Case Grammar
Semantic Role Labeling

Semantics of events in sentences

- In a sentence, a verb and its semantic roles form a proposition; the verb can be called the predicate and the roles are known as arguments.

When Disney offered to pay Mr. Steinberg a premium for his shares, the New York investor didn't demand the company also pay a premium to other shareholders.

Example semantic roles for the verb "pay" (using verb-specific roles)

When [$_{payer}$ Disney] offered to [$_V$ pay] [$_{recipient}$ Mr. Steinberg] [$_{money}$ a premium] for [$_{commodity}$ his shares], the New York investor …

CASE Grammar

- **Fillmore, Charles (1968) *"The Case for Case."***
- A response to Chomsky's disregard for any semantics
  - "A semantically justified syntactic theory"

- Given a sentence, it is possible to say much more than this NP is the subject and this NP is the object

- Chomsky's Transformational Grammar would reduce active & passive versions of the same deep structure, but doesn't go far enough to reveal why this is is possible semantically
  - *A crowbar could open that door easily.*
  - *That door could be opened easily with a crowbar.*

CASE Grammar

- Focuses on conceptual events
  - for each event or situation, there is a limited number of roles/cases which people or objects play in the situation
  - roles reflect ordinary human judgments about:
    - Who did the action?
    - Who / what was it done to?
    - What was it done with?
    - Where was it done?
    - What was the result?
    - When was it done?

CASE Grammar (cont'd)

- Syntactic similarities hide semantic dissimilarities
  - *__We__ baked every Saturday morning.*
  - *The __pie__ baked to a golden brown.*
  - *This __oven__ bakes evenly.*
  - 3 subject NPs perform very different roles in regard to *bake*

- Syntactic dissimilarities hide semantic similarities
  - John$_{\text{agent}}$ broke the window$_{\text{theme}}$.
  - John$_{\text{agent}}$ broke the window$_{\text{theme}}$ with a rock$_{\textit{instrument}}$.
  - The rock$_{\textit{instrument}}$ broke the window$_{\text{theme}}$.
  - The window$_{\text{theme}}$ broke.
  - The window$_{\text{theme}}$ was broken by John$_{\text{agent}}$.

Cases (aka Thematic Roles or Theta Roles)

- Fillmore's original set of roles
  - Agentive (A)
  - Instrumental (I)
  - Locative (L)
  - Dative (D)
  - Neutral (N)
  - Factitive (F)

Cases (cont'd)

- **Agentive (A)**
  - the instigator of the action, an animate being
    - ***John*** *opened the door.*
    - *The door was opened by* ***John****.*
- **Instrumental (I)**
  - the thing used to perform the action, an inanimate object
    - *The* ***key*** *opened the door.*
    - *John opened the door with the* ***key****.*
    - *John used the* ***key*** *to open the door.*

Cases (cont'd)

- **Locative (L)**
  - the location or spatial orientation of the state or action identified by the verb
    - ***Chicago*** *is windy.*
    - *It's windy in* ***Chicago****.*
- **Dative (D)**
  - the case of animate being affected by the state or action identified by the verb
    - ***John*** *believed that he would win.*
    - *We persuaded* ***John*** *that he would win.*
    - *We made* ***him*** *a jacket.*

Cases (cont'd)

- **Neutral (N)**
  - The thing being acted upon

- **Objective (O):** the case of anything representable by a noun whose role in the action or state is identified by the semantic interpretation of the verb itself
    - *The **door** opened.*
    - *The wind opened the **door**.*

- **Factitive (F):** the case of the object or being resulting from the action or state identified by the verb, or understood as a part of the meaning of the verb
    - *We made him a **jacket**.*

Verb-specific Roles

- Difficult to fit many verbs and roles into the general thematic roles
  - Many general sets are proposed; not uniform agreement
  - Generalized semantic roles now often called proto roles
    - Proto-agent, proto-patient, etc.
- Verb-specific roles are proposed in systems
  - PropBank annotates the verbs of Penn Treebank
    - Extended with NomBank for nominalizations
  - FrameNet annotates the British National Corpus

