

Machine Learning of Natural Language

Walter Daelemans
CNTS Language Technology Group
Department of Linguistics
University of Antwerp
Belgium
walter.daelemans@ua.ac.be

Antal van den Bosch
ILK Research Group
Dept. of Language and Information Science
Tilburg University
The Netherlands
antal.vdnbosch@uvt.nl

ECML-PKDD-2006
Tutorial
Berlin, Friday September 22, 2006

Tutorial overview

1. Machine Learning ↔ Natural Language Processing

- State of the art in NLP
- The marriage with ML
- ML driven by NLP

2. Language data challenges

- The curse of modularity
- Very very large corpora
- Zipf and Dirichlet

3. Issues in ML of NL

- Eager - Lazy dimension; Forgetting exceptions is harmful
- Methodological problems
- Search in features x algorithm space

Machine Learning \leftrightarrow Natural Language Processing

- State of the art in NLP
 - Why NLP and language technology?
 - Typical modules
 - Typical applications
- The marriage with ML
- ML driven by NLP

Why Natural Language Processing?

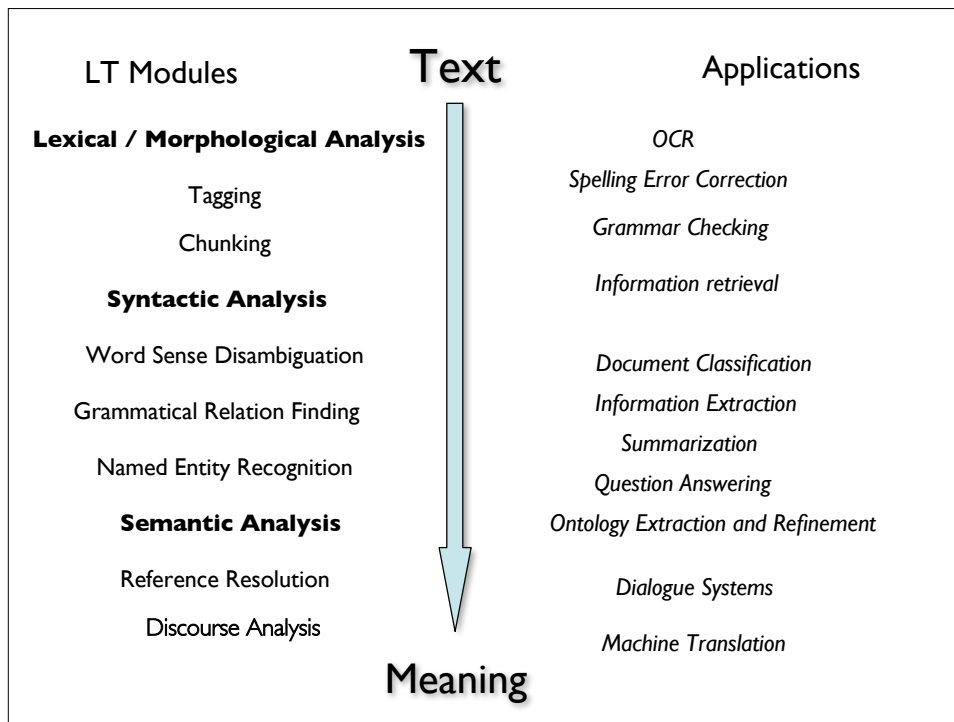
- Language
 - is the main medium for knowledge representation and knowledge transfer
- Access to unstructured and semi-structured data (text, text fields in databases, ...)
 - is a potential goldmine for knowledge discovery and decision making
 - Solving the information overload bottleneck
 - Solving the multilinguality bottleneck

Something to think about:

- Estimates by the Gartner Group
 - predict that unstructured information doubles every three months (!?)
 - The amount of *new* information doubles every year
- *Even if exaggerated, this means that soon nobody will have a complete overview of anything important*

Natural Language Processing

- Applications
 - Question Answering, Information Extraction, Ontology learning from text, ...
- Modules and Data
 - Tagging, parsing, semantic role assignment, word sense disambiguation, ...
 - Computational lexicons, (annotated) corpora (with annotation protocols), ...
- Real Applications
 - Human Resources, Enterprise Information Management, Biomedical data mining and visualization, Intelligence, ...
 - Includes (invisible) embedded NLP applications and modules



