Automatic Induction of Semantic Verb Classes incorporating Selectional Preferences

Sabine Schulte im Walde
Christian Hying
Helmut Schmid
Christian Scheible

Institut für Maschinelle Sprachverarbeitung
SFB 732/D4, Universität Stuttgart

Universität Potsdam
November 15, 2007

SFB 732, Project D4

Project D4 in the SFB 732

Incremental Specification in Context →
Modular Lexicalisation of Probabilistic Context-Free Grammars

- Statistical parsing with treebank grammars
- Modular extensions of unlexicalised PCFGs
- Goals:
  » Modelling context by multi-dimensional soft clusters
  » Induction of lexical information:
  verb senses and verb classes, subcategorisation and selectional restrictions, verb alternations
  » Statistical disambiguation for parse trees

Semantic Verb Classifications

Semantic Verb Classifications

- Groupings of verbs according to semantic properties
- Classes refer to general semantic level; idiosyncratic lexical semantic properties are underspecified
- Intuitive examples:
  - » motion with a vehicle: *drive, fly, row,* etc.
  - » break a solid surface with an instrument: *break, crush, fracture, smash,* etc.
- Manual definitions for several languages: *English* (Levin 1993; Fellbaum 1998; Fillmore et al. 2003), *Spanish* (Vázquez et al. 2000), etc.

SVCs: Interest & Application

- Theoretical linguistics: organise verbs with respect to common properties, such as meaning components (Koenig & Davis 2001), or shared argument structure (Levin 1993)

- Computational Linguistics:
  underspecification / generalisation over shared properties
  → data sparseness in processing natural language
  → applications: word sense disambiguation (Dorr & Jones 1996; Kohomban & Lee 2005), machine translation (Prescher et al. 2000; Koehn & Joang 2007), document classification (Klavans & Kan 1998), etc.

Class Induction & Result

- **Verbs → classes**

- Verbs in common class: as similar as possible

- Verbs in different classes: as dissimilar as possible

- Parameters in automatic induction:
  verbs, verb properties, algorithm

Verb Properties

- Model semantic similarity of interest
- Similarity at the syntax-semantics interface
- Potentially salient features:
  - » syntactic frames
  - » prepositional phrases
  - » argument role fillers
  - » adverbial adjuncts, etc.
- Our choice: selectional preferences

Selectional Preferences

- Semantic realisation of a predicate’s complement
- Reference to the syntactic function and the thematic role
- Example: *drink tea, drink coffee, drink beer,* etc.
  → *drink a beverage* (→ *drink a substance*)
- Preference: degree of acceptability
- Requires inventory (and organisation) of semantic categories → clusters / WordNet

WordNet

- Lexical semantic taxonomy developed at Princeton University (Miller, 1990; Fellbaum, 1998)
- Psycholinguistic research on human lexical memory
- Organisation of English nouns, verbs, adjective, and adverbs into sets of synonymous words (synsets)
- Lexical and conceptual relations between (parts of) synsets: antonymy, hypernymy/hyponymy, etc.
- Words with several senses are assigned to multiple synsets
- WordNet "family": multi-lingual WordNets

WordNet-based SelPref Approaches

- Input:
  corpus-based tuples <predicate, function, noun>
  with respect to a specific functional relationship
  and co-occurrence frequency counts
- Rely on WordNet synsets and WordNet (hypernym) hierarchy
- Task: find WordNet concept(s) that best describe the selectional preferences for the predicate-frame function

WordNet Preferences: *Example*

{entity}

{object, inanimate object, physical object}

{substance, matter}

{fuel} {food, nutrient} {body substance}

{beverage, drink, potable}

{**tea**} {**coffee**, java} {**milk**}

direct objects of *drink*

Verb Class Model

Verb Class Model

- Assumption: verbs in common class agree on selectional preferences
- Soft-clustering approach with *n* verb classes
- Verbs can be assigned to several classes
  → polysemy of verb senses
- Training algorithms: Expectation-Maximisation and Minimum Description Length
- Source for generalising concepts: WordNet

