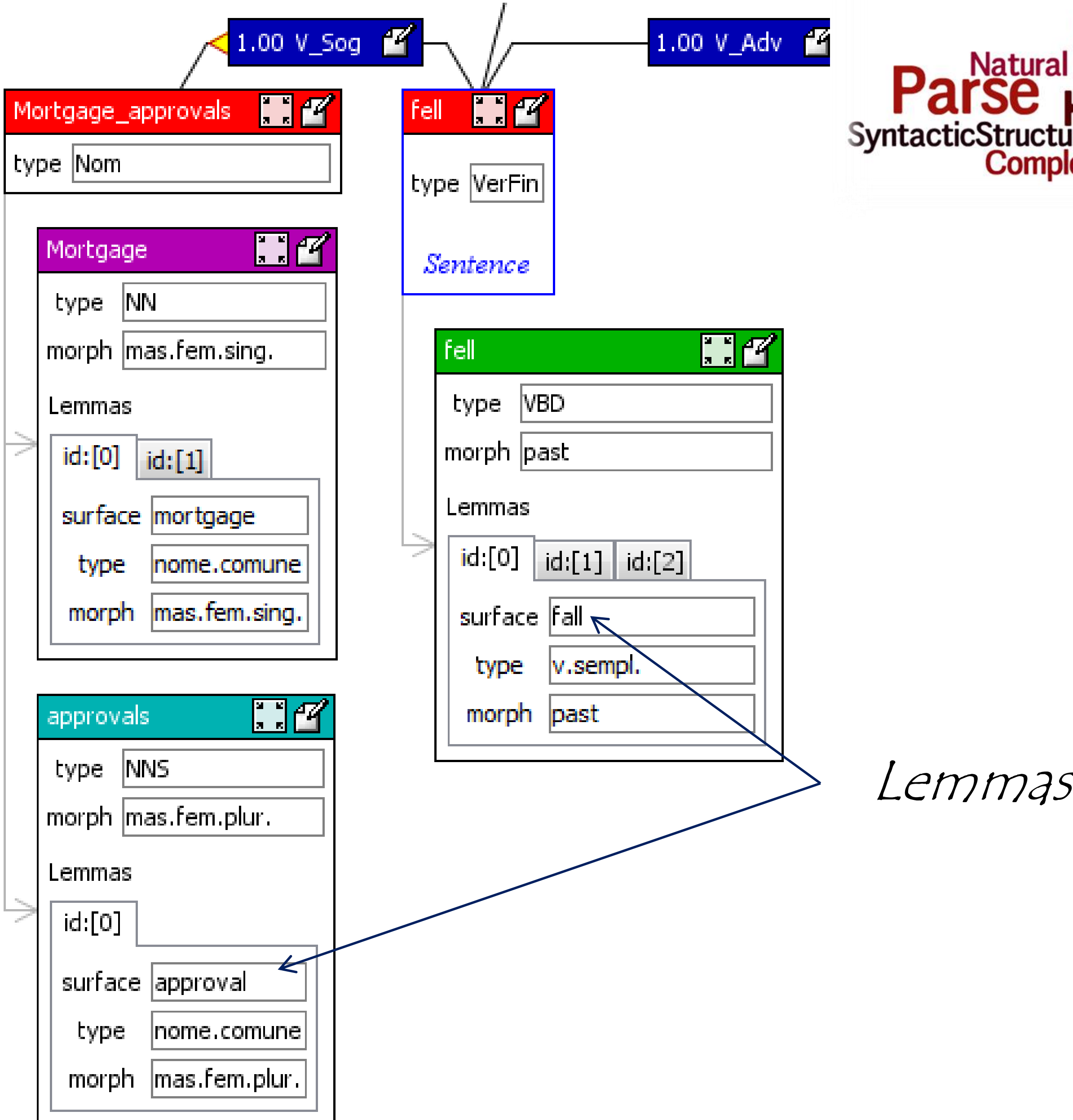
What's in a Parse Tree?



Tokens & POS tags

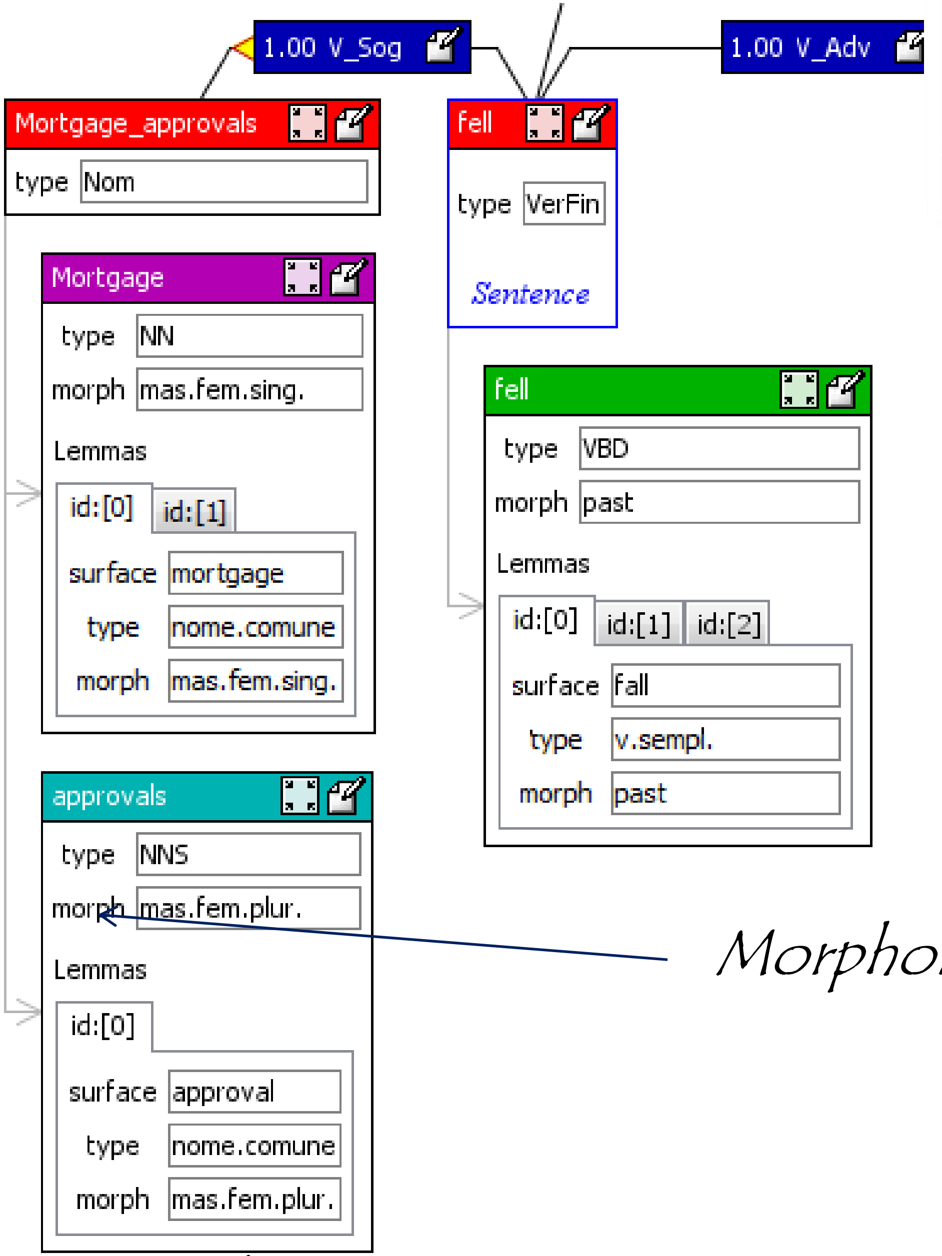
FT (July, 29): *Mortgage approvals fell sharply in*

What's in a Parse Tree?



FT (July, 29): *Mortgage approvals fell sharply in*

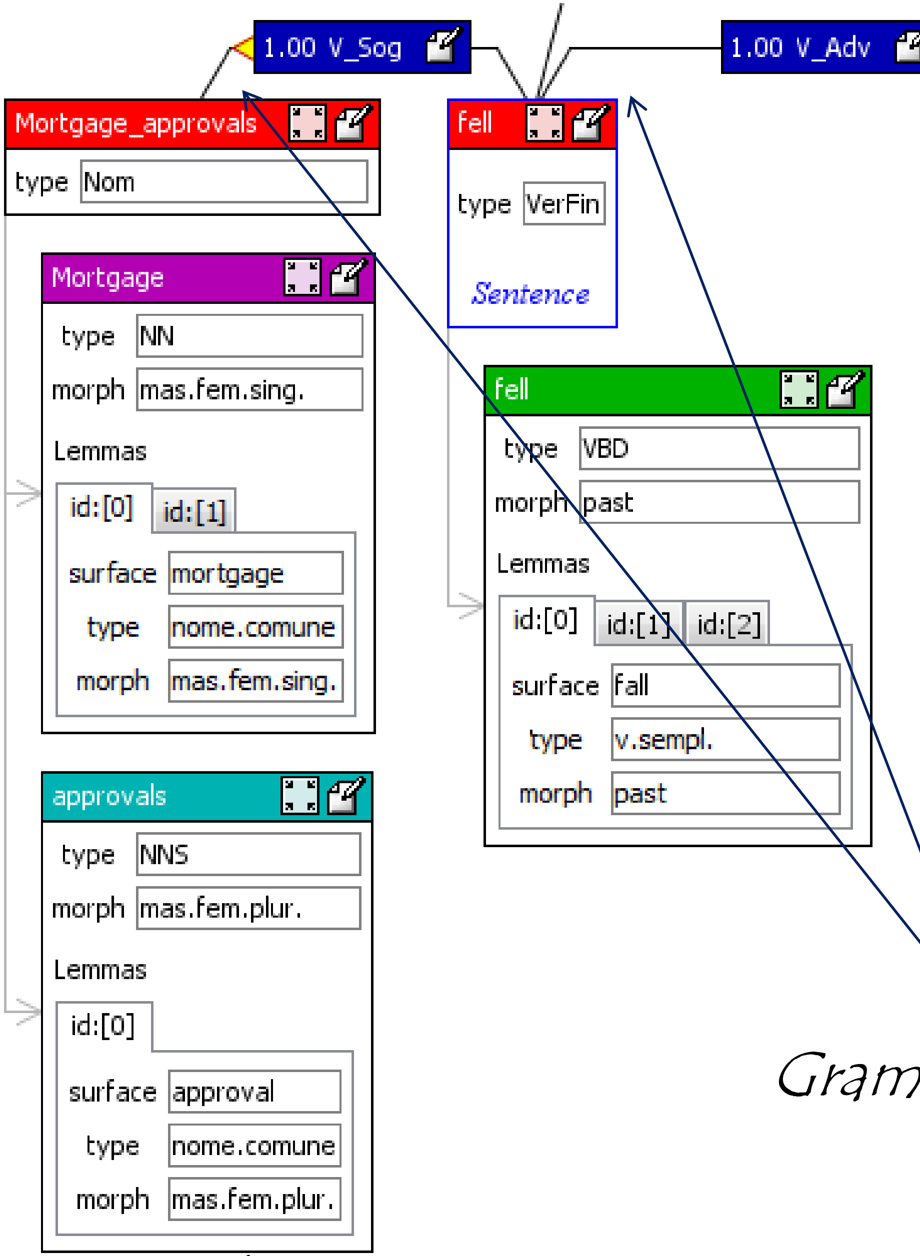
What's in a Parse Tree?



Morphological Features

FT (July, 29): *Mortgage approvals fell sharply in*

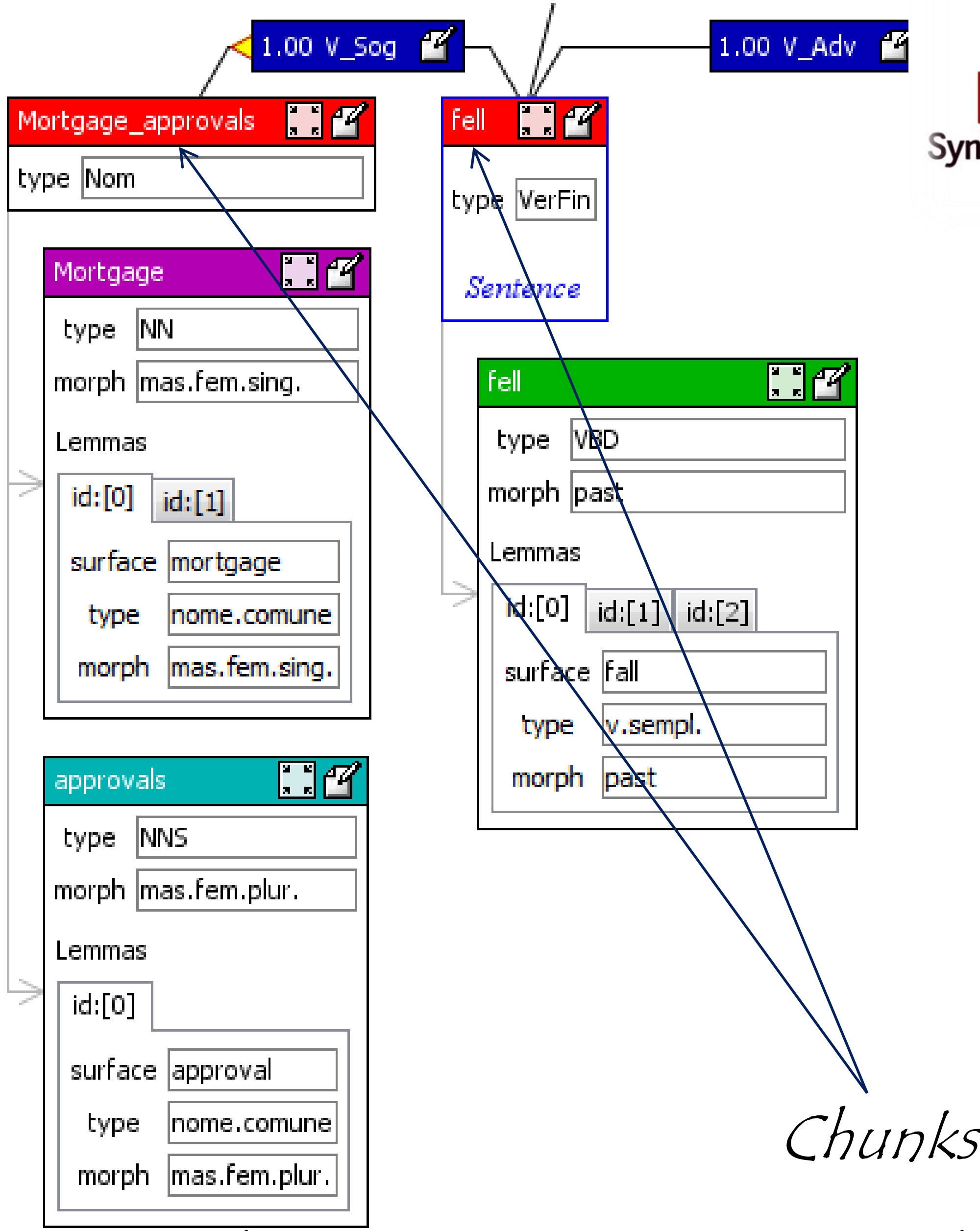
What's in a Parse Tree?