Propbank

- Propbank is a corpus with annotation of semantic roles, capturing the semantic role structure of each verb sense
  - Funded by ACE to Martha Palmer and Mitch Marcus at U Penn
- Each verb sense has a frameset, listing its possible semantic roles
    - Argument notation uses numbers for the annotation
    - First sense of accept (accept.01)
      - Arg0: acceptor
      - Arg1: thing accepted
      - Arg2: accepted-from
      - Arg3: attribute
- The frameset roles are standard across all syntactic realizations in the corpus of that verb sense
  - Each verb has a frameset file describing the args as above
    - Example texts are also given

Roles consistent with VerbNet

- Propbank builds on VerbNet to assign more specific roles.
- VerbNet is one extension of Levin's verb classes, giving semantic roles from about 20 possible roles
  - Agent, Patient, Theme, Experiencer, etc.
  - Similar to the theta roles
- Each class consists of a number of synonymous verbs that have the same semantic and syntactic role structure in a frame
- Whenever possible, the Propbank argument numbering is made consistent for all verbs in a VerbNet class.
  - There is only 50% overlap between Propbank and VerbNet verbs.
- Example from frameset file for "explore", which has a VN class:

```
<roleset id="explore.01" name="explore, discover new places or things" vncls="35.4">
<roles> <role descr="explorer" n="0">
        <vnrole vncls="35.4" vntheta="Agent"/></role>
        <role descr="thing (place, stuff) explored" n="1">
        <vnrole vncls="35.4" vntheta="Location"/></role>
</roles>
```

Semantic Role Notation for Propbank

- The first two numbered arguments correspond, approximately, to the core case roles:
  - Arg0 – Prototypical Agent
  - Arg1 – Prototypical Patient or Theme
  - Remaining numbered args are verb specific case roles, Arg2 through Arg5
- Another large groups of roles are the adjunctive roles (which can be applied to any verb) and are annotated as ArgM with a suffix:
  - ArgM-LOC – location
  - ArgM-EXT – extent
  - ArgM-DIR – direction
  - ArgM-ADV – general purpose adverbial
  - ArgM-DIS – discourse connective
  - ArgM-MOD – modal verb

  ArgM-CAU - cause
  ArgM-TMP - time
  ArgM-PNC – purpose
  ArgM-MNR - manner
  ArgM- NEG – negation

Adjunctive and additional arguments

- Example of adjunctive arguments
  - Not all core arguments are required to be present
    - See Arg2 in this example.
  - Arguments can be phrases, clauses, even partial words.

When Disney offered to pay Mr. Steinberg a premium for his shares, the New York investor didn't demand the company also pay a premium to other shareholders.

Example of Propbank annotation (on demand):
[$_{\text{ArgM-TMP}}$ When Disney offered to pay Mr. Steinberg a premium for his shares], [$_{\text{Arg0}}$the New York investor] did [$_{\text{ArgM-NEG}}$ n't] [$_{\text{V}}$ demand] [$_{\text{Arg1}}$ the company also pay a premium to other shareholders].

Where for demand, Arg0 is "asker", Arg1 is "favor", Arg2 is "hearer"

Prepositional phrases and additional args

- Arguments that occur as the head of a prepositional phrase are annotated as the whole phrase
  - Consistent with other ArgM's that are prepositional phrases

  [$_{\text{Arg1}}$ Its net income] [$_{\text{V}}$ declining] [$_{\text{ArgM-EXT}}$ 42%] [$_{\text{Arg4}}$ to $121 million] [$_{\text{ArgM-TMP}}$ in the first 9 months of 1989]

- Additional arguments are
  - ArgA – causative agents
  - C-Arg* - a continuation of another arg (mostly for what is said)
  - R-Arg* - refererence to another arg (mostly for "that")