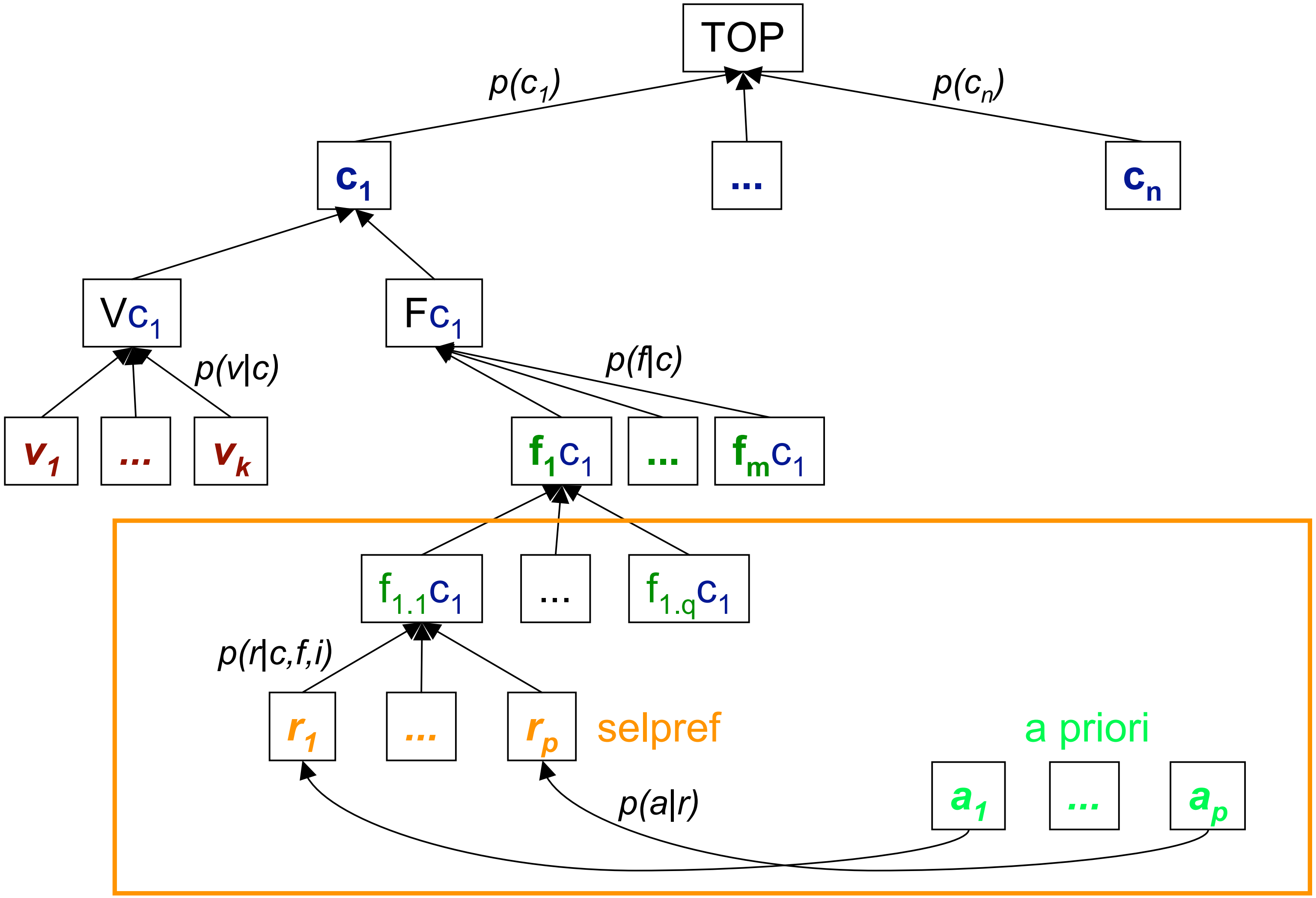
Verb Class Probabilistic Model

p(drink, subj:obj, girl, tea)