Grammatical Relation

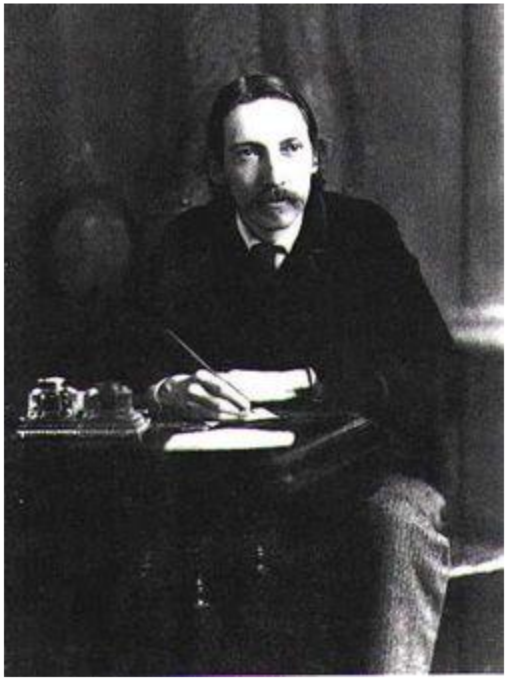
FT (July, 29): *Mortgage approvals fell sharply in*

What's in a Parse Tree?

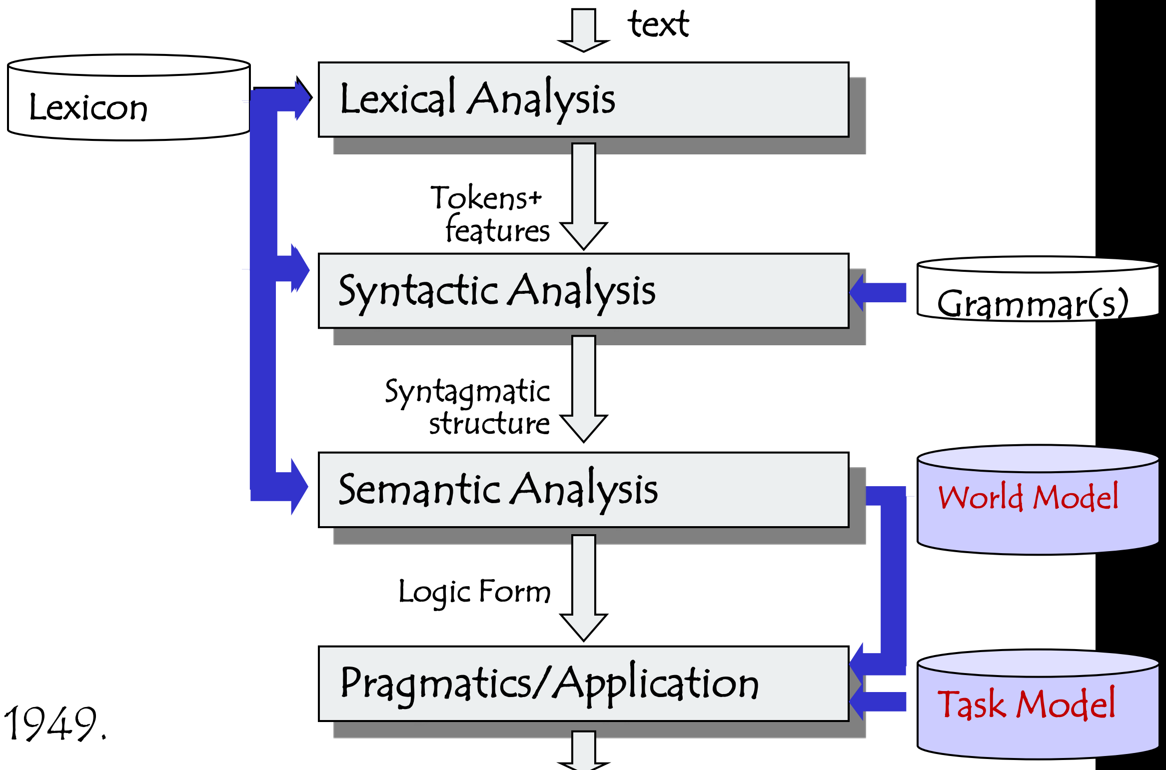


FT (July, 29): *Mortgage approvals fell sharply in*

Language as a rule system

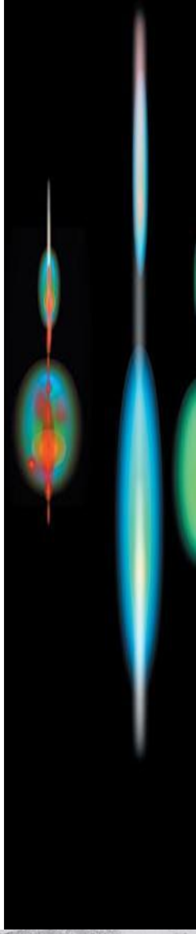


Every language is an alphabet of symbols the employment of which assumes a past **shared** by its interlocutors



(*) J.L.Borges, "L'aleph", 1949.

.. a different perspective



- … meaning is acquired and recognised within the language practice where it evolves
  - *The meaning of one word is determined by the rules of its use within a certain linguistic game*

L. Wittgenstein, *Philosophical Investigations* (1953).

- Capturing meaning from texts corresponds to link them to a common practice, throughout (possibly qualitative) equivalences and analogies

Lesson learned

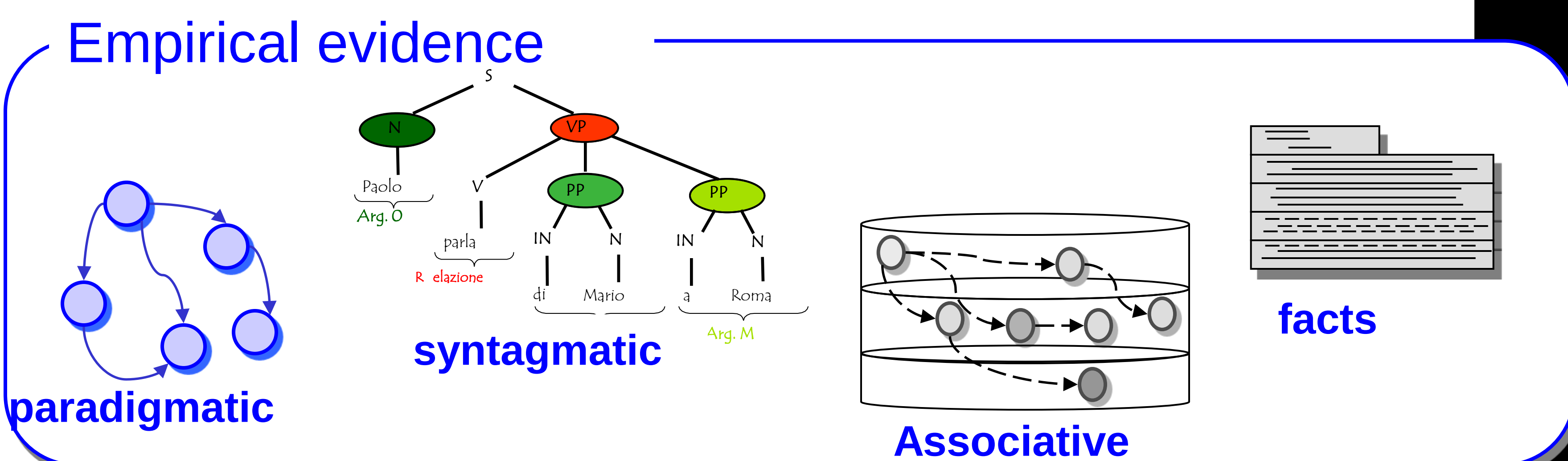
- Speech Recognition
- Empirical NLP/CL
  - Statistical parsing
  - Statistical MT
- Information retrieval
  - "*words stand for themselves*"
  - Content cannot be <u>recoded</u> in a general way *--* IR has gained from "*decreasing ontological expressiveness*"
  - Successful QA and IE are "superficial"

From linguistic data to knowledge

- Describing a meaning by labeling it outside the text is useful to consolidate the interpretation process but is hardly applied to linguistic recognition
- **Interpretation emerge from the experience of linguistic facts that share the same *context***
- It is a form of ***induction*** *from examples*

Vision

- Learning from scratch is not necessary and dangerous …
  - Linguistic *bias* : (basic) theory + representation
  - Inductive model: from data to knowledge
- … as much as the current Jelinek's view (LREC 2006)
- Induction:
  - statistics, neural networks, Support Vector Machines
- Representation + induction =
  linguistic knowledge



Linguistic inferences_ e.g. QA

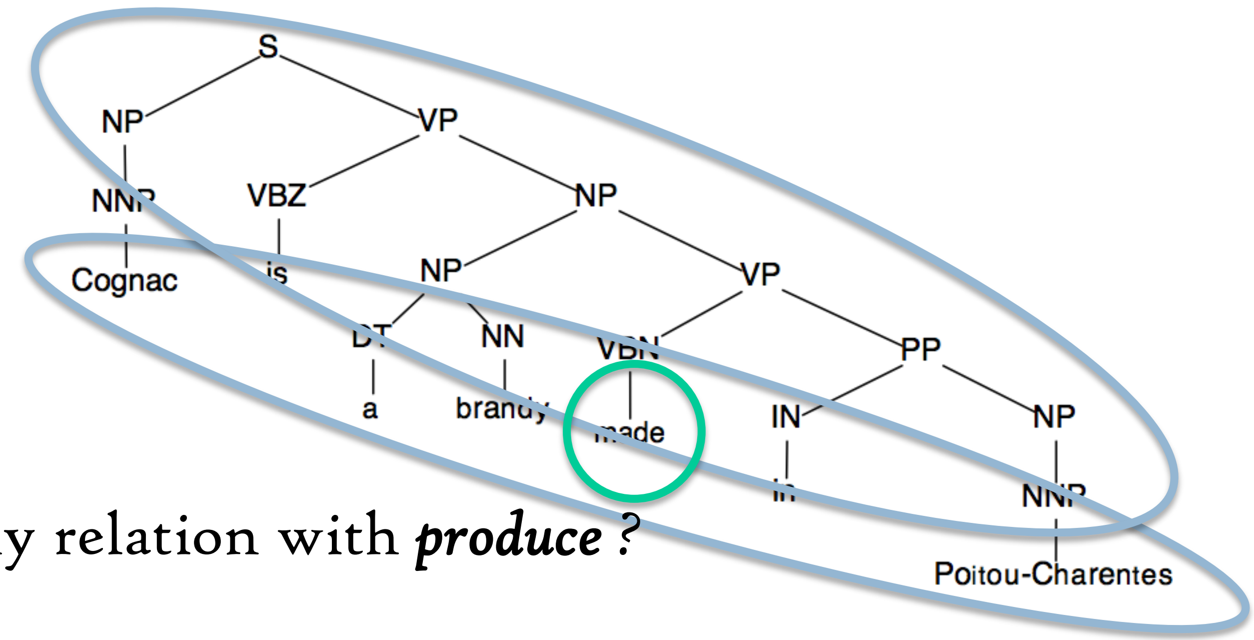
*What **French province is Cognac produced in ?***

*The grapes which **produce** the **Cognac** grow in the **French province** …*

***Cognac is** a brandy <u>made</u> **in** Poitou-Charentes .*

Linguistic inferences: e.g. QA

Syntactic and Semantic Types constraint the linguistic information, that contributes to a variety of crucial inferences at the:

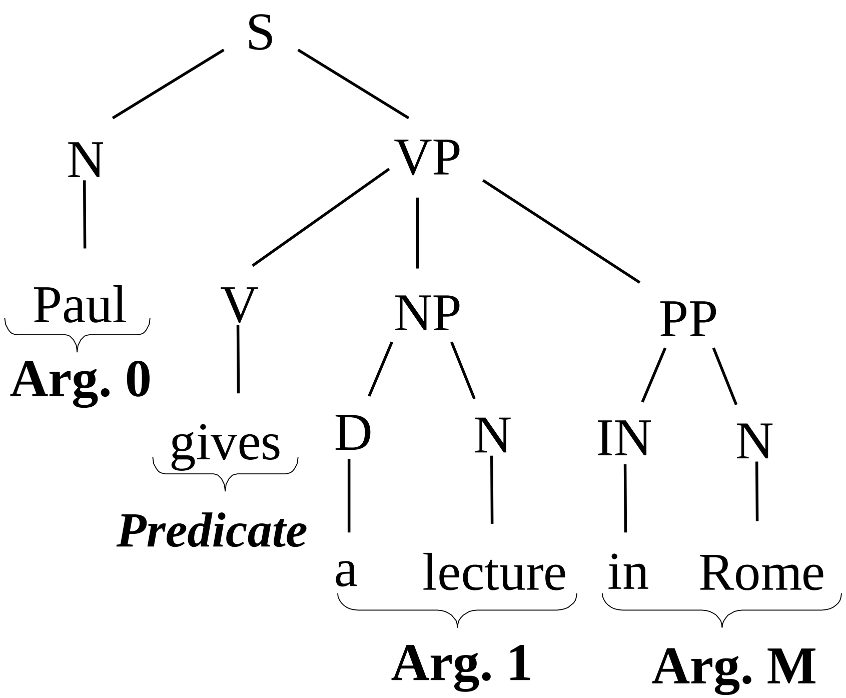


Is there any relation with ***produce*** ?

- Lexical level (e.g. sinonimy recognition)
- Syntactic level (e.g. tree matching for syntactic disambiguation)
- Semantic level (e.g. predicate recognition)

Predicate and Arguments

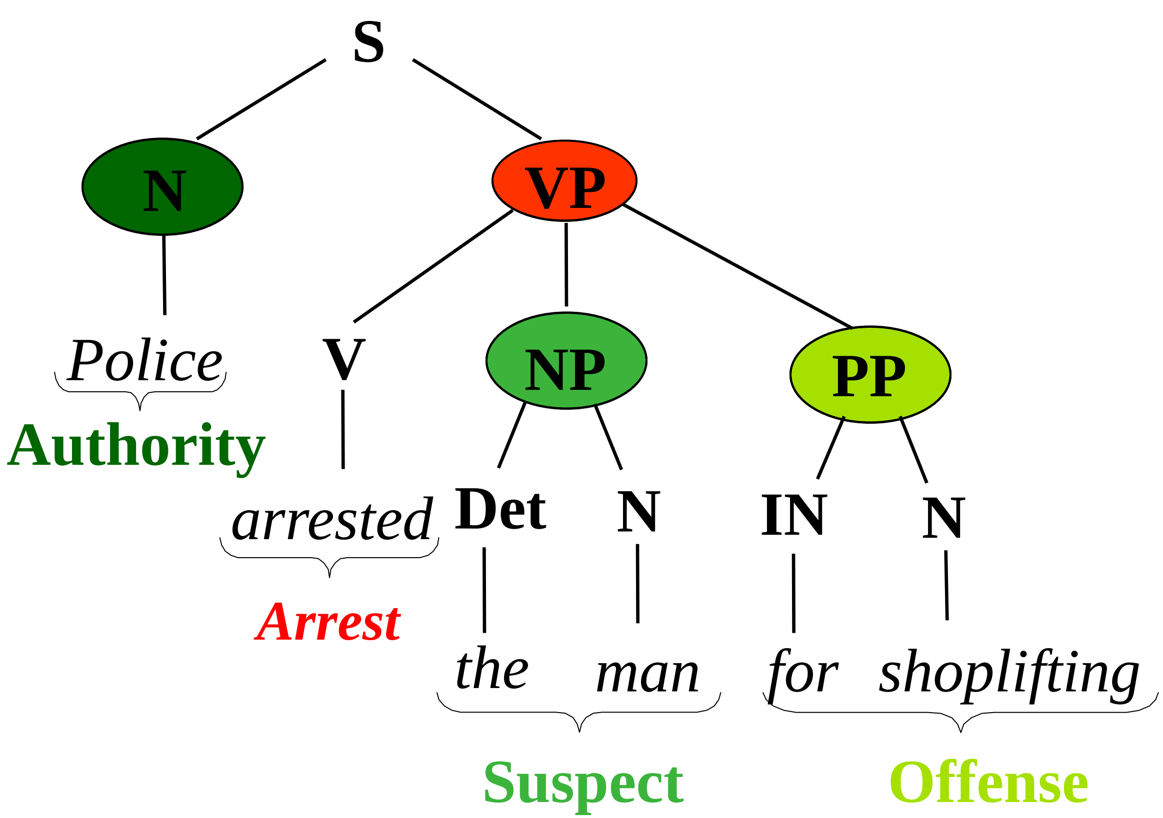
- The syntax-semantic mapping



- Different semantic annotations (e.g. PropBank vs. FrameNet)