Propbank Annotations

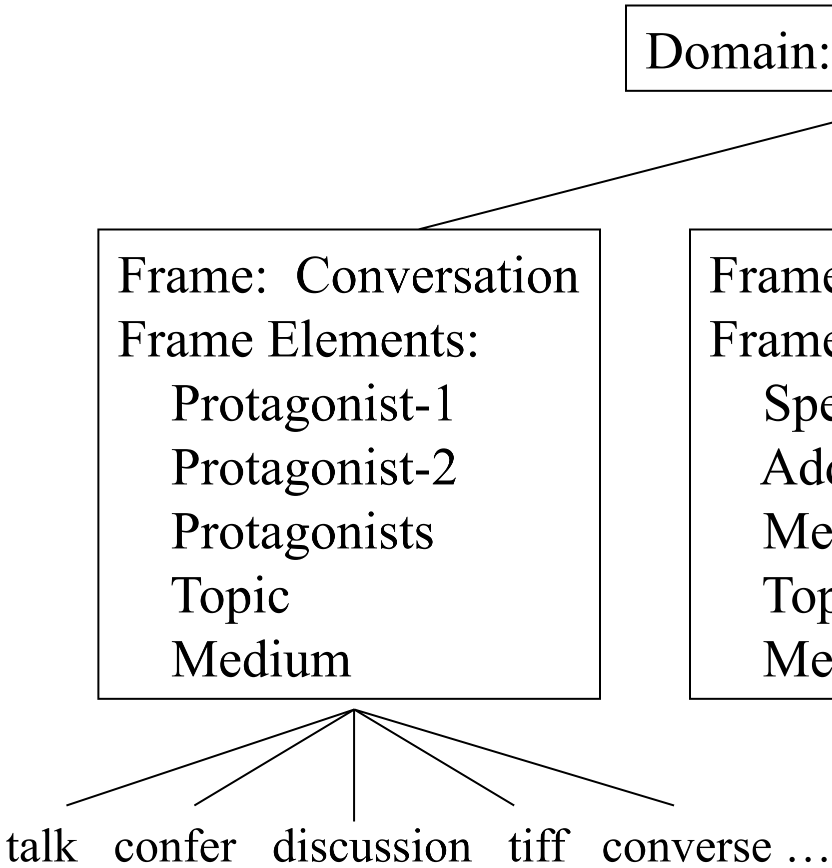
- Framesets were created by looking at sample sentences containing each verb sense.
  - ~ 4500 frames (in 3314 framesets for each verb)
- Corpus is primarily newswire text from Penn Treebank
  - Annotated the Wall Street Journal section, and, more recently, the "Brown" corpus
  - Verbs and semantic role annotations added to the parse trees
- Annotators are presented with roleset descriptions of a verb and the (gold) syntactic parses of a sentence in Treebank, and they annotate the roles of the verb.
  - Lexical sampling – annotated on a verb-by-verb basis.
  - ~40,000 sentences were annotated
- Interannotater agreement
  - Identifying argument and classifying role: 99%
    - kappa statistic of .91overall and .93 if ArgM's excluded

FrameNet

- Project at International Computer Science Institute with Charles Fillmore
  - http://framenet.icsi.berkeley.edu/
- Similar goal to document the syntactic realization of arguments of predicates in the English language
- Starts from semantic frames (e.g. Commerce) and defines frame elements (e.g. Buyer, Goods, Seller, Money)
- Annotates example sentences chosen to illustrate all possibilities
  - But latest release includes 132,968 sentences
  - British National Corpus

Example of FrameNet frames

- Semantic frames are related by topic domain



Comparison of FrameNet and Propbank

- FrameNet semantic roles are consistent for semantically related verbs (not just synonyms as in the VerbNet subset of PropBank)
- Commerce examples:

FrameNet annotation:

[$_{\text{Buyer}}$ Chuck] *bought* [$_{\text{Goods}}$ a car] [$_{\text{Seller}}$ from Jerry][$_{\text{Payment}}$ for $1000].
[$_{\text{Seller}}$ Jerry] *sold* [$_{\text{Goods}}$ a car] [$_{\text{Buyer}}$ to Chuck] [$_{\text{Payment}}$ for $1000].

Propbank annotation:

[$_{\text{Arg0}}$ Chuck] *bought* [$_{\text{Arg1}}$ a car] [$_{\text{Arg2}}$ from Jerry][$_{\text{Arg3}}$ for $1000].
[$_{\text{Arg0}}$ Jerry] *sold* [$_{\text{Arg1}}$ a car] [$_{\text{Arg2}}$ to Chuck] [$_{\text{Arg3}}$ for $1000].