Machine Learning in NLP

- To train modules
 - E.g. parser, WSD-module, ...
- To construct / adapt LT applications
 - E.g. Information Extraction rules, statistical Machine Translation (SMT)
- To achieve Text Data Mining
- To model language acquisition

```

<!DOCTYPE MBSP SYSTEM 'mbsp.dtd'>
<MBSP>
<S cnt="s1">
  <NP rel="SBJ" of="s1_1">
    <W pos="DT">The</W>
    <W pos="NN" sem="cell_line">mouse</W>
    <W pos="NN" sem="cell_line">lymphoma</W>
    <W pos="NN">assay</W>
  </NP>
  <W pos="openparen">(</W>
  <NP>
    <W pos="NN" sem="cell_line">MLA</W>
  </NP>
  <W pos="closeparen">)</W>
  <VP id="s1_1">
    <W pos="VBG">utilizing</W>
  </VP>
  <NP rel="OBJ" of="s1_1">
    <W pos="DT">the</W>
    <W pos="NN" sem="DNA_part">Tk</W>
    <W pos="NN" sem="DNA_part">gene</W>
  </NP>
  <VP id="s1_2">
    <W pos="VBZ">is</W>
    <W pos="RB">widely</W>
    <W pos="VBN">used</W>
  </VP>
  <VP id="s1_3">
    <W pos="TO">to</W>
    <W pos="VB">identify</W>
  </VP>
  <NP rel="OBJ" of="s1_3">
    <W pos="JJ">chemical</W>
    <W pos="NNS">mutagens</W>
  </NP>
  <W pos="period">.</W>
</S>
</MBSP>

```

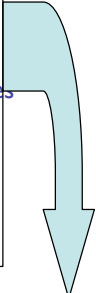
Example: Text Data Mining (Discovery)

- Find relevant information
 - Information extraction
 - Text categorization
- Analyze the text
 - Tagging - Parsing - Named Entity Classification - Semantic Roles, ...
- Discovery of new information
 - Integrate different sources: structured and unstructured
 - Data mining

Don Swanson 1981: medical hypothesis generation

- stress is associated with migraines
- stress can lead to loss of magnesium
- calcium channel blockers prevent some migraines
- magnesium is a natural calcium channel blocker
- spreading cortical depression (SCD) is implicated in some migraines
- high levels of magnesium inhibit SCD
- migraine patients have high platelet aggregability
- magnesium can suppress platelet aggregability
- ...

Text analysis output



Magnesium deficiency implicated in migraine (?)

The move to ML

- Acquisition

OLD: Construct a (rule-based) model about the domain of the transformation vs.

NEW: Induce a stochastic model from a corpus of “examples” of the transformation

- Processing

OLD: Use rule-based reasoning, deduction, on these models to solve new problems in the domain vs.

NEW: Use statistical inference (generalization) from the stochastic model to solve new problems in the domain.

Advantages

Deductive

- Linguistic knowledge and intuition can be used
- Precision

Inductive

- Fast development of model
- Good coverage
- Good robustness (preference statistics)
- Knowledge-poor
- Scalable / Applicable

Problems

Deductive

- Representation of sub/irregularity
- Cost and time of model development
- (Not scalable / applicable)

Inductive

- Sparse data
- Estimation of relevance statistical events
- Understandability

Importance of language data for Machine Learning

- Huge amounts of data
- Data with interesting properties
 - Distributions extraordinaire: Zipf, Dirichlet
 - small disjuncts
- Not a toy problem
- Confrontation with efficiency, scalability, and tractability issues

Tutorial overview

1. Machine Learning \leftrightarrow Natural Language Processing
 - State of the art in NLP
 - The marriage with ML
 - ML driven by NLP
2. Language data challenges
 - The curse of modularity
 - Very very large corpora
 - Zipf and Dirichlet
3. Issues in ML of NL
 - Eager - Lazy dimension; Forgetting exceptions is harmful
 - Methodological problems
 - Search in features x algorithm space

Language data challenges

- The curse of modularity
 - Cascading errors
- Very very large corpora
 - Learning curve experiments
- Zipf and Dirichlet
 - Many words are rare (Zipf)
 - But once seen, they reoccur quickly (Dirichlet)

The curse of modularity (I)