Linking syntax to semantics

- *Police arrested the man for shoplifting*



Frame Semantics

Frame: KILLING		
A KILLER or CAUSE causes the death of the VICTIM.		
Frame Elements	KILLER	**John** drowned Martha.
	VICTIM	John drowned **Martha**.
	MEANS	The flood exterminated the rats **by cutting off access to food**.
	CAUSE	**The rockslide** killed nearly half of the climbers.
	INSTRUMENT	It's difficult to suicide **with only a pocketknife**.
Predicates	annihilate.v, annihilation.n, asphyxiate.v,assassin.n, assassinate.v, assassination.n, behead.v, beheading.n, blood-bath.n, butcher.v, butchery.n, carnage.n, crucifixion.n, crucify.v, deadly.a, decapitate.v, decapitation.n, destroy.v, dispatch.v, drown.v, eliminate.v, euthanasia.n, euthanize.v, …	

Semantics in NLP: Resources

- Lexicalized Models
  - Propbank
  - NomBank
- Framenet
  - Inspired by frame semantics
  - Frames are lexicalized prototoypes for real -world situations
  - Participants are called frame elements (roles)

Generative vs. Discriminative Learning in NLP

- **Generative models** (e.g. HMMs) require
  - The design of a model of *visible* and *hidden* variables
  - The definition of *laws of association* between hidden and visible variables
  - *Robust estimation methods* from the available samples
- Limitations:
  - Lack of precise generative models for language phenomena
  - Data sparseness: most of the language phenomena are simply too rare for robust estimation even in large samples

Generative vs. Discriminative Learning

- **Discriminative models** are not tight to any model (i.e. specific association among the problem variables).
- They learn to discriminate negative from positive evidence without building an explicit model of the target property
- They derive useful evidence from training data only through observed individual features by optimizing some function of the recognition task (e.g. error)
- Examples of discriminative models are the perceptrons (i.e. linear classifiers)

Linear Classifiers (1)

An hyperplane has equation :

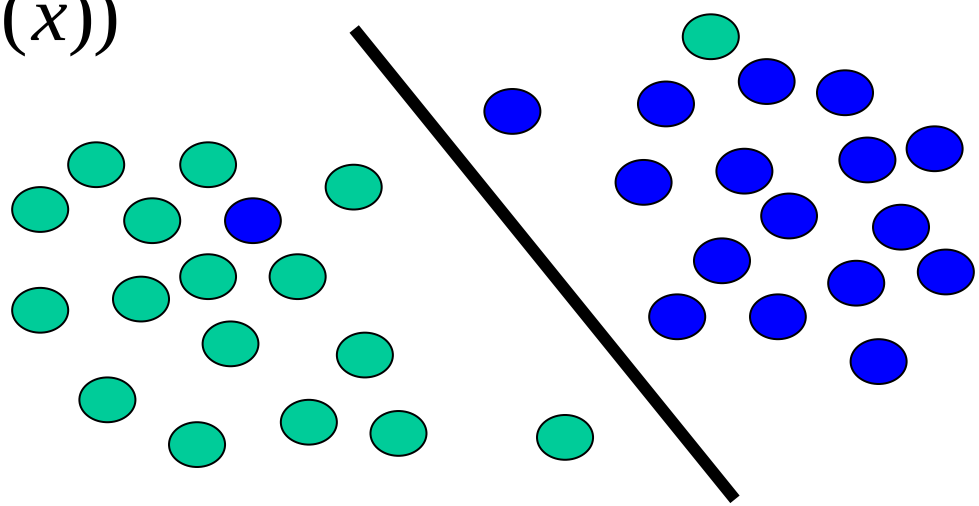
$$f(\vec{x}) = \vec{x} \cdot \vec{w} + b, \quad \vec{x}, \vec{w} \in \Re^n, b \in \Re$$

$\vec{x}$ is the vector of the instance to be classified

$\vec{w}$ is the hyperplane gradient

Classification function:

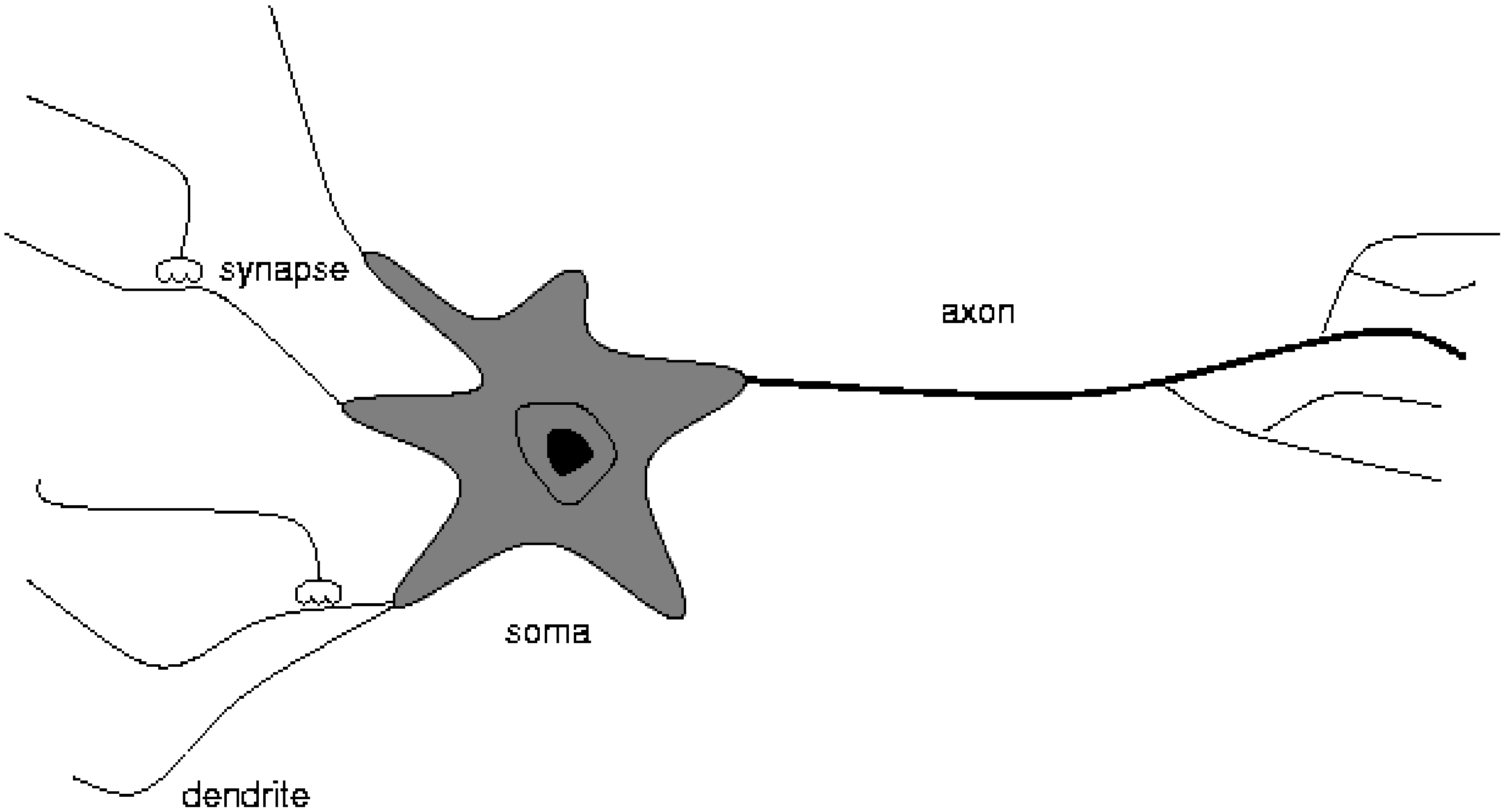
$$h(x) = \text{sign}(f(x))$$



Linear Classifiers (2)

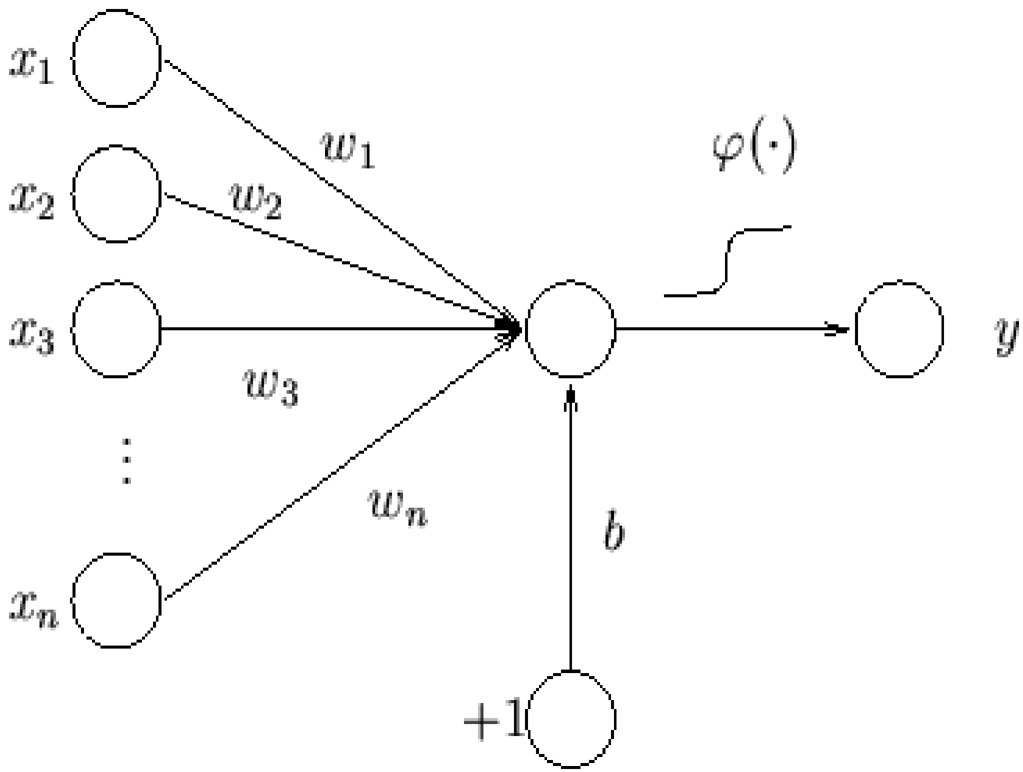
- Computationally simple.
- Basic idea: select an hypothesis that makes no mistake over training-set.
- The separating function is equivalent to a neural net with just one neuron (perceptron)

A neuron



Perceptron

$$\varphi(\vec{x}) = \operatorname{sgn}\left(\sum_{i=1..n} w_i \times x_i + b\right)$$



Duality

- The decision function of linear classifiers can be written as follows:

$$h(x) = \mathrm{sgn}(\vec{w} \cdot \vec{x} + b) = \mathrm{sgn}(\sum_{j=1..\ell} \alpha_j y_j \vec{x}_j \cdot \vec{x} + b) =$$

$$\mathrm{sgn}(\sum_{i=1..\ell} \alpha_j y_j (\vec{x}_j \cdot \vec{x}) + b)$$

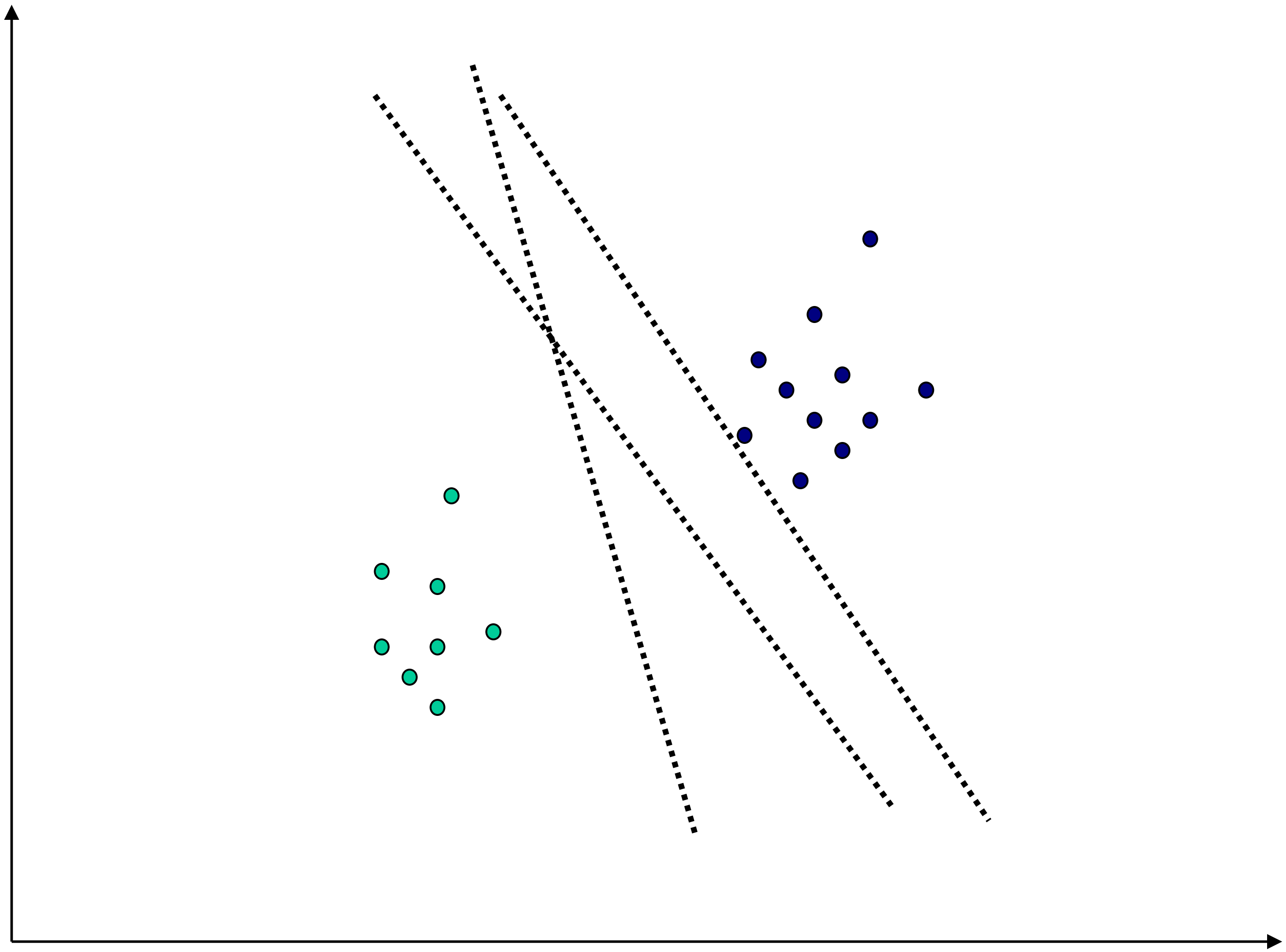
- as well the adjustment function

$$\text{if } y_i(\sum_{j=1..\ell} \alpha_j y_j \vec{x}_j \cdot \vec{x}_i + b) \le 0 \quad \text{then } \alpha_i = \alpha_i + \eta$$