Frame for buy:
Arg0: buyer
Arg1: thing bought
Arg2: seller
Arg3: price paid
Arg4: benefactive

Frame for sell:
Arg0: seller
Arg1: thing sold
Arg2: buyer
Arg3: price paid
Arg4: benefactive

Automatic SRL

- Define an algorithm that will process text and recognize roles for each verb
- Assume previous levels of Natural Language Processing (NLP) on text
  - Part-of-speech (POS) tagging,
  - Chunking, i.e. recognizing noun and verb phrases,
  - Clauses,
  - Parse trees
- Machine Learning approaches are typical

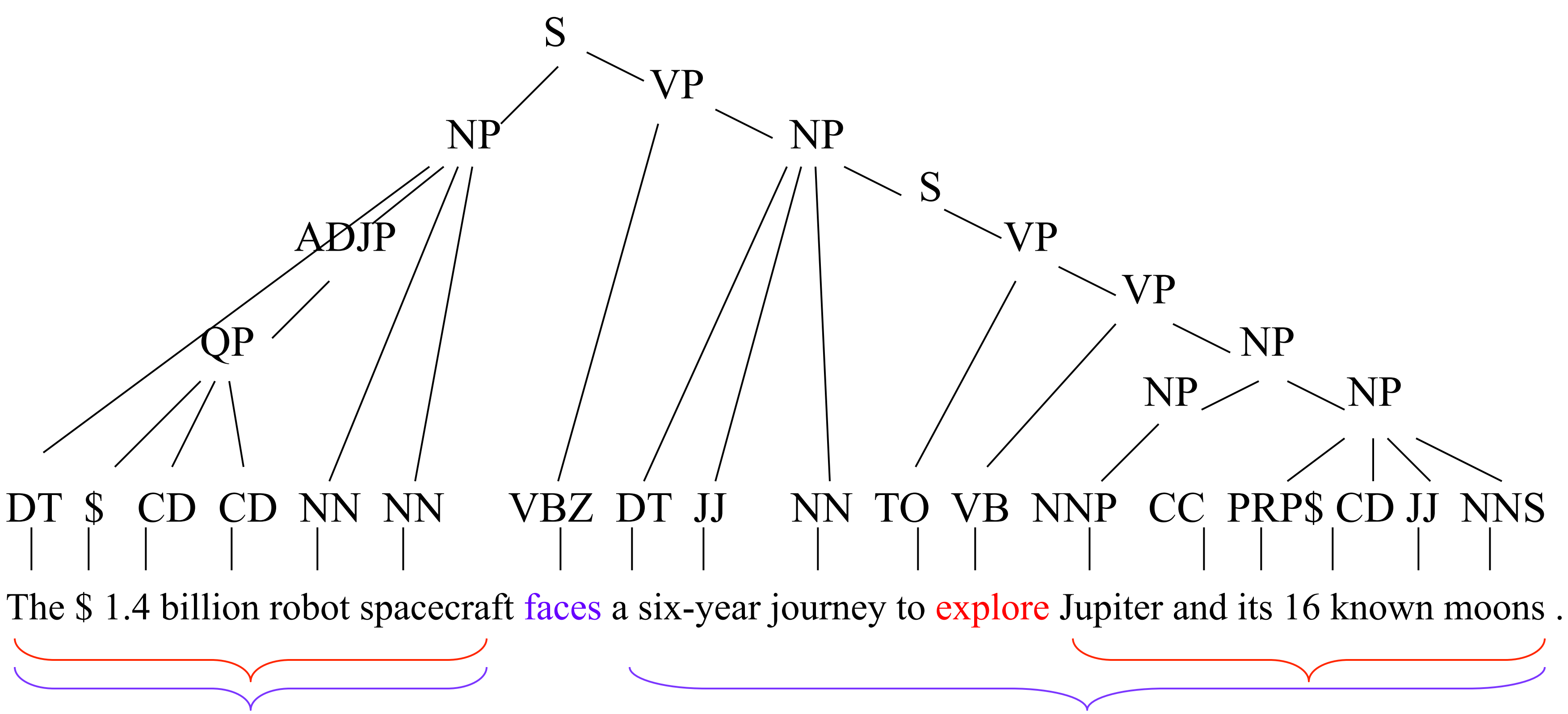
Machine Learning Approach

- Given a verb in a sentence, the problem is to find and label all arguments
- Reformulate as a classification task: For each constituent in the parse tree of the sentence, label it as to what argument, if any, it is for the verb