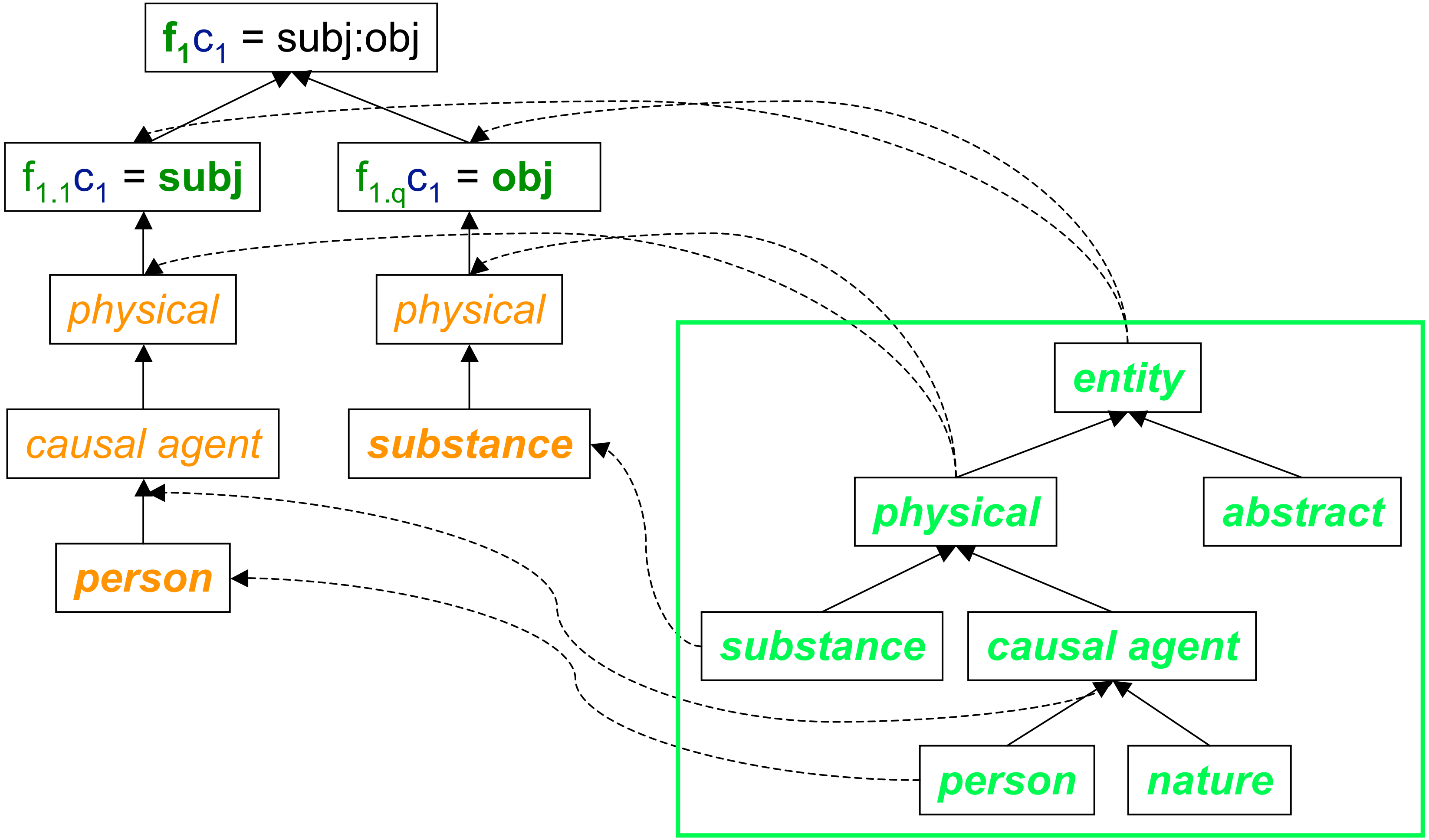
$$p(v, f, a_1, \ldots, a_n) = \sum_{c \in C} p(c)\; p(v \mid c)\; p(f \mid c) \prod_{i=1}^{n_f} \sum_{r \in W} p(r \mid c, f, i)\; p(a_i \mid r)$$

p(c)	probability of verb class *c*
p(v|c)	probability of verb *v* in class *c*
p(f|c)	probability of frame *f* in class *c*
p(r|c,f,i)	probability that i^{th} argument of frame *f* in class *c* is realised by WordNet concept *r*
p(a|r)	probability that WordNet concept *r* is realised by argument head *a*

Implementation: Graph Structure



Implementation: Graph Structure



Verb Class Model: *Steps*

1. Input: verb-frame-argument tuples <$v,f,a_1,\ldots,a_n$>
   » verb v,
   » subcategorisation frame f,
   » list of argument heads $a_1,\ldots,a_n$