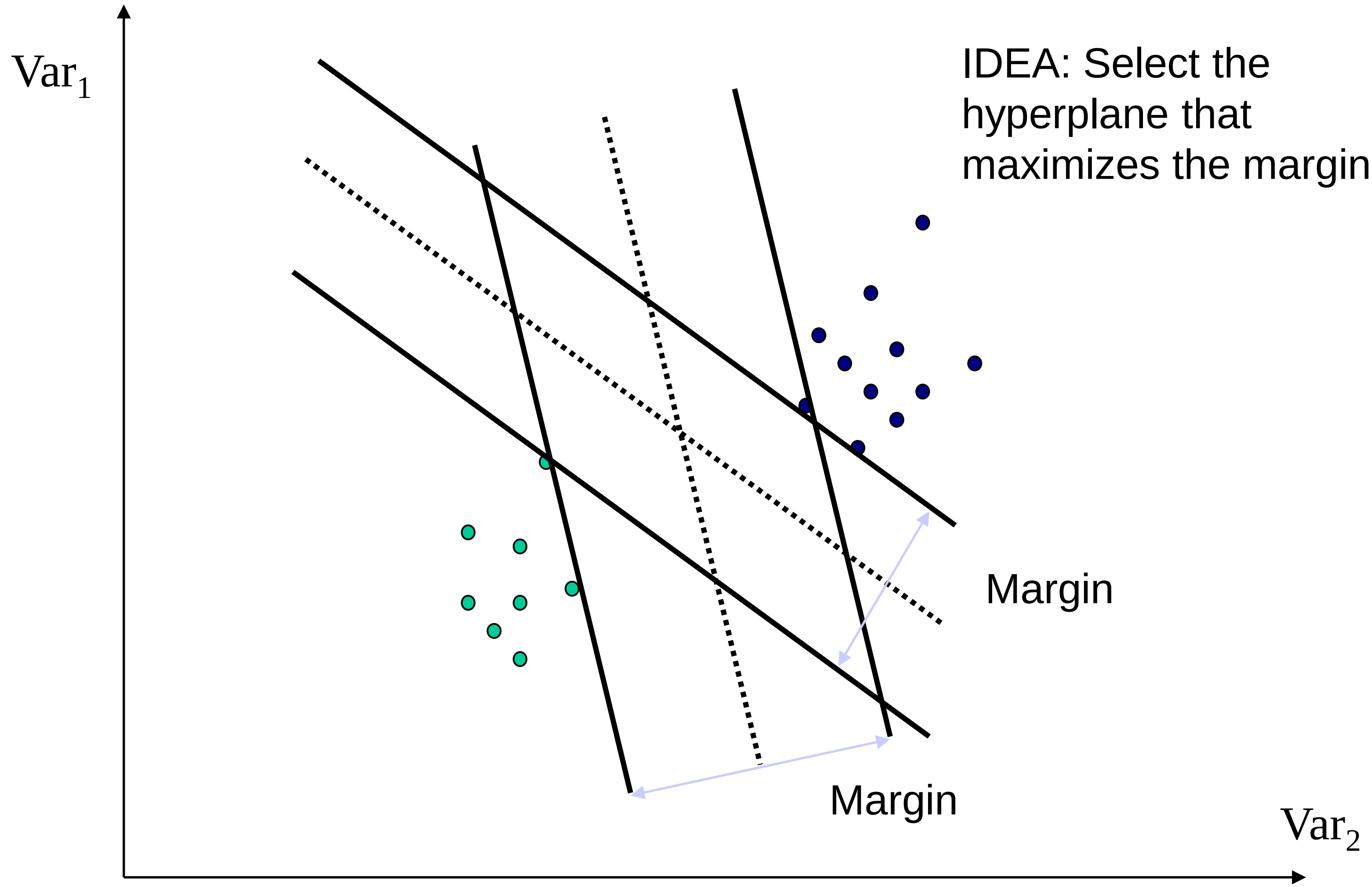
- The learning rate η impacts only in the re-scaling of the hyperplanes, and does not influence the algorithm ($\eta = 1$).

➔ Training data only appear in the scalar products!!

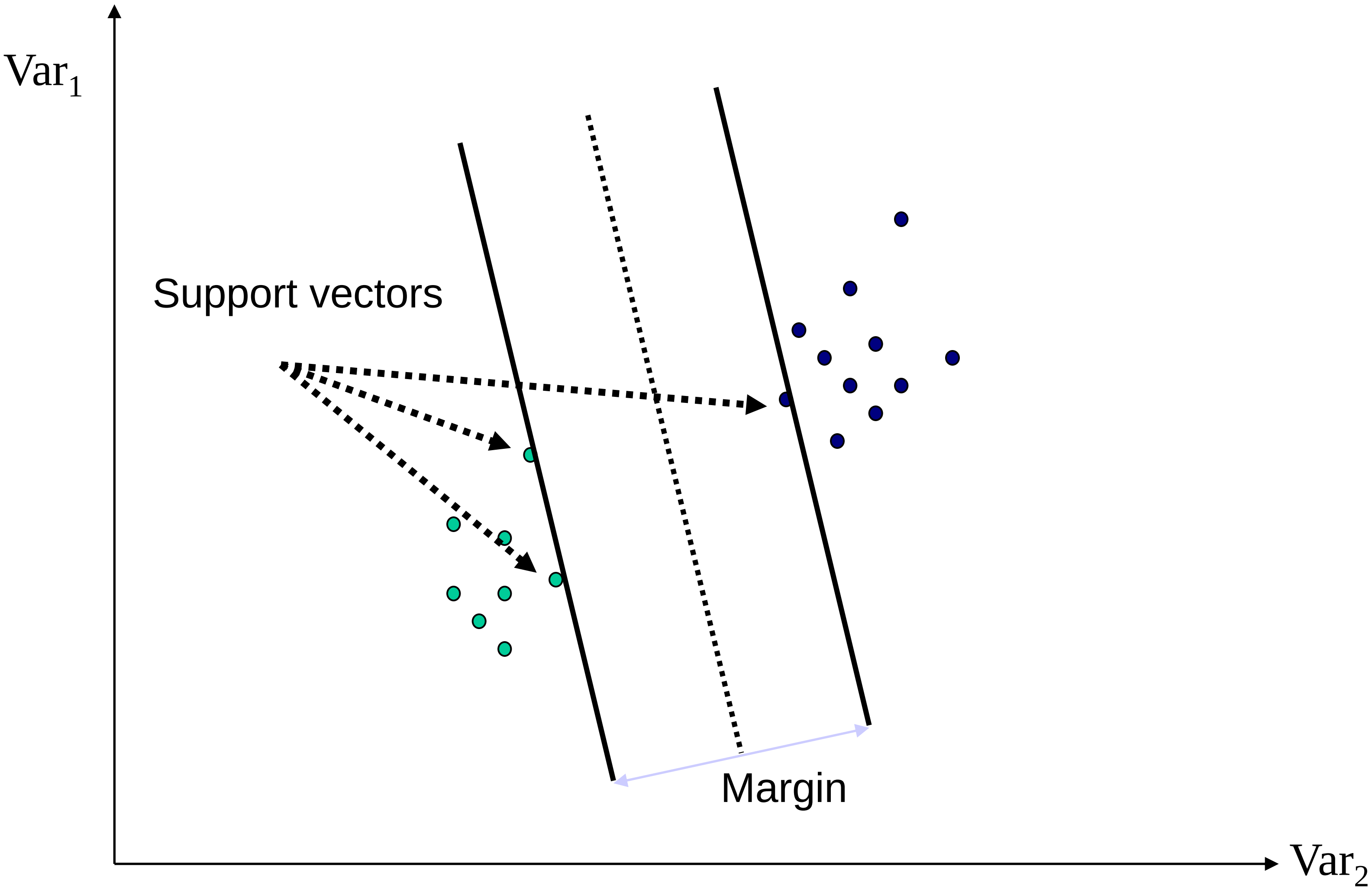
Which hyperplane?



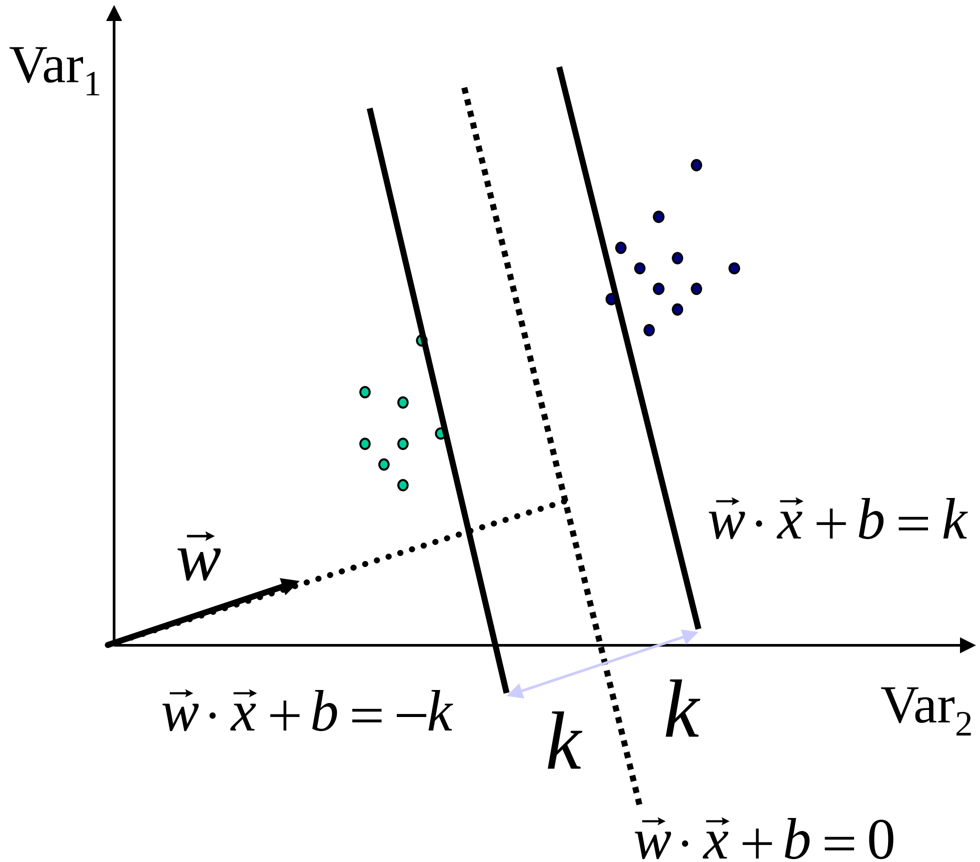
Maximum Margin Hyperplanes



Support Vectors



How to get the maximum margin?



The geometric margin is:

$$\frac{2|k|}{\|w\|}$$

Optimization problem

$$MAX\ \frac{2|k|}{\|\vec{w}\|}$$

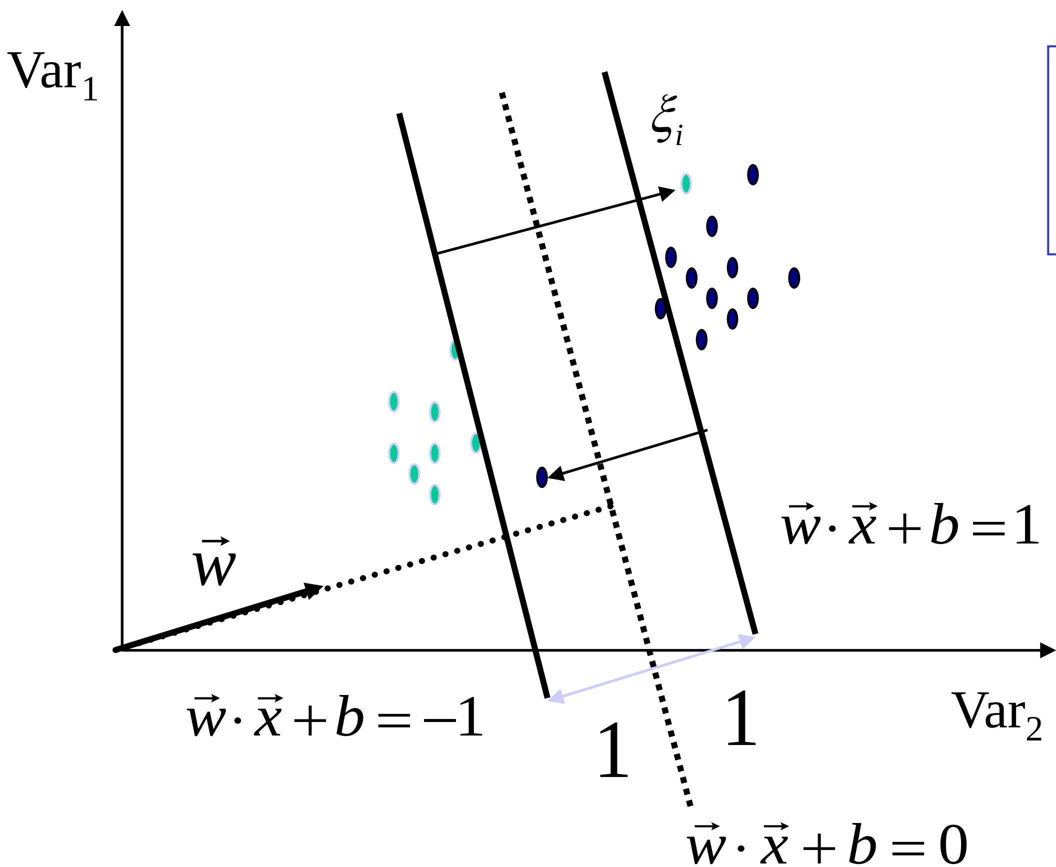
$\vec{w}\cdot\vec{x}+b\geq+k$, if $\vec{x}$ is a positive ex.

$\vec{w}\cdot\vec{x}+b\leq-k$, se $\vec{x}$ is a negativ ex.

