

Machine Learning of Language: A Model and a Problem

Walter Daelemans

daelem@uia.ua.ac.be

<http://cnts.uia.ac.be>

CNTS, University of Antwerp

ILK, Tilburg University

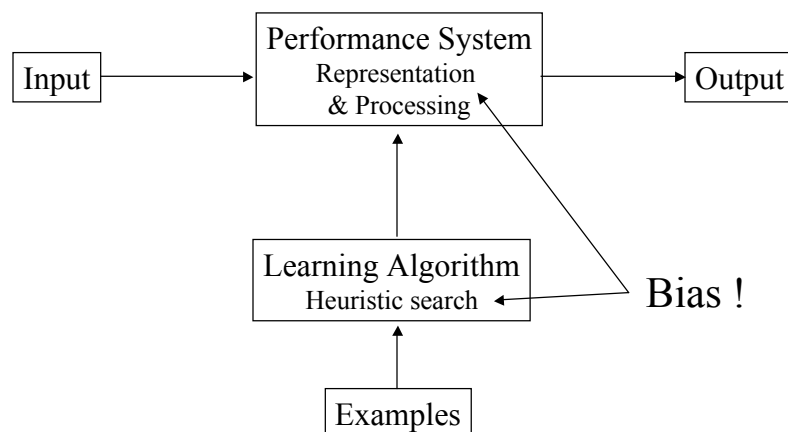
ESSLLI-02 workshop ML of NL

Outline

- Machine Learning of Language
 - Which inductive algorithm has the right bias for language learning?
- Memory-Based Learning
- So, MBL has the right bias?
 - No free lunch (a priori)
 - No lunch whatsoever ! (a posteriori)
- Comparative Machine Learning Methodology

- Machine Learning may alleviate the problems of mainstream statistical methods in NLP
 - Rule induction (understandable induced theories)
 - Inductive Logic Programming (incorporating linguistic knowledge)
 - Memory-based learning (similarity-based smoothing with sparse data)
 - ...
- So, what is the best machine learning method for NLP? Which method has the right “bias”?

(Supervised) Learning



Memory-Based Learning

- Basis: k nearest neighbor algorithm:
 - store all examples in memory
 - to classify a new instance X , look up the k examples in memory with the smallest distance $D(X, Y)$ to X
 - let each nearest neighbor vote with its class
 - classify instance X with the class that has the most votes in the nearest neighbor set
- Choices:
 - similarity metric
 - number of nearest neighbors (k)
 - voting weights

Memory-Based Learning

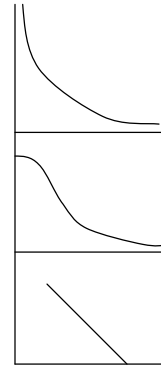
Metrics:

- $D(X, Y) = \sum_i d(x_i, y_i)$
 - $d(x, y)$ (overlap)
symbolic: 1 if $x \neq y$, 0 if $x = y$
numeric: $|x_i - y_i| / (\max_i - \min_i)$
 - $d(x, y)$ (modified value difference metric)
 $\sum_i |P(c_i|x) - P(c_i|y)|$
- $D(X, Y) = \sum_i w_i d(x_i, y_i)$
Feature Weighting, e.g. IG, GR, Chi-squared etc.

Memory-Based Learning

Voting options:

- Equal weight for each nearest neighbor
- Distance weighted voting
 - Inverse distance $1/D(X,Y)$ (Wettschereck, 1994)
 - RBF-style gaussian voting function (Shepard, 1987)
 - Linear voting function (Dudani, 1976)



(NB: weighted NN distribution can be used as conditional probability)

Memory-Based Language Processing

MBLP (MBL for NLP) seems to outperform more greedy learning algorithms consistently for a wide range of NLP tasks

Machine Learning 1999 (Daelemans, van den Bosch, Zavrel; Forgetting exceptions is harmful in language learning)

TiMBL Reference Guide 1998-2002 (Daelemans, Zavrel, van der Sloot, van den Bosch)

<http://ilk.kub.nl/>

The properties of NLP tasks ...