- For each constituent, define features of semantic roles
  - Each feature describes some aspect of a text phrase that can help determine its semantic role of a verb
    - Examples include what the verb is, POS tags, position in parse tree, etc.
- Machine Learning process:
  - Training:
    - collect examples of semantic roles with features and semantic role label
    - ML training program uses examples to produce decision algorithm
  - Classification:
    - Run decision algorithm on text phrases and it will decide which, if any, semantic role it plays with respect to a verb

Parse Tree Constituents

- Each syntactic constituent is a candidate for labeling
- Define features from sentence processed into parse tree with Part-of-Speech tags on words



Typical Argument Features

- These features are defined for each constituent:
- PREDICATE: The predicate word from the training data.
  - "face" and "explore"
  - Usually stemmed or lemmatized
- PHRASE TYPE: The phrase label of the argument candidate.
  - Examples are NP, S, for phrases, or may be POS tag if a single word
- POSITION: Whether the argument candidate is before or after the predicate.
- VOICE: Whether the predicate is in active or passive voice.
  - Passive voice is recognized if a past participle verb is preceded by a form of the verb "be" within 3 words.
- SUBCATEGORY: The phrase labels of the children of the predicate's parent in the syntax tree.
  - subcat of "faces" is "VP -> VBZ NP"

Argument Features

- PATH: The syntactic path through the parse tree from the argument constituent to the predicate.
  - Arg0 for "faces": NP -> S -> VP -> VBZ
- HEAD WORD: The head word of the argument constituent
  - Main noun of NP (noun phrase)
  - Main preposition of PP (prepositional phrase)
- Many additional features
  - Head Word POS: The part of speech tag of the head word of the argument constituent.
  - Temporal Cue Words: Special words occurring in ArgM-TMP phrases.
  - Governing Category: The phrase label of the parent of the argument.
  - Grammatical Rule: The generalization of the subcategorization feature to show the phrase labels of the children of the node that is the lowest parent of all arguments of the predicate.

Highlights of Automatic SRL Research

- Burst of research in SRL from 2002 - 2009:
  - different machine learning approaches
  - features
- Gildea and Jurafsky, 2002. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3):245-288. Used a probabilistic model, full parse, on FrameNet.
- CoNLL-2004 shared task. 10 teams used a variety of approaches, chunks + clauses, Propbank.
- Senseval-3 semantic role task, 2004. 8 teams used a variety of approaches, full parses, FrameNet.
- CoNLL-2005 shared task. 21 teams used a variety of approaches, full parses, Propbank.

CoNLL-2005 Shared Task

- Each year, CoNLL defines a task to develop some aspect of natural language processing with systems that use machine learning.
  - Provides data for training and developing systems for about 3 months
  - Then provides test data; everyone runs their system and returns the results for scoring
  - Competitive in that scores are published in a comparative way
  - Collaborative in that a session of the annual conference is devoted to discussion of the progress in this task
    - Novel approached are encouraged
- The CoNLL-2004 shared task aimed at evaluating machine learning SRL systems based on partial syntactic information.
  - Best results are approximately 70 in F measure.
- The 2005 shared task evaluated machine learning SRL systems based on full parse information

Input data

- For each sentence, the following data is given for all the data sets:
  - Target verbs
  - Named Entities,
    - with a category from Person, Organization, Location or Miscellaneus.
  - PoS tags,
  - partial parses, including noun and verb chunks and clauses
  - col2 : full parses from Collins‘ statistical parser,
  - cha: full parses of Charniak’s statistical parser,
  - VerbNet senses of target verbs.
- In addition, the training and development sets have the gold standard correct propositional arguments

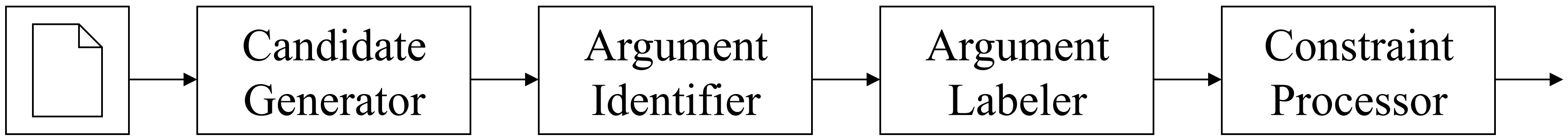
Example input data (column format)