The optimization problem

- The optimal hyperplane satyisfies:
  - Minimize $\tau(\vec{w}) = \frac{1}{2}\|\vec{w}\|^2$
  - Under: $y_i\,((\vec{w}\cdot\vec{x}_i) + b) \geq 1, i = 1,\ldots,l$
- The dual problem is simpler

Soft Margin SVMs



New constraints:

$$\boxed{\begin{array}{l} y_i(\vec{w}\cdot\vec{x}_i+b)\geq 1-\xi_i \quad \forall\vec{x}_i \\ \xi_i\geq 0 \end{array}}$$

Objective function:

$$\boxed{\min \frac{1}{2}\|\vec{w}\|^2+C\sum_i \xi_i}$$

C is the *trade-off* between margin and errors

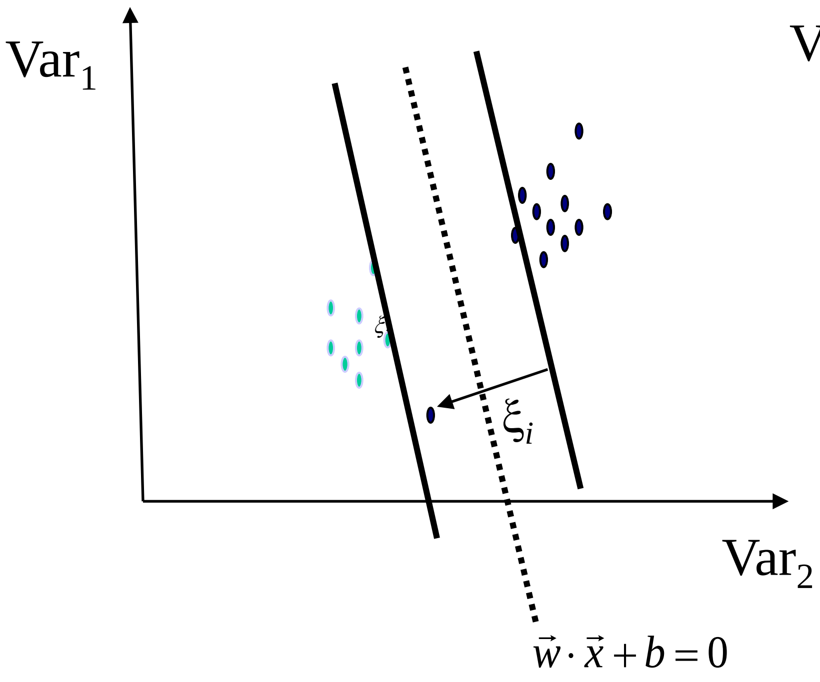
Dual optimization problem

$$\sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y_i y_j \alpha_i \alpha_j (\vec{x_i} \cdot \vec{x_j} + \frac{1}{C} \delta_{ij})$$

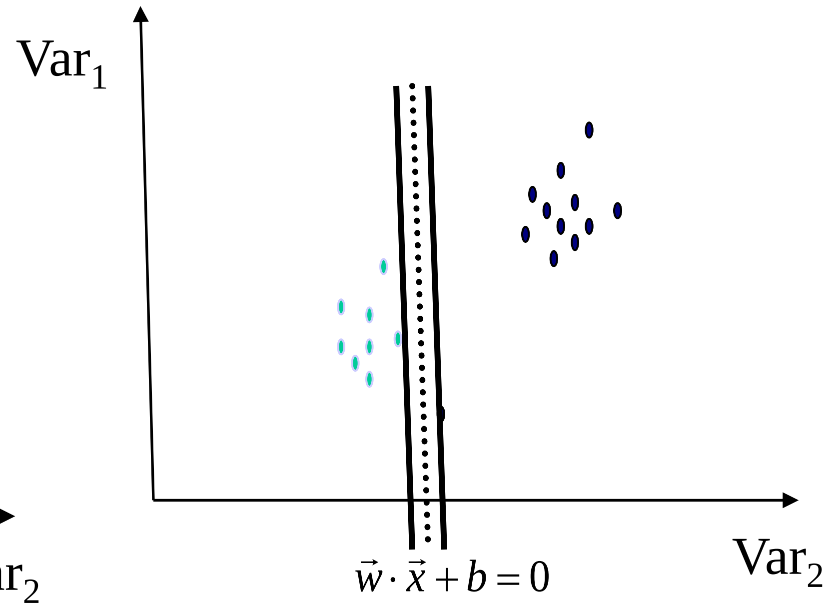
$$\alpha_i \geq 0, \quad \forall i = 1, .., m$$

$$\sum_{i=1}^{m} y_i \alpha_i = 0$$

Robustness: *Soft* vs *Hard Margin* SVMs



Soft Margin SVM



Hard Margin SVM

Soft vs Hard Margin SVMs

- A *Soft-Margin* SVM has always a solution
- A *Soft-Margin* SVM is more robust wrt *odd* training examples
  - *Insufficient Vocabularies*
  - *High ambiguity of linguistic features*
- An *Hard-Margin* SVM requires no parameter

Kernel Functions in SVM Learning

The Perceptron Dual Algorithm and Kernels

- We can rewrite the deecision function as follows:

$$h(x) = \mathrm{sgn}(\vec{w} \cdot \phi(\vec{x}) + b) = \mathrm{sgn}(\sum_{j=1..\ell} \alpha_j y_j \phi(\vec{x}_j) \cdot \phi(\vec{x}) + b) =$$

$$= \mathrm{sgn}(\sum_{i=1..\ell} \alpha_j y_j k(\vec{x}_j, \vec{x}) + b)$$

- The updating function (in the perceptron) becomes:

$$\text{if } \; y_i(\sum_{j=1..\ell} \alpha_j y_j \phi(\vec{x}_j) \cdot \phi(\vec{x}_i) + b) = y_i(\sum_{j=1..\ell} \alpha_j y_j k(\vec{x}_j, \vec{x}_i) + b) \leq 0$$

$$\text{then } \; \alpha_i = \alpha_i + \eta$$

Classification Function: the dual form

$$\operatorname{sgn}(\vec{w}\cdot\vec{x}+b)=\operatorname{sgn}\left(\sum_{j=1..\ell}\alpha_j y_j \vec{x}_j\cdot\vec{x}+b\right)$$

- Note that input data only appear in the inner product
- The matrix $G=\left(\left\langle \mathbf{x}_i\cdot\mathbf{x}_j\right\rangle\right)_{i,j=1}^{l}$ is called *Gram matrix*

Kernel functions: definition

Def. 2.26 *A kernel is a function k, such that* $\forall\ \vec{x}, \vec{z} \in X$

$$k(\vec{x}, \vec{z}) = \phi(\vec{x}) \cdot \phi(\vec{z})$$

where ϕ *is a mapping from* X *to an (inner product) feature space.*

- Kernels express implicit mappings such as:

$$\vec{x} \in \Re^n, \qquad \vec{\phi}(\vec{x}) = (\phi_1(\vec{x}), \phi_2(\vec{x}), \ldots, \phi_m(\vec{x})) \in \Re^m$$

Valid Kernels (1)

Def. B.11 *Eigen Values*
Given a matrix $\boldsymbol{A} \in \mathbb{R}^m \times \mathbb{R}^n$*, an egeinvalue* λ *and an egeinvector* $\vec{x} \in \mathbb{R}^n - \{\vec{0}\}$ *are such that*

$$\mathbf{A}\vec{x} = \lambda\vec{x}$$

Def. B.12 *Symmetric Matrix*
A square matrix $\boldsymbol{A} \in \mathbb{R}^n \times \mathbb{R}^n$ *is symmetric iff* $\boldsymbol{A}_{ij} = \boldsymbol{A}_{ji}$ *for* $i \neq j$ $i = 1, .., m$ *and* $j = 1, .., n$*, i.e. iff* $\boldsymbol{A} = \boldsymbol{A}'$.

Def. B.13 *Positive (Semi-) definite Matrix*
A square matrix $\boldsymbol{A} \in \mathbb{R}^n \times \mathbb{R}^n$ *is said to be positive (semi-) definite if its eigenvalues are all positive (non-negative).*

Valid kernels (2)

Proposition 2.27 *(Mercer's conditions)*
Let X be a finite input space with $K(\vec{x}, \vec{z})$ a symmetric function on X. Then $K(\vec{x}, \vec{z})$ is a kernel function if and only if the matrix

$$k(\vec{x}, \vec{z}) = \phi(\vec{x}) \cdot \phi(\vec{z})$$

is positive semi-definite (has non-negative eigenvalues).

- Main idea: IF *the Gram matrix is semidefinite positive* THEN *the mapping ϕ that realizes the kernel function exists. This constitutes a space F where separability is better modelled.*

Polynomial kernel and the conjunction of features

- The initial vectors can be mapped into a higher dimensional space (c=1)

$$\Phi(< x_1, x_2 >) \rightarrow (x_1^2, x_2^2, \sqrt{2}x_1x_2, \sqrt{2}x_1, \sqrt{2}x_2, 1)$$

- More expressive, as (x_1x_2) encodes

stock+market vs. *downtown+market*

features

- We can smartly compute the scalar product as $)=$

$$\Phi(\vec{x})\times\Phi(\vec{z}) = (x_1^2, x_2^2, \sqrt{2}x_1x_2, \sqrt{2}x_1, \sqrt{2}x_2, 1)\times(z_1^2, z_2^2, \sqrt{2}z_1z_2, \sqrt{2}z_1, \sqrt{2}z_2, 1)$$
$$= x_1^2z_1^2 + x_2^2z_2^2 + 2x_1x_2z_1z_2 + 2x_1z_1 + 2x_2z_2 + 1 =$$
$$= (x_1z_1 + x_2z_2 + 1)^2 = \boxed{(\vec{x}\times\vec{z} + 1)^2 = K_{p2}\ (\vec{x}, \vec{z})}$$

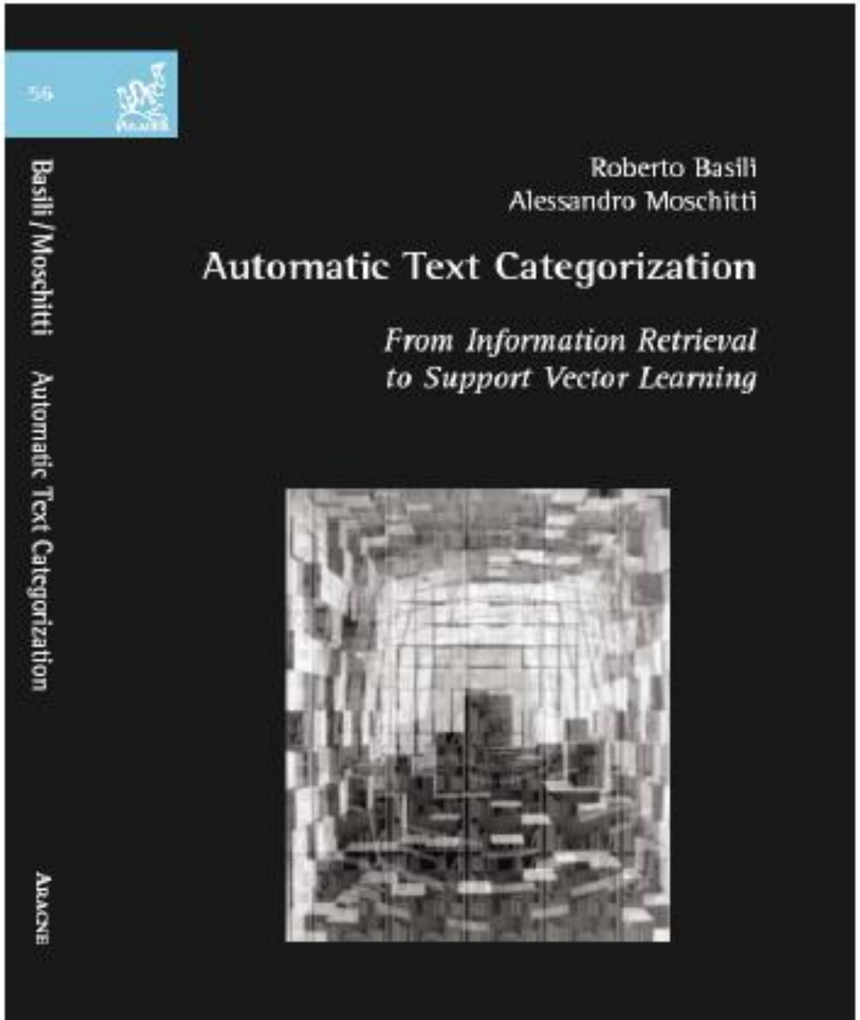
NLP-oriented kernels

- Semantic kernels
  - Latent Semantic Kernels (Cristianini et al., 2003)
  - KB kernels, such as (Basili et al., 2005)
- String or sequence kernels
  - (Lodhi et al. 2001)
- Tree kernels (Collins & Duffy, 2001)
  - Partial Tree kernels (Moschitti, ECML 2005)
  - … see later slides

References

- Basili, R., A. Moschitti *Automatic Text Categorization: From Information Retrieval to Support Vector Learning* , Aracne Editrice, Informatica, ISBN: 88-548-0292-1, 2005
- *A tutorial on Support Vector Machines for Pattern Recognition* (*C.J.Burges*)
  - **URL:** http://www.umiacs.umd.edu/~joseph/support-vector-machines4.pdf
- *The Vapnik-Chervonenkis Dimension and the Learning Capability of Neural Nets (E.D: Sontag)*
  - **URL:** http://www.math.rutgers.edu/~sontag/FTP_DIR/vc-expo.pdf
- Computational Learning Theory
  (Sally A Goldman Washington University St. Louis Missouri)
  - http://www.learningtheory.org/articles/COLTSurveyArticle.ps
- *AN INTRODUCTION TO SUPPORT VECTOR MACHINES* (*and other kernel-based learning methods*), N. Cristianini and J. Shawe-Taylor Cambridge University Press.
- *The Nature of Statistical Learning Theory*, V. N. Vapnik - Springer Verlag (December, 1999)

An introductory book on SVMs, Kernel methods and Text Categorization



Overview

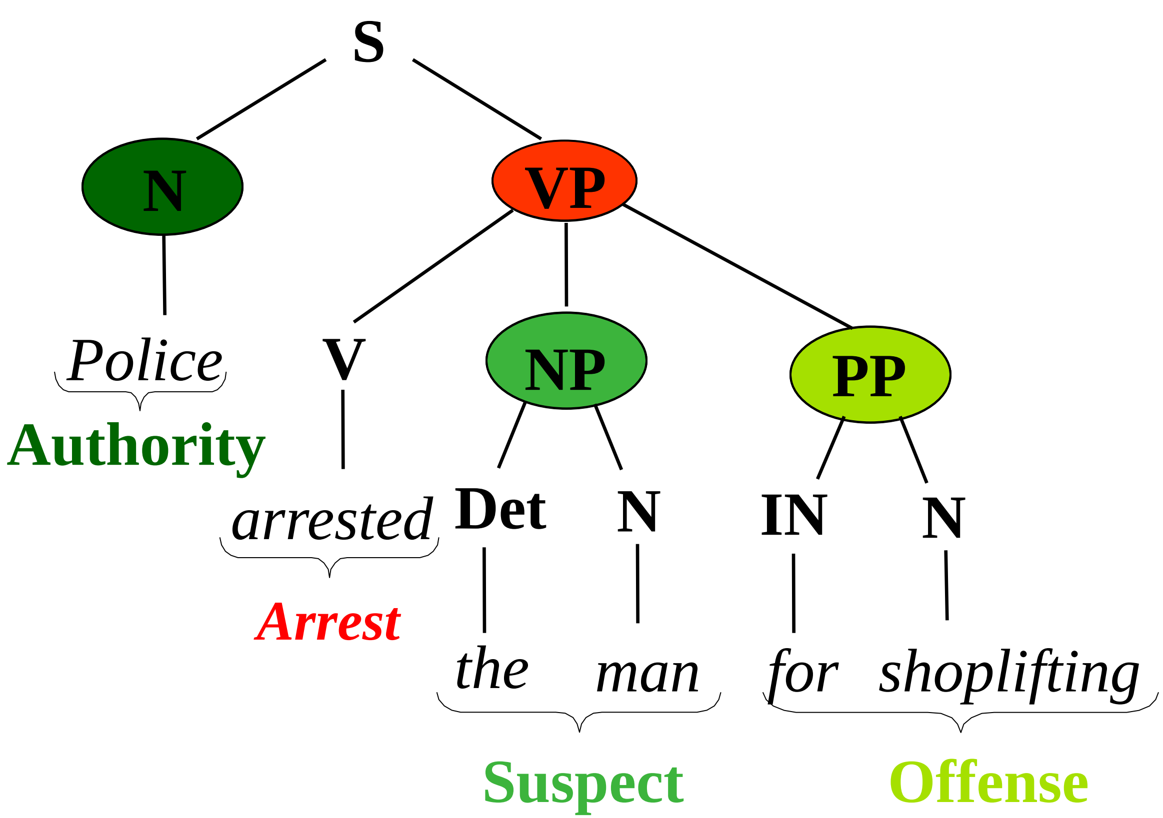
- Session I: Machine Learning for NLP
  - Support Vector Machines for NLP
  - Kernels for HLTs
    - Sequence and Tree Kernels
- Session II: Semantic Role Labeling
  - Standard Linguistic Features for SRL
  - The role of Syntax
  - Future Work: Semantic Tree Kernels (SPTK)

Semantic Role Labeling @ UTV

- An important application of SVM is Semantic Role labeling wrt Propbank or Framenet
- In the UTV system, a cascade of classification steps is applied:
  - Predicate detection
  - Boundary recognition
  - Argument categorization (Local models)
  - Reranking (Joint models)
- Input: a sentence and its parse trees

Linking syntax to semantics

- *Police arrested the man for shoplifting*



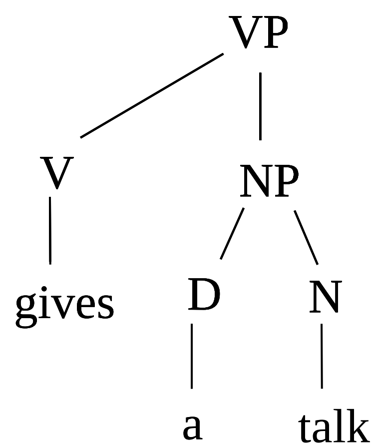
Motivations

- Modeling syntax in Natural Language learning task is complex, e.g.
  - Semantic role relations within predicate argument structures and
  - Question Classification
- Tree kernels are natural way to exploit syntactic information from sentence parse trees
  - useful to engineer novel and complex features.
- How do different tree kernels impact on different parsing paradigms and different tasks?
- Are they efficient in practical applications?