- Modularity: dividing one complex problem in several subproblems
- Adding intermediate representation levels
- Assumption or hope:
 - Complexity of main problem = x
 - Complexity of sub problem a = y
 - Complexity of sub problem b = z
 - $x > y+z$ or not much smaller
- Obvious benefits for non-linear problems:
 - hidden layers in neural nets
 - SVM feature space transforms

The curse of modularity (2)

- **Expert-based** modularizations are often biased by
 - Choice of formalism
 - Strong interpretation of **Minimal description length** principle: smaller theories are better
- Which can cause
 - Spurious ambiguity
 - Sequentiality where parallelism is needed
 - Blocking of information
 - **Cascaded propagation of errors**

The curse of modularity (3)

Imaginary 5-modular system in which error is disjointly added:

module #	performance	
	isolation	cascaded
1	95%	95%
2	95%	90%
3	95%	86%
4	95%	81%
5	95%	77%

The curse of modularity (4)

- Actual 5-modular system for English word pronunciation (Van den Bosch, 1997):
 - Morphological segmentation 94.9%
 - Letter-grapheme conversion 98.6%
 - Grapheme-phoneme conversion 96.3%
 - Syllabification 99.5%
 - Stress assignment 92.0%
 - “worst case” phonemes plus stress 88.5%
- Combined phonemes plus stress:
 - Plain propagation 85.9%
 - Adaptive training 89.4%

The curse of modularity (5)

- 3-modular system for English word pronunciation
 - Morphological segmentation 94.9%
 - Letter-phoneme conversion 96.3%
 - Stress assignment 92.0%
 - “worst case” phonemes plus stress 88.5%
- Combined phonemes plus stress:
 - Plain propagation 90.5%
 - Adaptive training 92.1%

The curse of modularity (6)

- Non-modular system for English word pronunciation:
Direct association of letters to stressed phonemes
 - Straight performance 92.6%
 - (recall: 3 modules, adaptive training 92.1%)
 - (recall: 5 modules, adaptive training 89.4%)
- Modularity is harmful

Case 2: PoS?

- Everyone agrees: POS is an abstraction level in (shallow) parsing

The delegation left without a warning .

DET NOUN VERB PREP DET NOUN PUNC

- Computed in almost every parsing system

Could words replace PoS?

Simple intuition:

- PoS disambiguate *explicitly*
suspect-N vs suspect-V vs suspect-A
- Context words disambiguate *implicitly*
... the suspect ...
... we suspect ...

Case study: setup

- Task that involves PoS
 - Shallow parsing of English
- Select input:
 - use gold-standard POS
 - use words only
 - use both
- Learn the task with increasing amounts of training data
 - which learning curve is higher?
 - which learning curve grows faster?
 - do they meet or cross?

Step 1: get parsed sentence

```
((S (ADVP-TMP Once)
  (NP-SBJ-1 he)
  (VP was
    (VP held
      (NP *-1)
      (PP-TMP for
        (NP three months))
      (PP without
        (S-NOM (NP-SBJ *-1)
          (VP being
            (VP charged)
          )
        )
      )
    )
  )
)) .))
```

Step 2: flatten

```
[ADVP Once-ADVP-TMP]
[NP he-NP-SBJ] [VP was held-VP/S]
[PP for-PP-TMP] [NP three months-NP]
[PP without-PP]
[VP being charged-VP/SNOM]
```

Step 3: make instances

- | | |
|--|-------------------|
| • ... _ Once he ... | I-ADVP - ADVP-TMP |
| • ... Once he was ... | I-NP - NP-SBJ |
| • ... he was held ... | I-VP - NOFUNC |
| • ... was held for ... | I-VP - VP/S |
| • ... held for three ... | I-PP - PP-TMP |
| • ... for three months ... | I-NP - NOFUNC |
| • ... three months without ... | I-NP - NP |
| • ... months without being ... | I-PP - PP |
| • ... without being charged ... | I-VP - NOFUNC |
| • ... being charged | I-VP - VP/S-NOM |
| • ... charged . _ ... | O - NOFUNC |

Case study: details

- experiments based on Penn Treebank (WSJ, Brown, ATIS)
 - 74K sentences, 1,637,268 tokens (instances)
 - 62,472 unique words, 874 chunk-tag codes
- 10-fold cross-validation experiments:
 - Split data 10 times in 90% train and 10% test
 - Grow every training set stepwise
- precision-recall on correct chunkings with correct type tags in test material