WORDS----> NE---> POS PARTIAL_SYNT FULL_SYNT------> VS TARGETS PROPS------->

The	*	DT	(NP*	(S*	(S(NP*	-	-	(A0*	(A0*
$	*	$	*	*	(ADJP(QP*	-	-	*	*
1.4	*	CD	*	*	*	-	-	*	*
billion	*	CD	*	*	*))	-	-	*	*
robot	*	NN	*	*	*	-	-	*	*
spacecraft	*	NN	*)	*	*)	-	-	*)	*)
faces	*	VBZ	(VP*)	*	(VP*	01	face	(V*)	*
a	*	DT	(NP*	*	(NP*	-	-	(A1*	*
six-year	*	JJ	*	*	*	-	-	*	*
journey	*	NN	*)	*	*	-	-	*	*
to	*	TO	(VP*	(S*	(S(VP*	-	-	*	*
explore	*	VB	*)	*	(VP*	01	explore	*	(V*)
Jupiter	(ORG*)	NNP	(NP*)	*	(NP(NP*)	-	-	*	(A1*
and	*	CC	*	*	*	-	-	*	*
its	*	PRP$	(NP*	*	(NP*	-	-	*	*
16	*	CD	*	*	*	-	-	*	*
known	*	JJ	*	*	*	-	-	*	*
moons	*	NNS	*)	*)	*)))))))	-	-	*)	*)
.	*	.	*	*)	*)	-	-	*	*

Typical architecture

- Our system followed a typical architecture that utilizes two different machine learning phases
  - Filter out implausible constituents from the parse trees
  - Use a machine learning classifier to decide if each of the remaining constituents is an argument to the verb
  - Use a machine learning classifier to decide which argument label (Arg0-Arg5, ArgM's, etc.) to put on the argument
  - Do some final constraint processing



Support Vector Machines (SVM)

- Both classifiers are trained with the libSVM software.
- libSVM is an open source software package
  - http://www.csie.ntu.edu.tw/~cjlin/libsvm
- Kernel functions: Radial Basis Functions (RBF)
  - Used grid experimental approach to optimize the two parameters (C and gamma)
- For the identification classifier
  - Binary classifier to decide if each parse tree constituent is an argument
- For the labeling classifier
  - N binary classifiers, each producing a probability estimate of whether an argument should have that label
  - Use the probabilities in the constraint problem

Classifier Training Set

- 18741 total number of features (attribute values)
- Example Count = 233100

A0 =	60328	%25	AM-LOC =	5688	C-A0 =	109
A1 =	79276	%34	AM-DIR =	1113	C-A1 =	2233
A2 =	18962	%8	AM-DIS =	4869	R-A0 =	4104
A3 =	3172	%1.3	AM-MOD =	9180	R-A1 =	2335
A4 =	2557	%1.1	AM-CAU =	1165	R-AM-MNR =	143
A5 =	68		AM-TMP =	16031	R-AM-LOC =	214
			AM-MNR =	6208	others	
			AM-PNC =	2175		
			AM-ADV =	8005		
			AM-NEG =	3220		