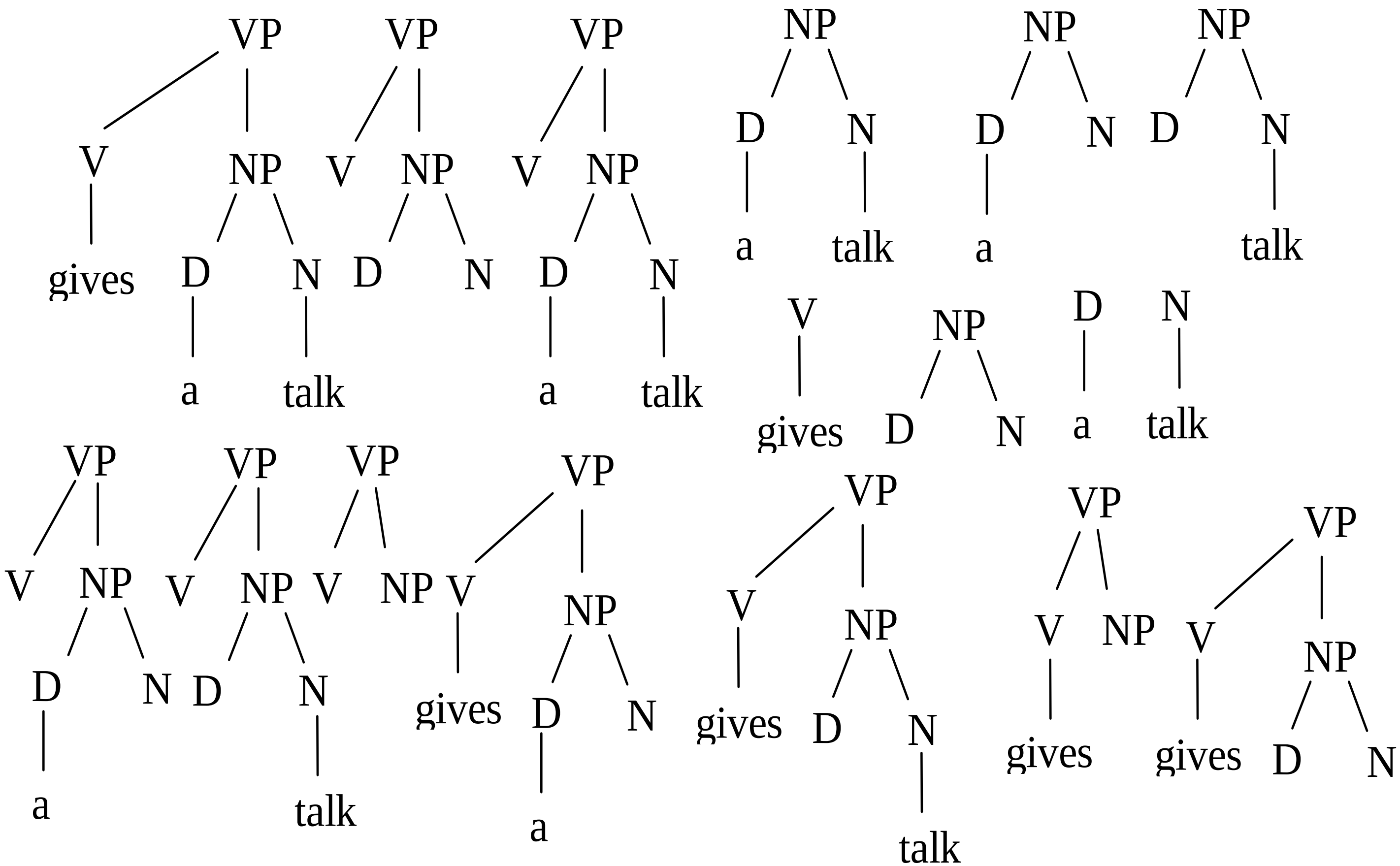
Tree kernels: Outline

- Nature and Definition of Tree kernels
- Different Types of Tree kernels
  - Subset (SST) Tree kernel
  - The Partial Tree kernel
- Adopting Tree kernels in SRL
- Extending Tree kernels with lexical similarity, the SPTK kernel

The Collins and Duffy's Tree Kernel (called SST in [Vishwanathan and Smola, 2002])

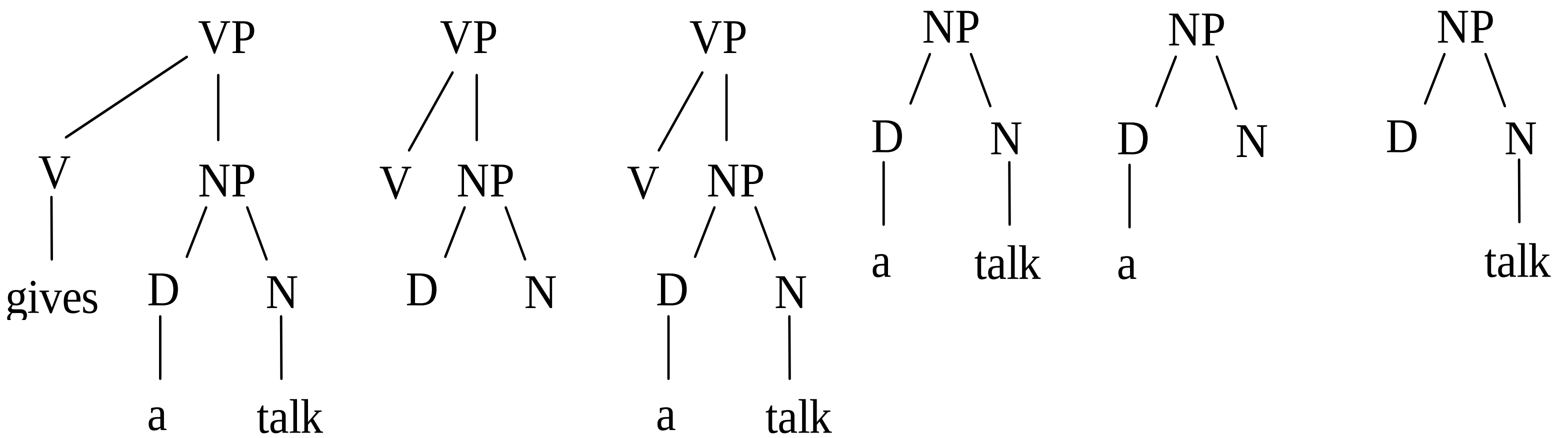


The overall fragment set



Explicit feature space

$$\vec{x} = (\mathbf{0, .., 1, .., 0, .., 1, .., 0, .., 1, .., 0, ., 1, .., 0, .., 1, .., 0, .., 1, .., 0})$$



- $\vec{x}_1 \cdot \vec{x}_2$ counts the number of common substructures

Implicit Representation

$$\vec{x}_1 \cdot \vec{x}_2 = \phi(T_1) \cdot \phi(T_2) = K(T_1, T_2) =$$
$$= \sum_{n_1 \in T_1} \sum_{n_2 \in T_2} \Delta(n_1, n_2)$$

Implicit Representation

$$\vec{x}_1 \cdot \vec{x}_2 = \phi(T_1) \cdot \phi(T_2) = K(T_1, T_2) =$$
$$= \sum_{n_1 \in T_1} \sum_{n_2 \in T_2} \Delta(n_1, n_2)$$

- [Collins and Duffy, ACL 2002] evaluate Δ in O(n²):

$$\Delta(n_1, n_2) = 0, \ \ \textbf{if the productions are different else}$$
$$\Delta(n_1, n_2) = 1, \ \ \textbf{if pre - terminals else}$$
$$\Delta(n_1, n_2) = \prod_{j=1}^{nc(n_1)} (1 + \Delta(ch(n_1, j), ch(n_2, j)))$$

Weighting

- Decay factor

$$\Delta(n_1, n_2) = \lambda, \quad \textbf{if pre - terminalselse}$$

$$\Delta(n_1, n_2) = \lambda \prod_{j=1}^{nc(n_1)} (1 + \Delta(ch(n_1, j), ch(n_2, j)))$$

- Normalization

$$K'(T_1, T_2) = \frac{K(T_1, T_2)}{\sqrt{K(T_1, T_1) \times K(T_2, T_2)}}$$

Partial Tree Kernel

- if the node labels of n_1 and n_2 are different then $\Delta(n_1, n_2) = 0$;

- else

$$\Delta(n_1, n_2) = 1 + \sum_{\vec{J}_1, \vec{J}_2, l(\vec{J}_1) = l(\vec{J}_2)} \prod_{i=1}^{l(\vec{J}_1)} \Delta(c_{n_1}[\vec{J}_{1i}], c_{n_2}[\vec{J}_{2i}])$$

- By adding two decay factors we obtain:

$$\mu\left(\lambda^2 + \sum_{\vec{J}_1, \vec{J}_2, l(\vec{J}_1) = l(\vec{J}_2)} \lambda^{d(\vec{J}_1) + d(\vec{J}_2)} \prod_{i=1}^{l(\vec{J}_1)} \Delta(c_{n_1}[\vec{J}_{1i}], c_{n_2}[\vec{J}_{2i}])\right)$$

SRL Demo

- Kernel-based system for SRL over raw texts …
- … based on the Charniak parser
- Adopts the Propbank standard but has also been applied to Framenet

Semantic Role Labeling Graphical Interface - Mozilla Firefox

File Modifica Visualizza Cronologia Segnalibri Strumenti ?

file:///C:/Docs/RobertoB/LAVORI_IN_CORSO/SemRoleLabeling/tool/getsent2.html

Personalizzazione coll...

Corso IUM ... | JULIE Lab -... | FrameNet | Seman... | Vivavoce - ... | Publications | UIR - Unio... | Workshop ... | Gestione S... | AI*IA 200... | Automatic ... | Errore ca

Kernel-based Semantic Role Labeling

ART

Artificial intelligence Research

Università di *Roma Tor Vergata*

TOR VERGATA

SRL User Interface

Enter a new sentence:

ANALYZE

Select an example sentence:

Run System ○ Show Results ◉

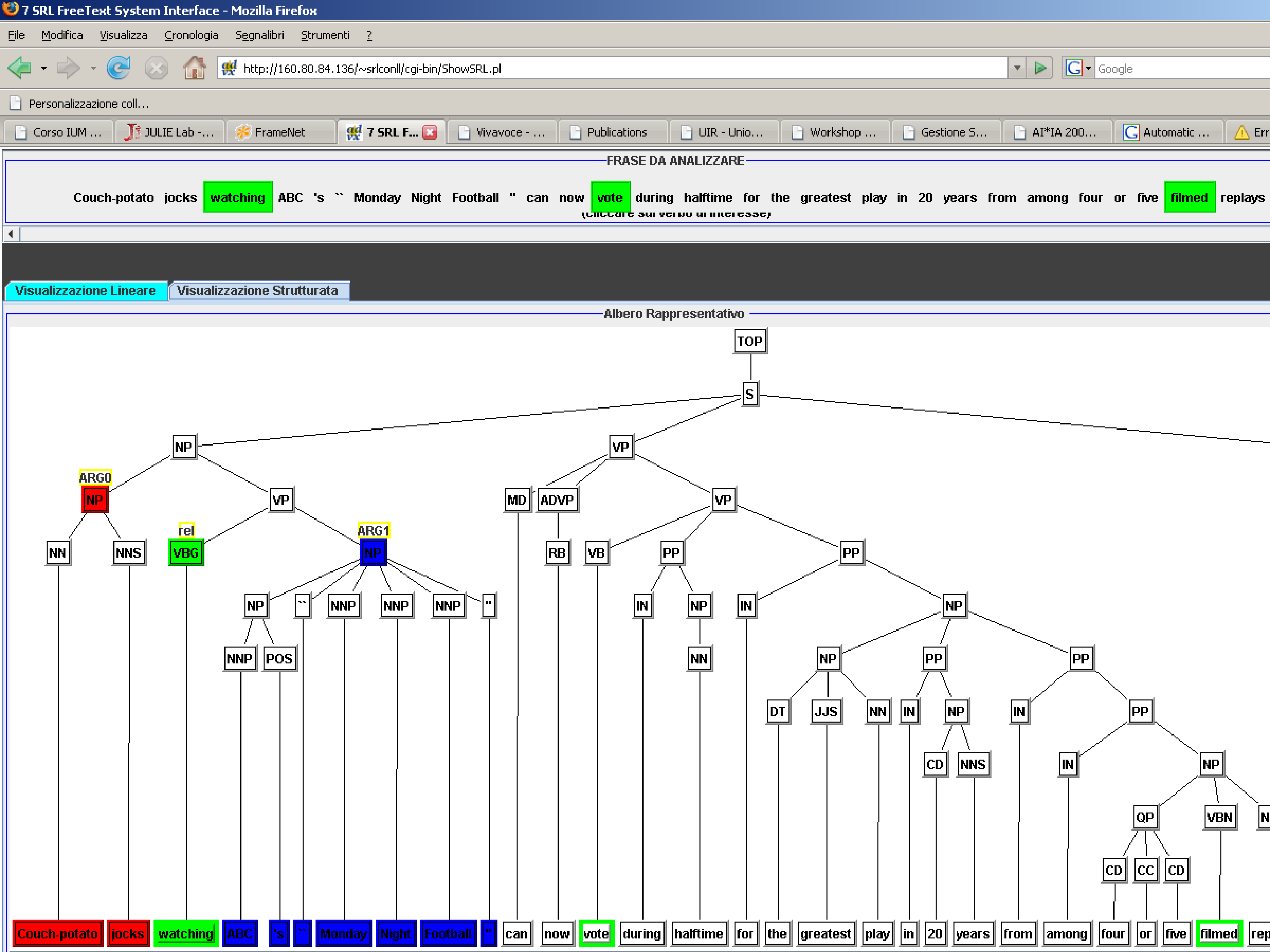
Run System ○ Show Results ◉

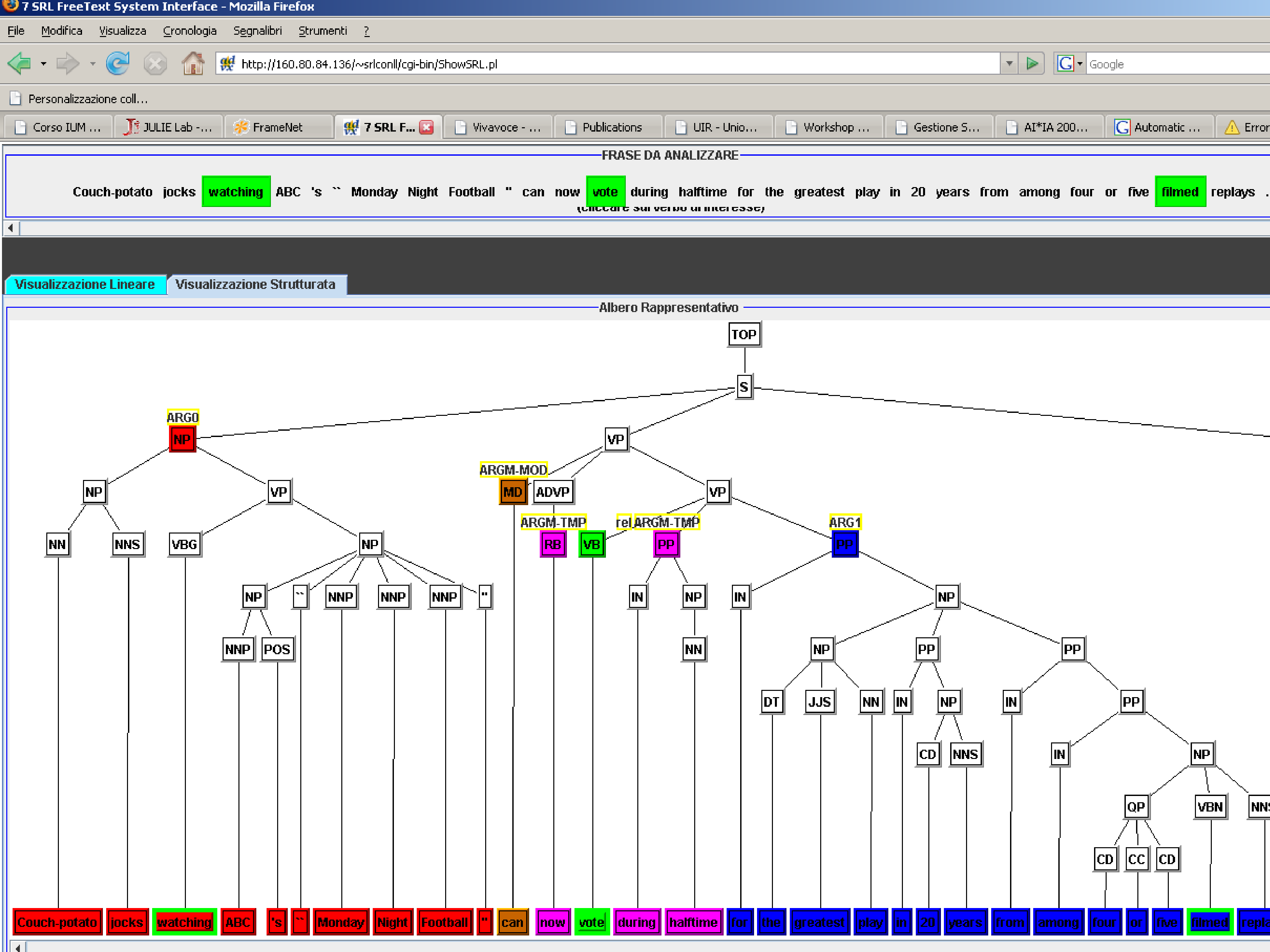
◉ Couch-potato jocks watching ABC's "Monday Night Football" can now vote during halftime for the greatest play in 20 years from among four or five filmed replays.

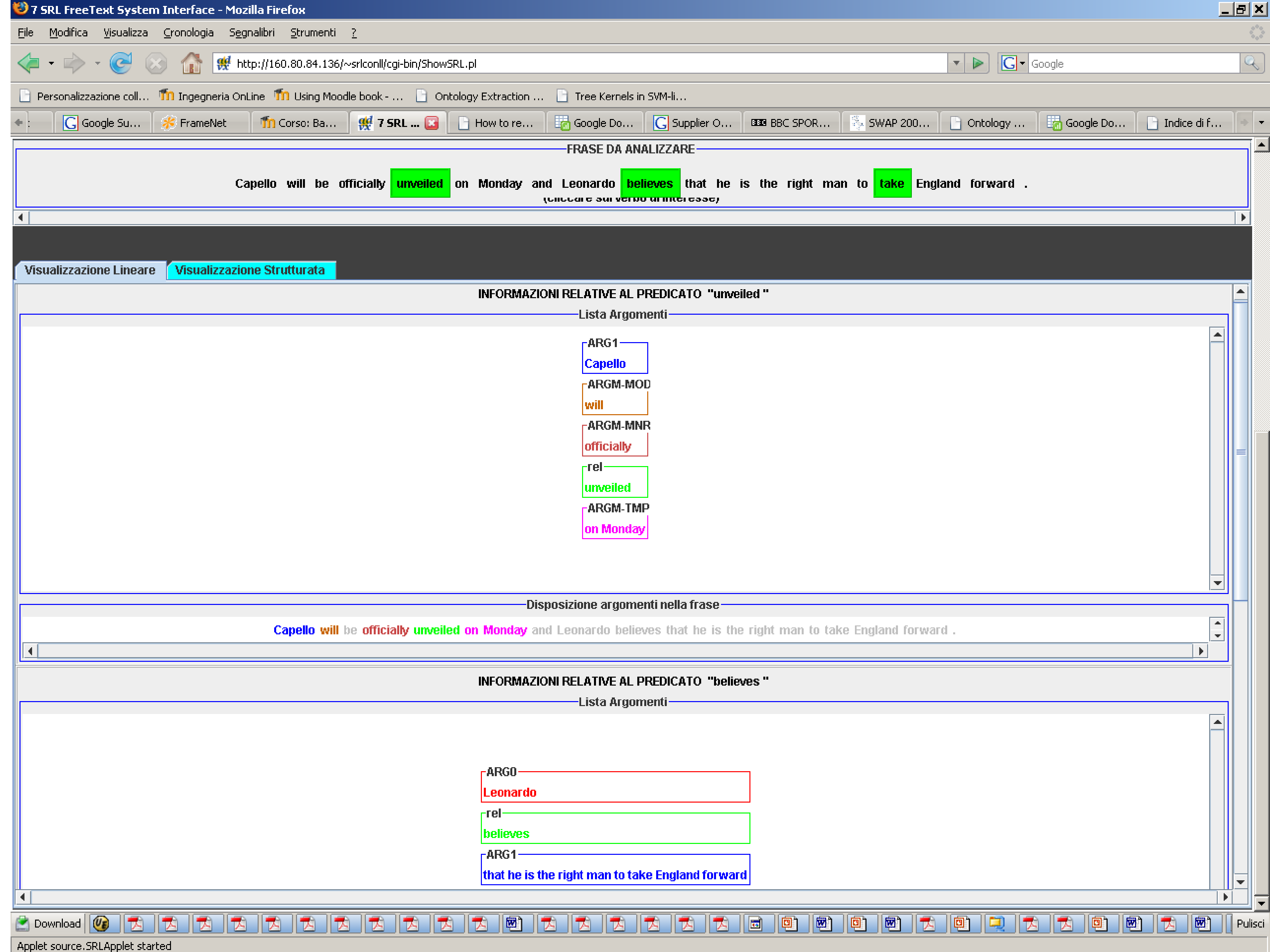
○ During last summer, two thousand trees were burnt by criminals.

○ Mary would like to understand why John betrayed her.

ANALYZE


7 SRL FreeText System Interface - Mozilla Firefox
File Modifica Visualizza Cronologia Segnalibri Strumenti ?
http://160.80.84.136/~srlconll/cgi-bin/ShowSRL.pl
Google
Personalizzazione coll...
Corso IUM ... JULIE Lab -... FrameNet 7 SRL F... Vivavoce - ... Publications UIR - Unio... Workshop ... Gestione S... AI*IA 200... Automatic ... Err
FRASE DA ANALIZZARE
Couch-potato jocks watching ABC 's `` Monday Night Football '' can now vote during halftime for the greatest play in 20 years from among four or five filmed replays
Visualizzazione Lineare
Visualizzazione Strutturata
Albero Rappresentativo
TOP
S
NP
VP
ARG0
NP
VP
MD
ADVP
VP
NN
NNS
rel
VBG
ARG1
NP
RB
VB
PP
PP
NP
``
NNP
NNP
NNP
''
IN
NP
IN
NP
NNP
POS
NN
NP
PP
PP
DT
JJS
NN
IN
NP
IN
PP
CD
NNS
IN
NP
QP
VBN
CD
CC
CD
Couch-potato jocks watching ABC 's `` Monday Night Football '' can now vote during halftime for the greatest play in 20 years from among four or five filmed


7 SRL FreeText System Interface - Mozilla Firefox
File Modifica Visualizza Cronologia Segnalibri Strumenti ?
http://160.80.84.136/~srlconll/cgi-bin/ShowSRL.pl
Google
Personalizzazione coll...
Corso IUM ... JULIE Lab -... FrameNet 7 SRL F... Vivavoce - ... Publications UIR - Unio... Workshop ... Gestione S... AI*IA 200... Automatic ... Error
FRASE DA ANALIZZARE
Couch-potato jocks watching ABC 's `` Monday Night Football '' can now vote during halftime for the greatest play in 20 years from among four or five filmed replays .
Visualizzazione Lineare
Visualizzazione Strutturata
Albero Rappresentativo
TOP
S
ARG0
NP
VP
ARGM-MOD
MD
ADVP
ARGM-TMP
RB
VB
rel
ARGM-TMP
PP
ARG1
PP
NP NN NNS VBG NNP POS NNP NNP NNP IN NN DT JJS NN IN CD NNS QP VBN CC
Couch-potato jocks watching ABC 's `` Monday Night Football '' can now vote during halftime for the greatest play in 20 years from among four or five filmed


7 SRL FreeText System Interface - Mozilla Firefox
File Modifica Visualizza Cronologia Segnalibri Strumenti ?
http://160.80.84.136/~srlconll/cgi-bin/ShowSRL.pl
Personalizzazione coll... Ingegneria OnLine Using Moodle book - ... Ontology Extraction ... Tree Kernels in SVM-li...
Google Su... FrameNet Corso: Ba... 7 SRL ... How to re... Google Do... Supplier O... BBC SPOR... SWAP 200... Ontology ... Google Do... Indice di f...
FRASE DA ANALIZZARE
Capello will be officially unveiled on Monday and Leonardo believes that he is the right man to take England forward .
Visualizzazione Lineare
Visualizzazione Strutturata
INFORMAZIONI RELATIVE AL PREDICATO "unveiled "
Lista Argomenti
ARG1
Capello
ARGM-MOD
will
ARGM-MNR
officially
rel
unveiled
ARGM-TMP
on Monday
Disposizione argomenti nella frase
Capello will be officially unveiled on Monday and Leonardo believes that he is the right man to take England forward .
INFORMAZIONI RELATIVE AL PREDICATO "believes "
Lista Argomenti
ARG0
Leonardo
rel
believes
ARG1
that he is the right man to take England forward
Download
Pulisci
Applet source.SRLApplet started