- NLP tasks are mappings between linguistic representation levels that are
 - context-sensitive (but mostly local!)
 - complex (sub/ir/regularity), pockets of exceptions
- Similar representations at one linguistic level correspond to similar representations at the other level
- Several information sources interact in (often) unpredictable ways at the same level
- Data is sparse

... fit the bias of MBL

- The mappings can be represented as (cascades of) *classification* tasks (*disambiguation* or *segmentation*)
- Locality is implemented through windowing over representations
- Inference is based on Similarity-Based Reasoning
- Adaptive data fusion / relevance assignment is available through feature weighting
- It is a non-parametric approach
- Similarity-based smoothing is implicit
- Regularities and subregularities / exceptions can be modeled uniformly

From POS tagging to IE

- POS tagging (Daelemans et al. 1996)
 - Time/NN flies/VBZ like/RB an/DT arrow/NN.
- NP chunking, classification approach: (Ramshaw & Marcus 1995):
 - Label each word with an NP-tag: I, O, B
e.g. The/I man/I gives/O Mary/I a/B book/I.
- Shallow Parsing (Buchholz et al.; 1999)
 - [NP-SUBJ-1 Time/NN] [VP-1 flies/VBZ] [ADVP like/RB
[NP an/DT arrow/NN]]
- Semantic Tagging = Information Extraction (Zavrel et al.)
 - [PersonalSection [leftNameContext name/NN :/PUNC]
[Name Pascal/NNP Tamino/NNP] [rightNameContext
:/PUNC BLANK/IGN]]

Memory-Based GR labeling

(Buchholz 2002)

Assigning labeled Grammatical Relation links
between words in a sentence:

GR's of Focus with Verbs (subject, object,
location, none)

Features

- **Focus:**
prep, adv-func, word₊₁, word₀, word₋₁, word₋₂,
POS₊₁, POS₀, POS₋₁, POS₋₂, Chunk₊₁, Chunk₀,
Chunk₋₁, Chunk₋₂.
- **Verb:**
POS, word
- **Distance:**
words, VPs, comma's
- **Class:**
GRtype

Results WSJ

Buchholz, Veenstra, Daelemans, 1999, EMNLP

- Useful for Question Extraction/Answering, IE
etc.

Does MBL have the right bias for NLP ?

The theoretical problem

- “No free lunch” theorems (Wolpert)
 - Problem of induction (Hume, 1748)
 - no inductive algorithm is universally better than any other; generalization performance of any inductive algorithm is zero when averaged over a uniform distribution of all possible classification problems (i.e. assuming a random universe)
- A posteriori justification *is* possible: what we can conclude from the empirical success of different inductive algorithms about the (probably) non-random universe
 - Comparative Machine Learning experiments

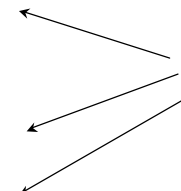
Why Comparative ML experiments in NLP ?

- Evaluate bias of ML method for some NLP task
- Evaluate the role of different information sources in solving a ML of NL task
- Examples:
 - EMNLP, CoNLL, ACL, ...
 - Competitions:
 - SENSEVAL
 - CoNLL shared tasks
 - TREC / MUC / DUC / ...

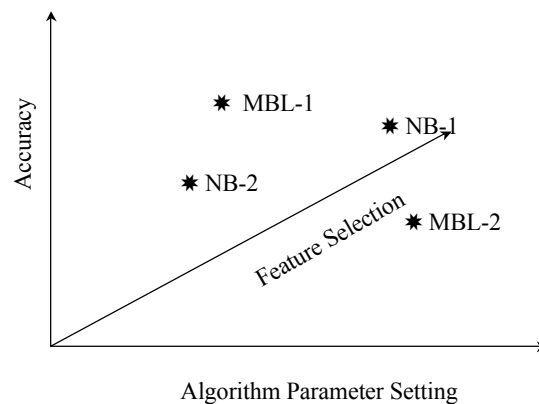
Example: WSD

- Mooney, EMNLP (1996)
 - NB & perceptron > DL > MBL
- Escudero, Marquez, & Rigau, ECAI (2000)
 - MBL > NB
- Lee & Ng, EMNLP (2002)
 - Knowledge sources and learning algorithms interact
 - 4 knowledge sources better than 1
 - SVM > Adb, NB, DT

What influences a ML experiment?

- Information sources
 - feature selection
 - feature representation
 - Algorithm parameters
 - Training data
 - sample selection
 - sample size (Banko & Brill 2001)
 - Combination methods
 - bagging, boosting
 - output coding
- + interactions
- 

Accuracy Landscapes



Current Practice Comparative ML Experiments

- Use default algorithm parameters
- Sometimes: algorithm parameter optimization
- Sometimes: feature selection
- Never: combined feature selection and parameter optimization
= combinatorial optimization problem
- Methodology: k-fold cross-validation, McNemar, paired t-test, learning curves, etc.