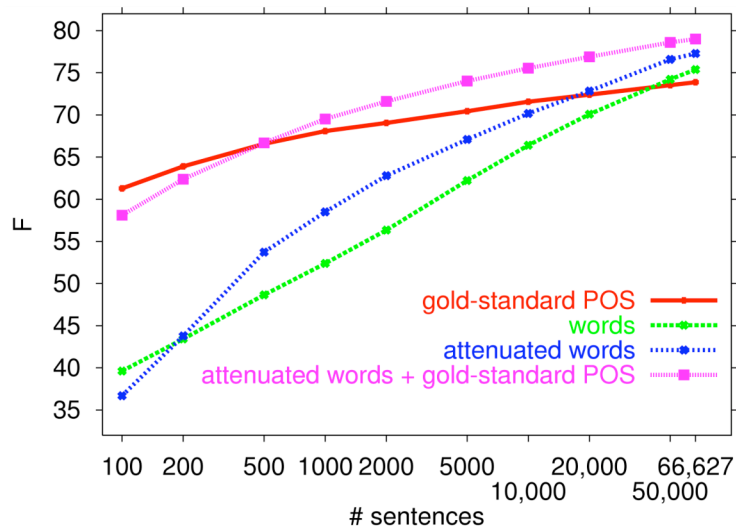
Case study: Extension

- Word attenuation (after Eisner 96):
 - Distrust low-frequency information (<10)
 - But keep whatever is informative (back-off)
 - Convert to MORPH-[CAP|NUM|SHORT|ss]

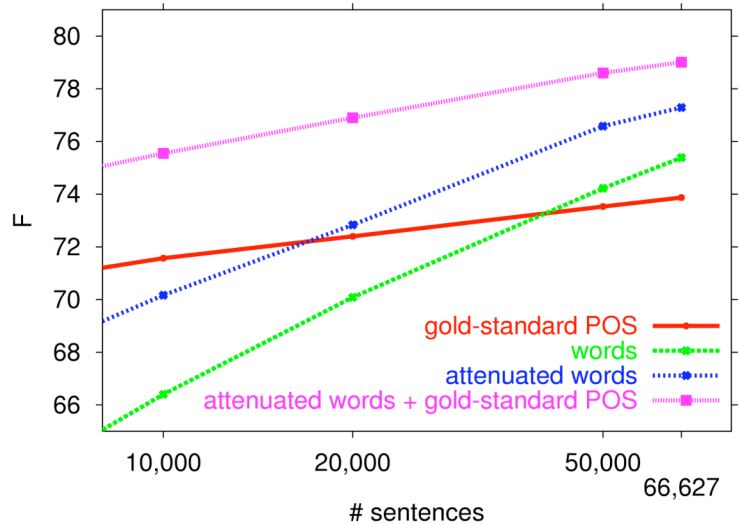
A Daikin executive in charge of exports when the high-purity halogenated hydrocarbon was sold to the Soviets in 1986 received a suspended 10-month jail sentence .

A MORPH-CAP executive in charge of exports when the MORPH-ty MORPH-ed MORPH-on was sold to the Soviets in 1986 received a suspended MORPH-th jail sentence .

Results: learning curves (I)



Results: learning curves (2)



Discussion

- Learning curve experiments
 - (cf. Banko & Brill 01)
 - Important dimension in experimental space next to feature/parameter selection
- More data, better statistics
 - Increased numbers of seen words
 - Improved frequencies of known words
- Given optimal attenuation, explicit PoS information close to unnecessary

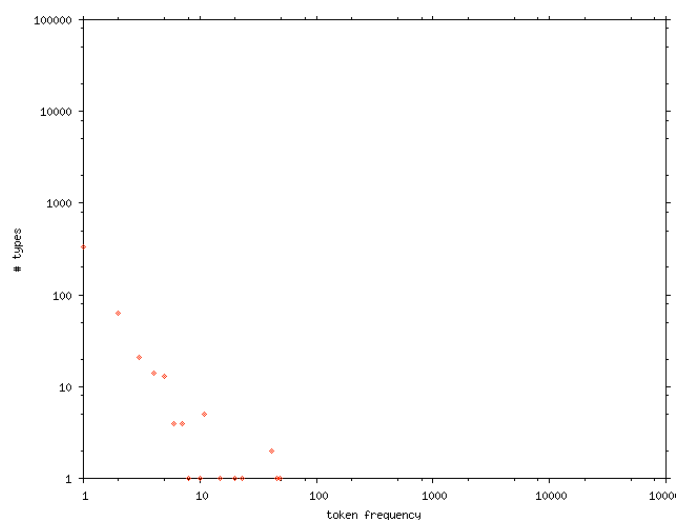
Zipf

- George Kingsley Zipf (1902-1950)
 - The psycho-biology of language (1935)
 - Human behavior and the principle of least effort (1949)
- Zipf's Law
 - Family: power laws (e.g. Pareto distribution)
 - Refinements: Ferrer i Cancho (2005)