   example: <*drink subj:obj girl tea*> *43*

2. Training: Estimation-Maximisation algorithm;
   Minimum-Description Length principle

3. Output: cluster analysis with two dimensions

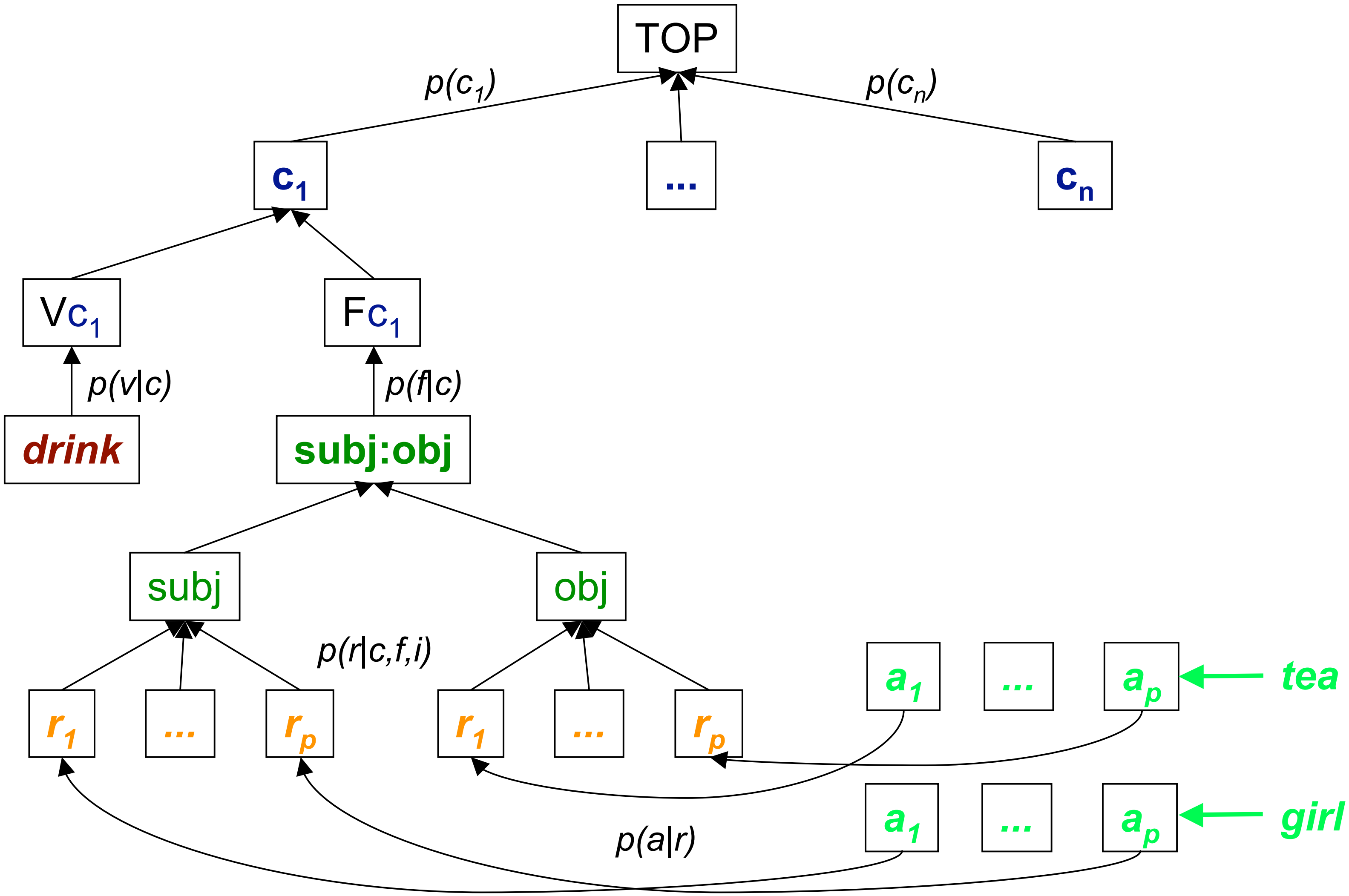
Verb Class Model: *EM Algorithm*

- Expectation-Maximisation algorithm (EM)
- Goal: finding maximum likelihood estimates of parameters in probabilistic models
- Model depends on unobserved latent variables
  → hidden data cluster c, selectional restriction r
- Properties (among others):
  » monotonicity: improvement of likelihood
  » sensitive to initialisation, training data, sparse data
  » guaranteed to find a local optimum in the search space
- Inside-Outside algorithm (IO):
  IO is an instance of EM, used for PCFGs

Verb Class Model: *EM Algorithm*

- Alternation between assessing frequencies and estimating probabilities
- **E-step = estimation**
  computes expectation of likelihood by including the latent variables as if they were observed: valuates probability distribution given the model parameters from the previous iteration → calculation of expected values
- **M-step = maximisation**
  computes maximum likelihood estimates by maximising expected likelihood: finds the new parameter set that maximises the distribution → calculation of ML values