Hypothesis

The variability in accuracy resulting from interactions of algorithm parameter settings and feature selection is higher than the accuracy difference between two algorithms given constant input features and default algorithm parameter settings.

Therefore: many published comparative machine learning experiment results (and their interpretation) are not reliable.

Experiment 1

(with Veronique Hoste)

- Investigate the effect of
 - algorithm parameter optimization
 - feature selection (forward selection)
 - interleaved feature selection and parameter optimization
- ... on the comparison of two inductive algorithms (lazy and eager)
- ... for a selection of NLP task datasets
 - Word Sense Disambiguation, tagging known words and unknown words, diminutive morphology

Algorithms compared

- Ripper
 - *Cohen, 95*
 - Rule Induction
 - Algorithm parameters: different class ordering principles; negative conditions or not; loss ratio values; cover parameter values
- TiMBL
 - *Daelemans/Zavrel/van der Sloot/van den Bosch, 99*
 - Memory-Based Learning
 - Algorithm parameters: ib1, igtrees; overlap, mvdm; 5 feature weighting methods; 4 distance weighting methods; 10 values of k

WSD (line)

Similar: little, make, then, time

	Ripper	TiMBL
Default	21.8	20.2
Optimized parameters	22.6	27.3
Optimized features	20.2	34.4
Optimized parameters + FS	33.9	38.6

Diminutive

	Ripper	TiMBL
Default	96.3	96.0
Optimized parameters	97.3	97.8
Optimized features	96.7	97.2
Optimized parameters + FS	97.6	97.9

Tagging (known-unknown)

	Ripper	TiMBL
Default	93.1 - 76.1	93.0 - 76.3
Optimized parameters	93.9 - 78.1	95.2 - 82.2
Optimized features	93.3 - 76.3	95.0 - 76.5
Optimized parameters + FS	94.5 - 78.1	96.5 - 82.2

Generalizations?

- In general, best features or best parameter settings are unpredictable for a particular task and for a particular ML algorithm
- Accuracy landscape is not well-behaved

Experiment 2

(with Veronique Hoste)

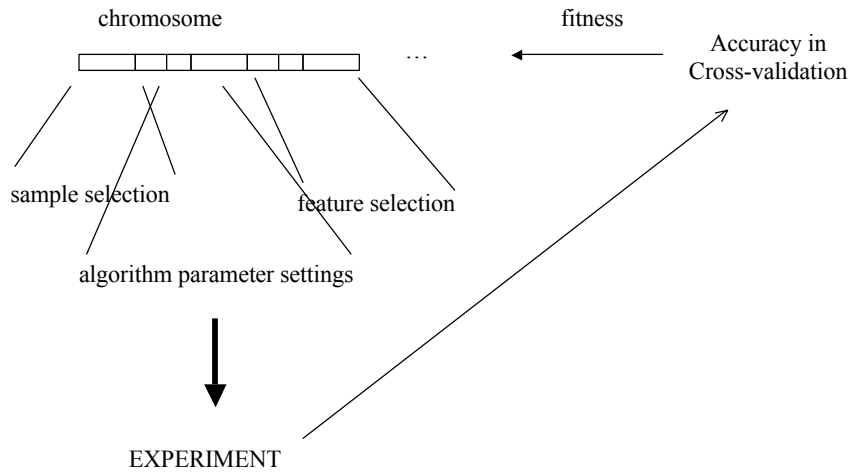
- Investigate the effect of
 - algorithm parameter optimization
- ... on the comparison of different knowledge sources for one inductive algorithm (TiMBL)
- ... for WSD
 - Local context
 - Local context and keywords

TiMBL-WSD (do)

Similar: experience, material, say, then

	Local Context	+ keywords
Default	49.0	47.9
Optimized parameters LC	60.8	59.5
Optimized parameters	60.8	61.0

Genetic Algorithms ?



Conclusion

- MBL seems to have the right bias for NLP tasks
- However, it is hard to show this empirically
 - Optimizing algorithm parameter setting and feature selection interaction has a huge effect on generalization accuracy and on the comparison of classifiers and information sources
- For many problems and algorithms, this optimization is computationally not feasible
- Current research: optimization using GAs