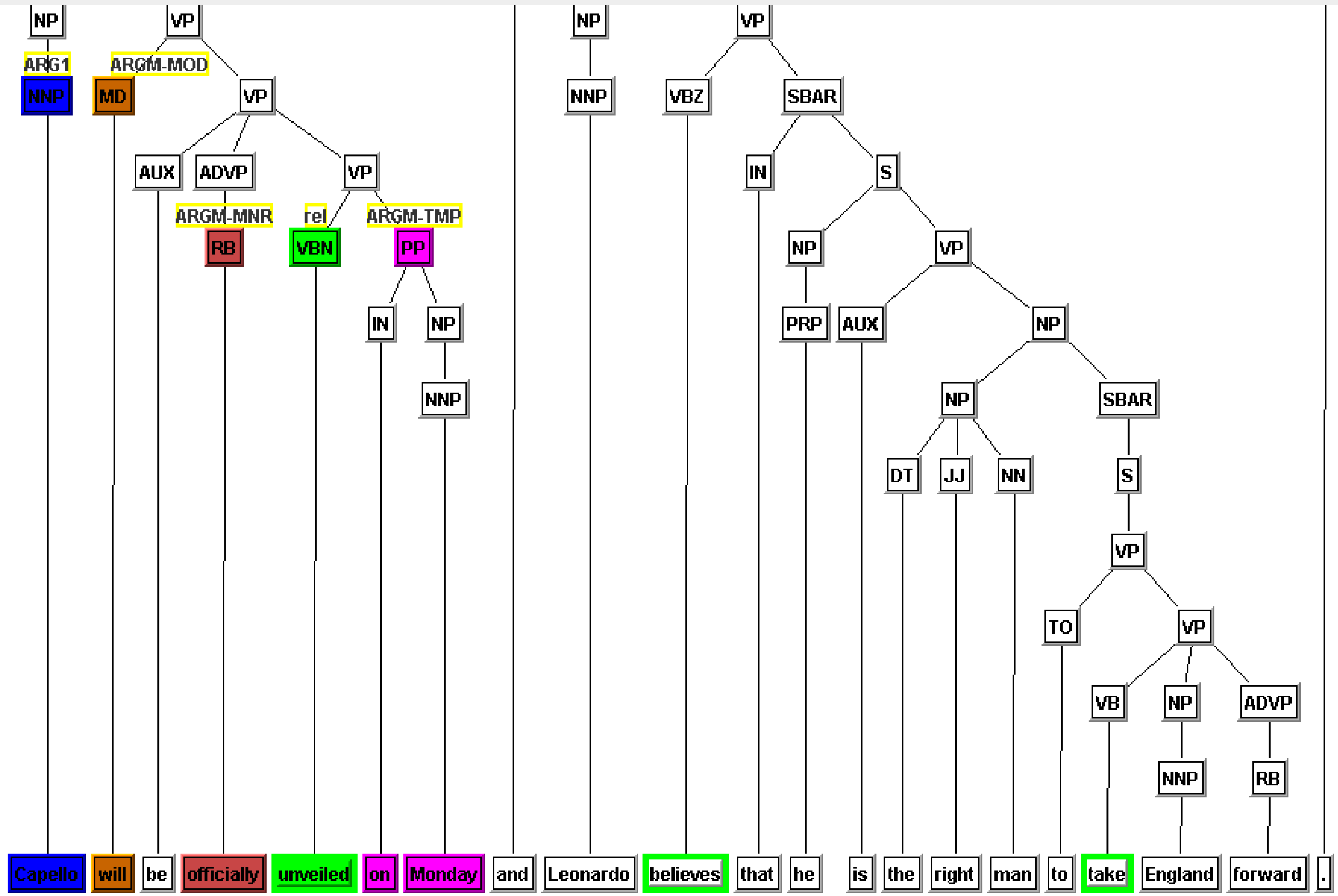
FRASE DA ANALIZZARE

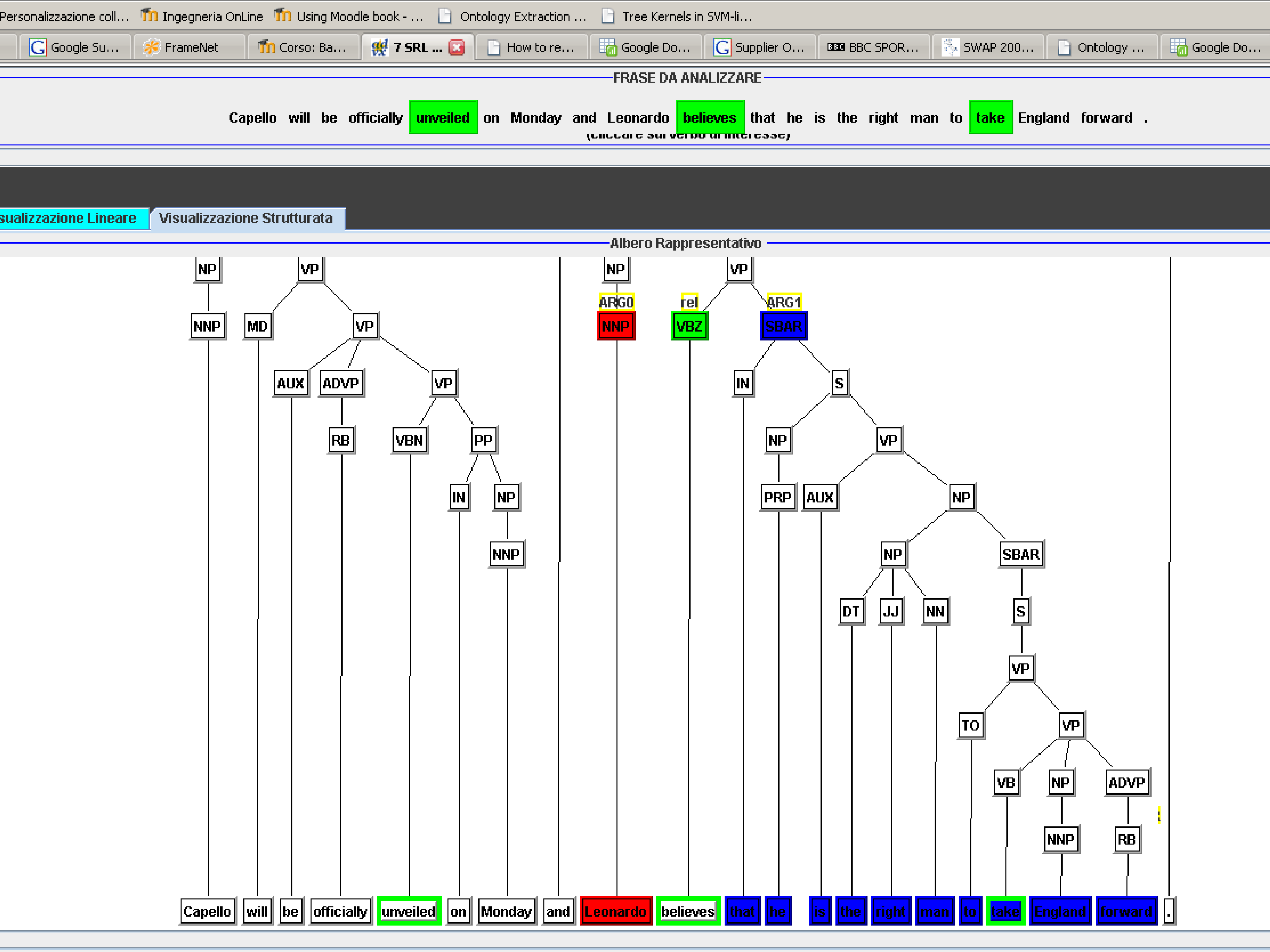
Capello will be officially **unveiled** on Monday and Leonardo **believes** that he is the right man to **take** England forward .

(cliccare sul verbo di interesse)

Visualizzazione Lineare | Visualizzazione Strutturata

Albero Rappresentativo




Personalizzazione coll... Ingegneria OnLine Using Moodle book - ... Ontology Extraction ... Tree Kernels in SVM-li...
Google Su... FrameNet Corso: Ba... 7 SRL ... How to re... Google Do... Supplier O... BBC SPOR... SWAP 200... Ontology ... Google Do...
FRASE DA ANALIZZARE
Capello will be officially unveiled on Monday and Leonardo believes that he is the right man to take England forward .
(cliccare sul verbo di interesse)
sualizzazione Lineare
Visualizzazione Strutturata
Albero Rappresentativo
NP VP NP VP
ARG0 rel ARG1
NNP MD VP NNP VBZ SBAR
AUX ADVP VP IN S
RB VBN PP NP VP
IN NP PRP AUX NP
NNP NP SBAR
DT JJ NN S
VP
TO VP
VB NP ADVP
NNP RB
Capello will be officially unveiled on Monday and Leonardo believes that he is the right man to take England forward .

Semantic Role Labeling via SVM Learning

- Two steps:
  - Boundary Detection
    - One binary classifier applied to the parse tree nodes
  - Argument Type Classification
    - Multi-classification problem, where *n* binary classifiers are applied, one for each argument class (i.e. frame element)
    - They are combined in a ONE-vs-ALL scheme, i.e. the argument type that is categorized by an SVM with the maximum score is selected

Typical standard flat features in SRL
(Gildea & Jurasfky, 2002)

- In argument classification each decision (i.e. one argument) is described by a set of individual (and mostly boolean) fetures, such as:
  - Phrase Type of the argument
  - Parse Tree Path, between the predicate and the argument
  - Head word
  - Predicate Word
  - Position
  - Voice

An example

Phrase Type

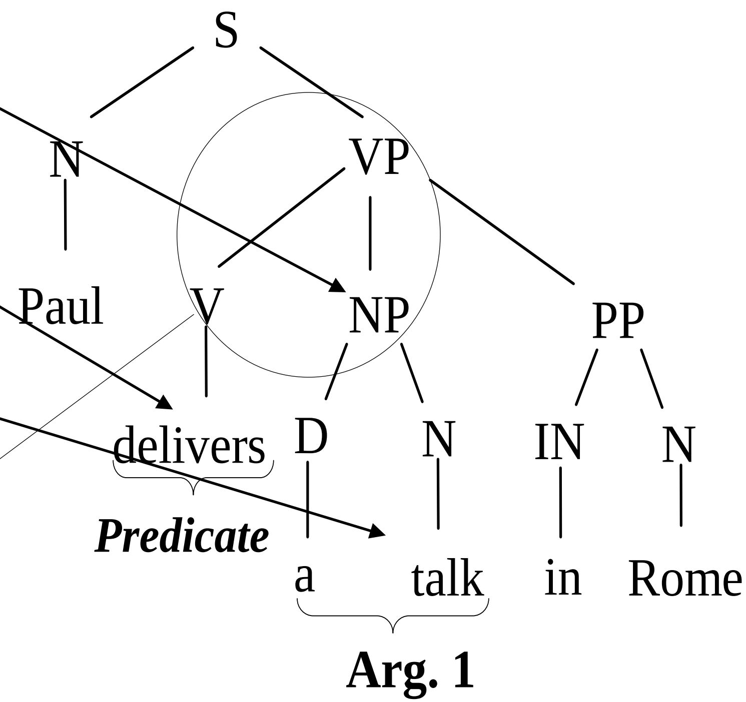
Predicate Word

Head Word

Parse Tree Path
NP↑VP ↓V

Position
Right

Voice
Active



Flat features (Linear Kernel)

- To each argument (i.e. an example) a vector of 6 feature values is associated

$$\vec{x} = (0,..,1,..,0,..,0,..,1,..,0,..,0,..,1,..,0,..,0,..,1,..,0,..,1,\ 1)$$

PT PTP HW PW P V

- The dot product counts the number of features in common

$$\vec{x} \cdot \vec{z}$$

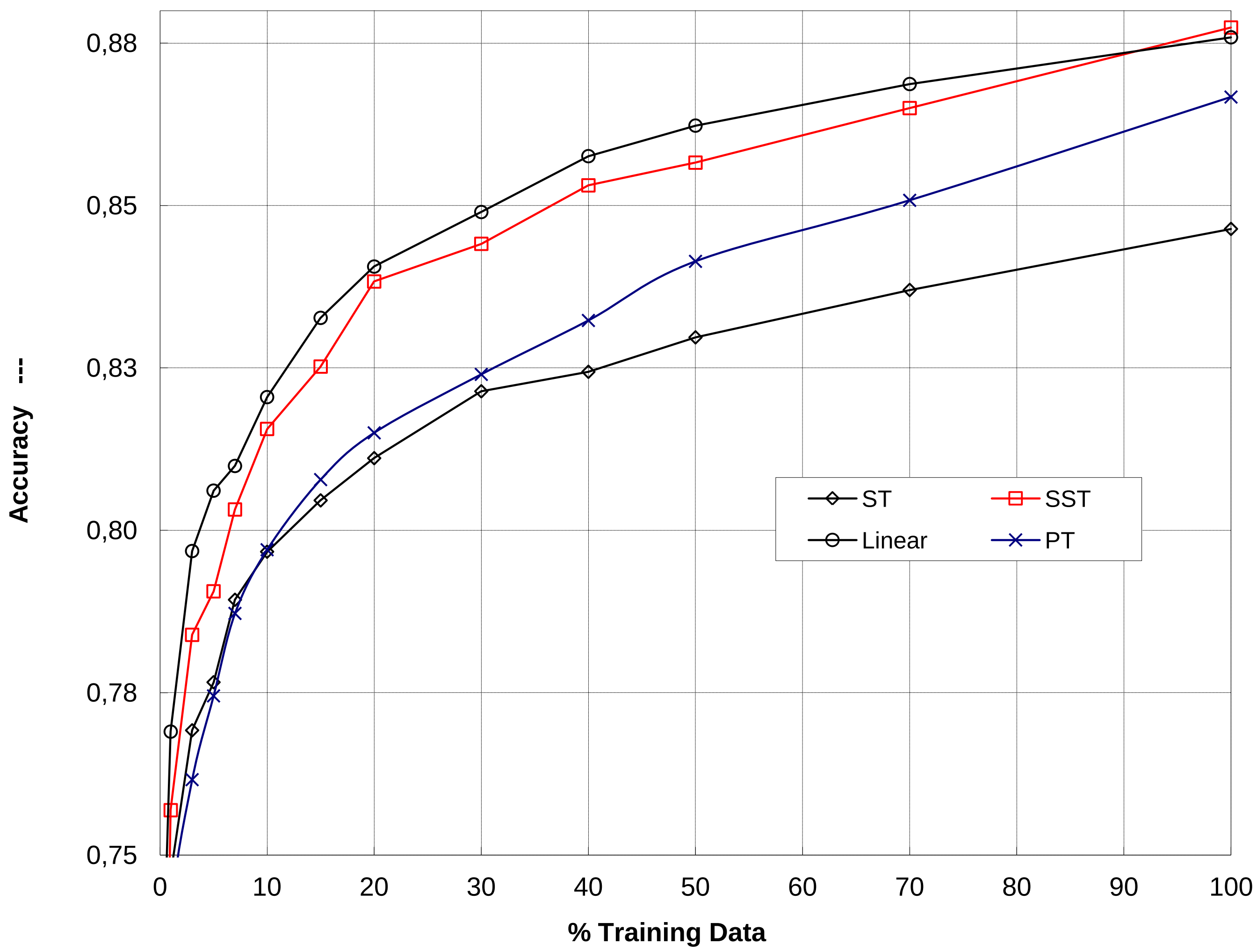
Automatic Predicate Argument Extraction

Deriving Positive/Negative example

Given a sentence, a predicate *p*:

1. Derive the sentence parse tree
2. For each node pair $\langle N_p, N_x \rangle$
   a. Extract a feature representation set *F*
   b. If N_x exactly covers the Arg-*i*, *F* is one of its positive examples
   c. *F* is a negative example otherwise

Argument Classification Accuracy



SRL in Framenet: Results

Eval Setting	Tree Kernels P	Tree Kernels R	Tree Kernels F_1	Tree Kernels + PK P	Tree Kernels + PK R	Tree Kernels + PK F_1
				PK alone		
BD	-	-	-	.887	.675	.767
BD Proj.	-	-	-	.850	.647	.735
BD+RC	-	-	-	.654	.498	.565
BD+RC Proj.	-	-	-	.625	.476	.540
	TK			**TK + PK**		
BD	.949	.652	.773	.915	.698	**.792**
BD Proj.	.919	.631	.748	.875	.668	**.758**
BD+RC	.697	.479	.568	.680	.519	**.588**
BD+RC Proj.	.672	.462	.548	.648	.495	**.561**
	TKL			**TKL + PK**		
BD	.938	.659	.774	.908	.701	.791
BD Proj.	.906	.636	.747	.868	.670	.757
BD+RC	.689	.484	.569	.675	.521	.588
BD+RC Proj.	.663	.466	.547	.644	.497	.561

Table 4.1: Results on FrameNet dataset. The table shows Precision, Recall, and F-measure achieved by the Polynomial Kernel (PK) and two different Tree Kernels (TK and TKL). Also, results for their combinations are shown. All experiments exploit 2% training data for Boundary Detection, and 90% for Role Classification.

Framenet SRL: best results

- Best system [Erk&Pado, 2006]
  - 0.855 Precision, 0.669 Recall
  - 0.751 F1
- Trento (+RTV) system (Coppola, PhD2009)

Enhanced PK+TK			
Eval Setting	P	R	F_1
BD (nodes)	1.0	.732	.847
BD (words)	.963	.702	.813
BD+RC (nodes)	.784	.571	.661
BD+RC (words)	.747	.545	.630

Table 4.2: Results on the FrameNet dataset. Best configuration from Table 4.1, raised to 90% of training data for BD and RC.

- (Croce et al, EMNLP 2011), about 89% in argument classification

Conclusions

- Kernel –based learning is very useful in NLP as for the possibility of embedding similarity measures for highly structured data
  - Sequence
  - Trees
- Tree kernels are a natural way to introduce syntactic information in natural language learning.
  - Very useful when few knowledge is available about the proposed problem.
  - Alleviate manual feature engineering (predicate knowledge)
- Different forms of syntactic information require different tree kernels.
  - Collins and Duffy's kernel (SST) useful for constituent parsing
  - The new Partial Tree kernel useful for dependency parsing

Conclusions (2)

- Experiments on SRL and QC show that
  - PT and SST are efficient and very fast
  - Higher accuracy when the proper kernel is used for the target task
- Open research issue are
  - Proper kernel design issues for the different tasks
  - Combination of syntagmatic kernels with semantic ones
    - An example is the Wordnet-based kernel in (Basili et al CoNLL 05))

… recent stories

- Distributional Analysis:
  - From document vectors to word spaces
  - Paradigmatic lexical similarity
- Croce, Moschitti and Basili paper at EMNLP 2011
  - Partial (and Semantically) Smoothed Tree Kernels (SPTK)
  - Syntagmatic and Lexical similarity
- Application of SPTK to verb classification (Croce et al., ACL 2012)

Tree-kernel: References

- Available over the Web:

  - A. Moschitti, *A study on Convolution Kernels for Shallow Semantic Parsing*. In proceedings of the 42-th Conference on Association for Computational Linguistic (ACL-2004), Barcelona, Spain, 2004.
  - A. Moschitti, *Efficient Convolution Kernels for Dependency and Constituent Syntactic Trees*. In Proceedings of the 17th European Conference on Machine Learning, Berlin, Germany, 2006.
  - M. Collins and N. Duffy, 2002, *New ranking algorithms for parsing and tagging: Kernels over discrete structures, and the voted perceptron*. In ACL02, 2002.
  - S.V.N. Vishwanathan and A.J. Smola. *Fast kernels on strings and trees*. In Proceedings of Neural Information Processing Systems, 2002.

More recent work

- Distributional Models
  - Basili & Pennacchiotti, JNLE 2010
  - Croce and Previtali, GEMS 2010
- SPTKs
  - Croce D. A. Moschitti, R. Basili, EMNLP 2011
  - Croce D., Filice S., R. Basili, Cicling 2012
  - Croce D., A. Moschitti, R. Basili, M. Palmer, ACL 2012.