Verb Class Model: *IO on Input Tuple*



Verb Class Model: *Cut-based MDL*

- Selectional preference: cut in the WordNet hierarchy (i.e., a set of disjunctive WordNet classes)
- Formalization of Occam's Razor: the best hypothesis for a given set of data is the one that requires the least code length in bits for the encoding of

  the model itself (model description length) and

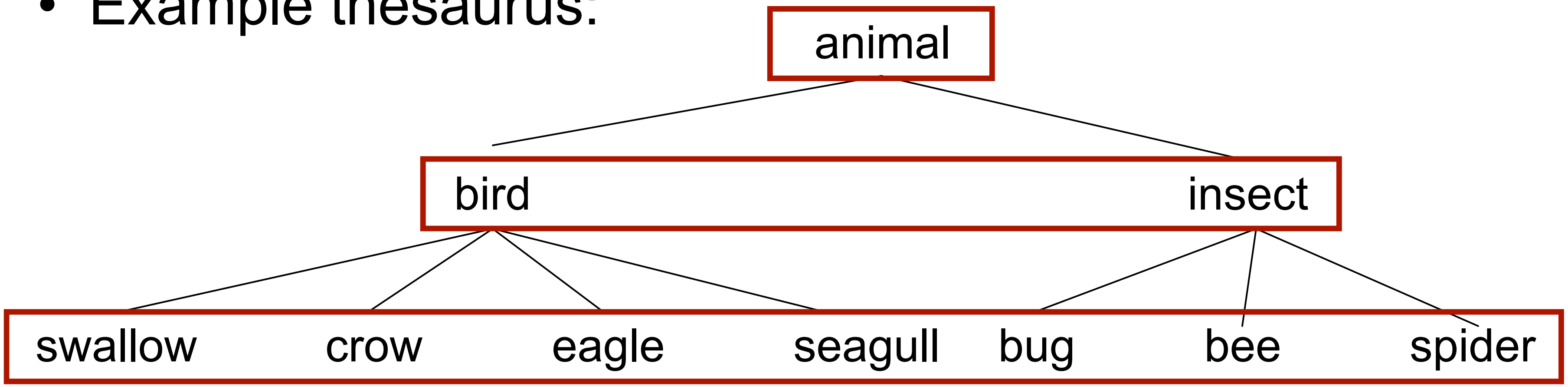
  the data observed through it (data description length)
- Principle from information theory: minimum description length (MDL) finds the cut in the hierarchy which minimises the sum of encoding both the model and the data

Verb Class Model: *MDL (Li & Abe)*

- Example data:

verb	*slot*	*noun*	*freq*
fly	subj	bird	4
fly	subj	bee	2
fly	subj	crow	2
fly	subj	eagle	2

- Example thesaurus:

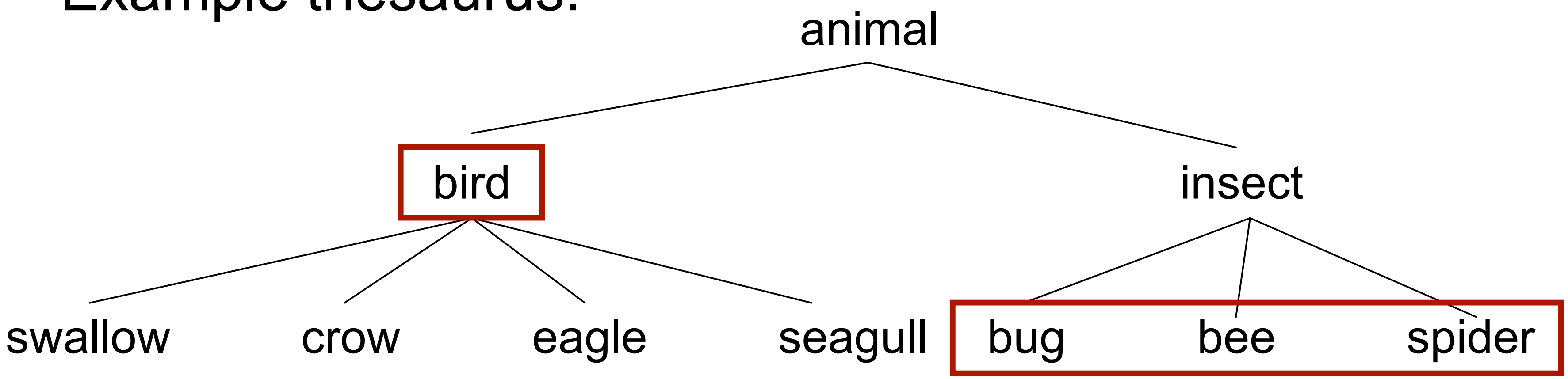


Verb Class Model: *MDL*

- Example data:

verb	*slot*	*noun*	*freq*
fly	subj	bird	4
fly	subj	bee	2
fly	subj	crow	2
fly	subj	eagle	2