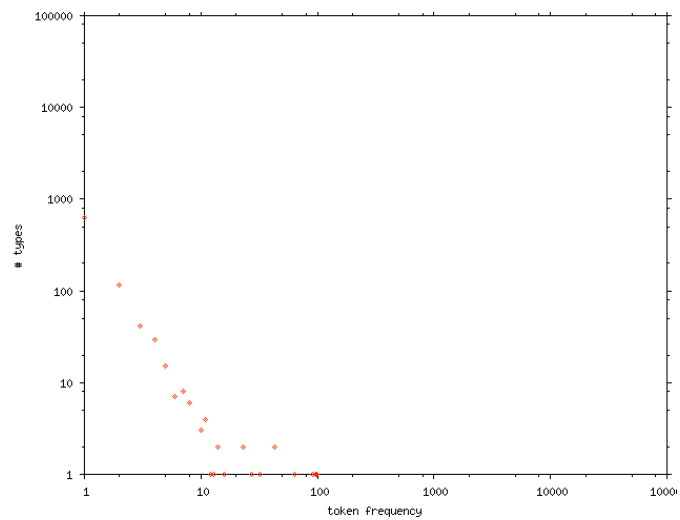
$$P_n \approx 1/n^a$$

- where P_n is the frequency of a word ranked n th and a is almost 1.