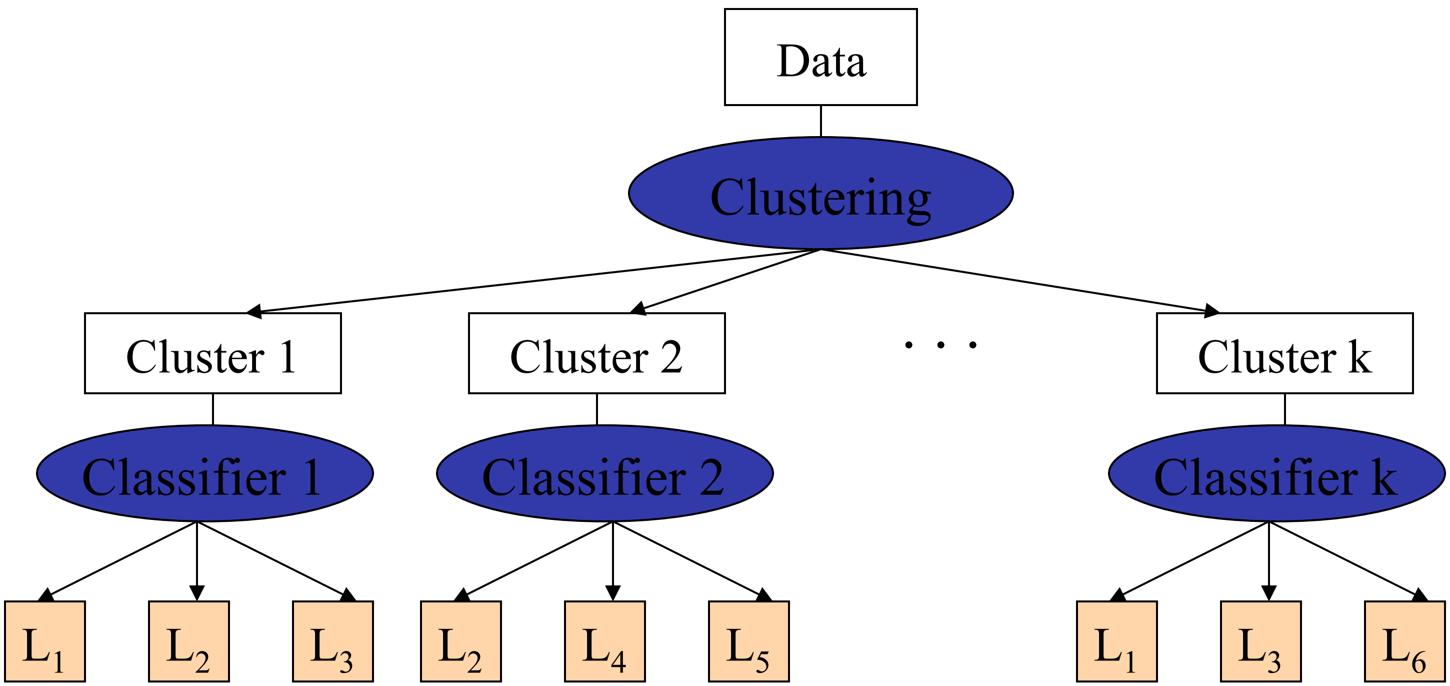
SRL problem constraints

- Main constraints
  - Two constituents cannot have the same argument label,
  - A constituent cannot have more than one label
  - If two constituents have (different) labels, they cannot have any overlap,
  - No argument can overlap the predicate.

- Additional constraints:
  - For R-Ax, there should be an Ax
  - For C-Ax, there should be an Ax

Cluster-Based Classification (CBC)

- A type of ensemble classification that divides the problem and trains a classifier for each subproblem, with a subset of the labels L_1, L_2, …
- Our approach divides the problem with clustering
- Used Support Vector Machines, libSVM package, as the base classifier



Results of Argument Labeling Classifier

- Compare the results of the CBC classifier on the entire SRL problem (identifier + labeler + post processor) with other systems (Koomen et al[1]), using a single parse tree, but from different parsers

	Precision	Recall	$F_{\beta=1}$
Charniak-1	75.40%	74.13%	74.76
Charniak-2	74.21%	73.06%	73.63
Charniak-3	73.52%	72.31%	72.91
Collins	73.89%	70.11%	71.95
CBC	80.63%	71.23%	75.64

- Results using a single parse tree are just part of the overall problem; best results (2005) combine results from different parse trees, e.g.

Joint Inference	80.05%	74.83%	77.35

[1] Peter Koomen, Vasin Punyakanok, Dan Roth, and Wen-tau Yih. Generalized inference with multiple semantic role labeling systems. Proceedings CoNLL-2005.

Current Direction of SRL

- Best English SRL results combining parse trees or combining the parsing task with the SRL task (joint inference) are at just over F-measure of 80
- CoNLL 2009 shared task is SRL again, but systems combined dependency parsing with semantic role labeling.
  - Joint detection of syntactic and semantic dependencies
  - Richer syntactic dependency set to aid in semantic processing
- See http://barcelona.research.yahoo.net/conll2008/ for a description of the task for English
- English, Catalan, Chinese, Czech, German, Japanese, Spanish
- Most systems, including top scoring systems, did not use joint inference
- Unanswered question: Can applications make good use of SRL?