- Example thesaurus:



Model Description Length

- Parameter:
  - » number of WordNet classes in cut: *k*
  - » total frequency going through WordNet: |*S*|

- Model Description Length:

$$MD = \frac{k}{2} \times \log_2 |S|$$

Data Description Length

- Parameter:
  - » frequency of class c: *f(c)*
  - » probability of noun in class c: *p(n)*, using size of class c, i.e., number of nouns in c: |*c*|, probability of class c: *p(c)*

$$p(n) = \frac{p(c)}{|c|}$$

- Data Description Length:

$$DD = -\sum_n f(c) \times \log_2 p(n)$$

MDL: *Example*

c	*bird*	*bug*	*bee*	*spider*	
f(c)	8	0	2	0	
|c|	4	1	1	1	
p(c)	0.8	0.0	0.2	0.0	
p(n)	0.2	0.0	0.2	0.0	
cut	[bird, bug, bee, spider]				
MD	4/2 x log 10 = 4.98				Σ = 28.20
DD	- (2+4+2+2) x log 0.2 = 23.22				
cut	[bird, insect]				
MD	1.66				Σ = 28.05
DD	26.39				

MDL Cut: *Example*

Class	*Prob*	*Examples*
DIRECT OBJECT OF *EAT*		
‹food, nutrient›	0.39	*pizza, egg*
‹life form, organism, living being›	0.11	*lobster, horse*
‹measure, quantity, amount›	0.10	*amount of*
DIRECT OBJECT OF *BUY*		
‹inanimate object, physical object›	0.30	*computer, painting*
‹asset›	0.10	*stock, share*
‹group, grouping›	0.07	*bank, company*
DIRECT OBJECT OF *FLY*		
‹entity›	0.35	*airplane, flag, executive*
‹linear measure, long measure›	0.28	*mile*
‹group, grouping›	0.08	*delegation*

Verb Class Model: *EM & MDL*

- Random initial assignment of frequencies/probabilities
- Initialisation of MDL cuts by top level *entity*
- Expansion of MDL cuts by next lower level
- Estimation of graph frequencies, using input tuples
- MDL cuts: leave or prune
- Maximisation of graph probabilities

Verb Class Model: *Examples*

- English
- German

Verb Class Model: *Interpretation*

- Modelling contextual dependencies by multi-dimensional soft clusters
- Induction of lexical information:
  - » verb senses and verb classes
  - » subcategorisation and selectional restrictions
  - » collocations
  - » verb alternations
- Application to sparse data problems in NLP
- Multi-lingual framework (given WordNet)

Verb Class Model: *Parameter*

- Preparation of tuples:
  - » frequencies of tuples
  - » frequencies of cluster objects (verbs, frames, nouns)
  - » special treatment of instances (e.g., pronouns)
- Number of clusters and number of iterations
- Initialisation of probabilities
- MDL model: cut-based vs. synset-based
- Calculation of preferences against the priori model

Verb Class Model: *Evaluation, tbd*

- Likelihood: calculate likelihood of held-out data, given the parameters of the cluster analysis: $L(x|\theta)$
- Pseudo-Word Disambiguation: create artificial verb-noun pairs and distinguish from existing such pairs
- Gold Standard: compare clusters and selectional preferences against existing resources (e.g., Levin classes; dictionary/encyclopedic knowledge)
- Application:
  » use verb class model in parser as lexical information
  » use model to predict compositionality of particle verbs

Related Work

- Soft-clustering (relying on the EM algorithm):
  Pereira et al. 1993; Rooth 1998; Rooth et al. 1999; Korhonen et al. 2003
- Hard classification/clustering of verbs:
  Merlo & Stevenson 2001; Schulte im Walde 2006; Joanis et al. 2007
- Selectional preference models:
  Resnik 1997; Li & Abe 1998; Abney & Light 1999; Ciaramita & Johnson 2000; Clark & Weir 2002; Erk 2007

Summary

- Soft-clustering verb class model:
  - » verb senses according to selectional preferences
  - » multi-lingual framework (WordNet-based)
- Application scenarios:
  - » induction of lexical information
  - » incorporation into NLP applications
- Next steps:
  - » variations and extensions of model
  - » evaluations