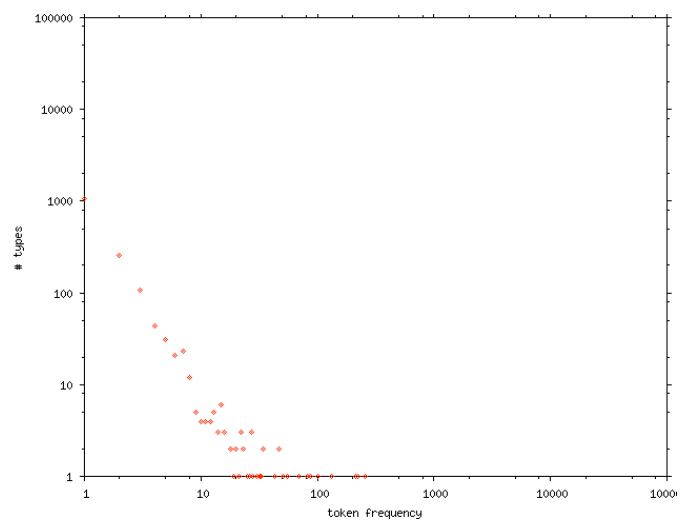
WSJ, first 1,000 words



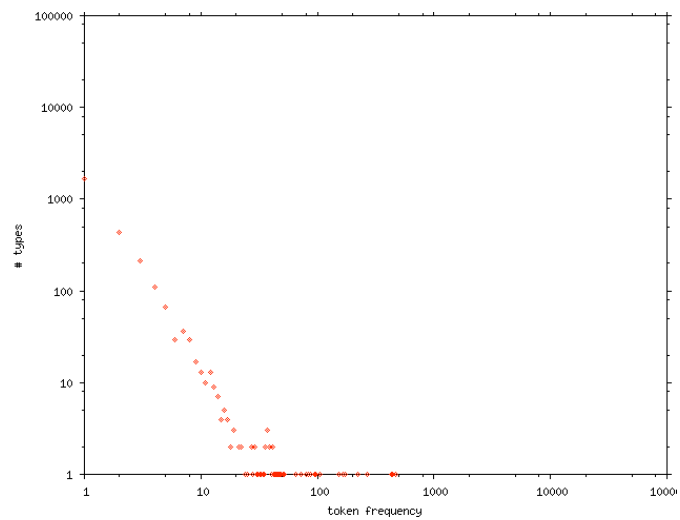
WSJ, first 2,000 words



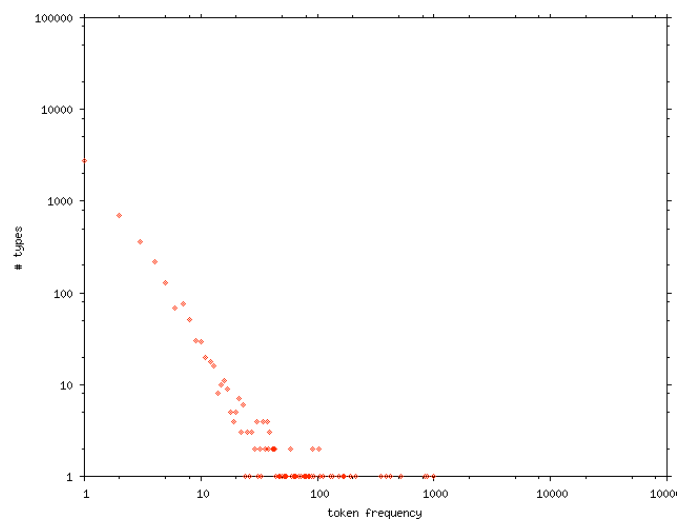
WSJ, first 5,000 words



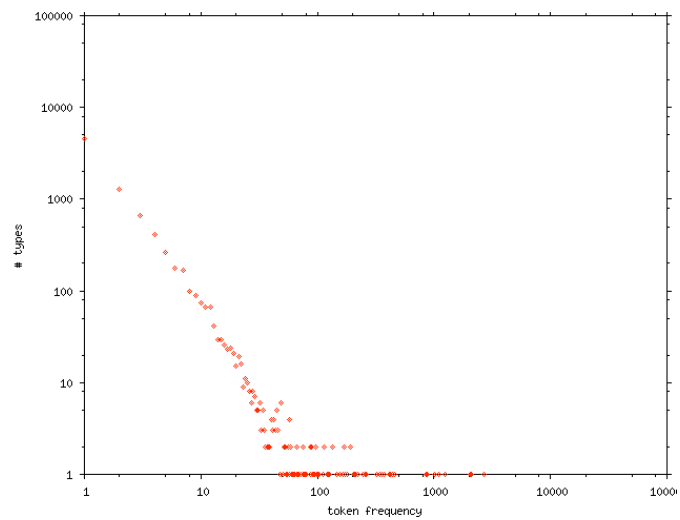
WSJ, first 10,000 words



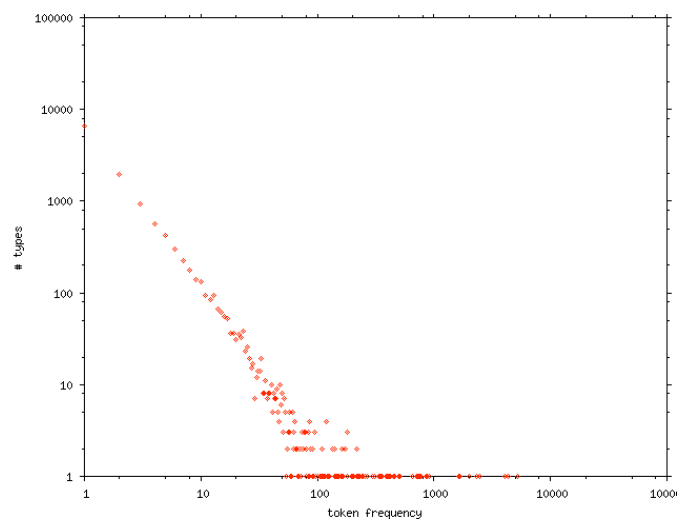
WSJ, first 20,000 words



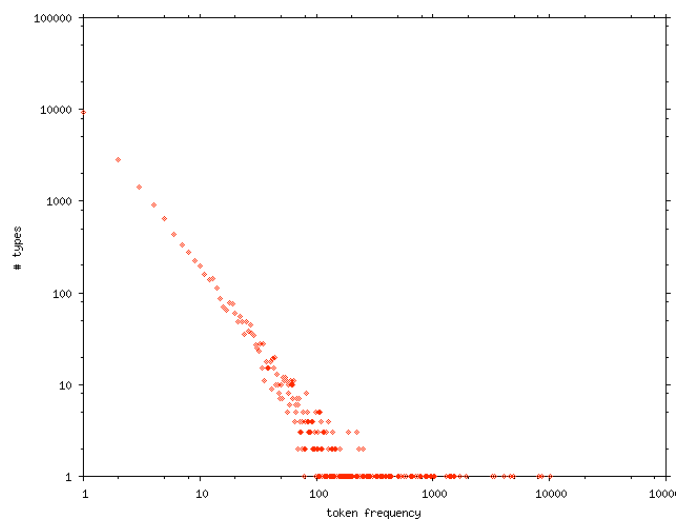
WSJ, first 50,000 words



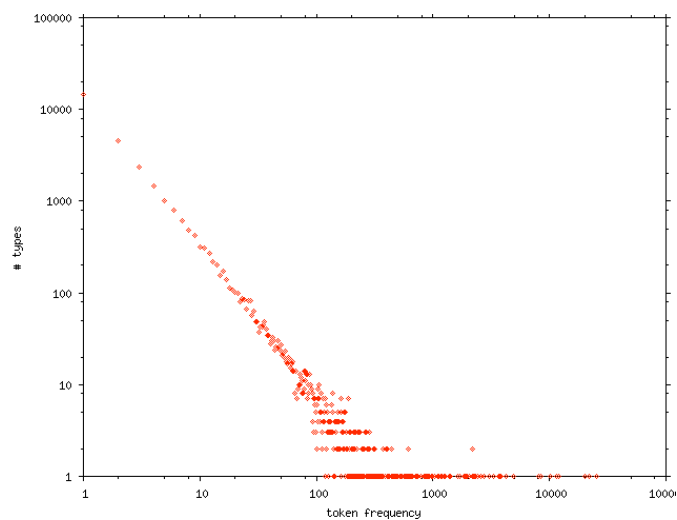
WSJ, first 100,000 words



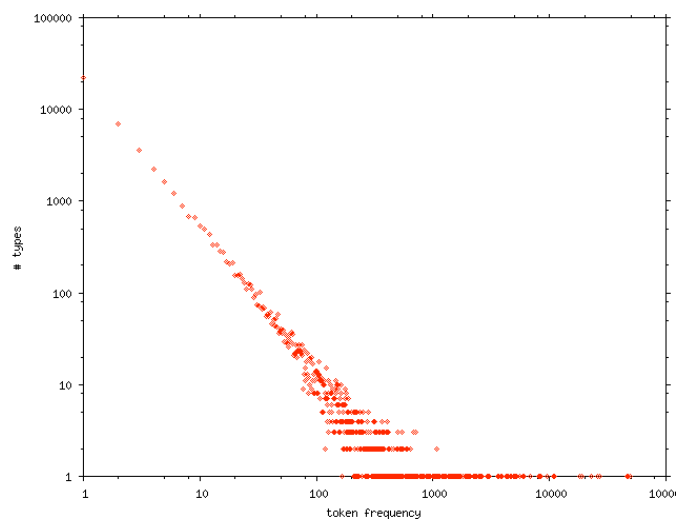
WSJ, first 200,000 words



WSJ, first 500,000 words



WSJ, all 1,126,389 words



Chasing Zipf's tail

- More data brings two benefits:
 - More observations of words already seen.
 - More new words become known (the tail)
- This effect **persists**, no matter how often the data is doubled.
- But, there will always be **unseen** words.

Word prediction

- “Archetypal problem of NLP” (Even-Zohar, Roth, Zelenko, 1999)
- Different from word *completion*
- Predict what?
 - the next word
 - *the missing word*
- The word itself, or a set of possible words, with probabilities

A special problem

- Examples abound in huge quantities
 - n -gram models in language modeling for speech recognition: “there’s no data like more data”
- When viewed as prediction task to be learned by predictors/classifiers,
 - issue of extremely many outcomes,
 - having same (Zipfian) distribution as input,
 - underlying problem is cosmic.

Big numbers

- There's no end to the amount of examples one can gather
 - *millions, billions*
- Each example contains several word positions to the left and right
 - restricted to some local context
- Each position easily carries
 - *thousands to hundreds of thousands* of values (the given context words)
- The prediction is also among
 - *thousands to hundreds of thousands* of values (the word to be predicted)

Some ML algorithms are out

- Support vector machines
 - Complicated handling of multi-class spaces
- Rule learning and decision trees
 - Number of classes and feature values are serious components of learning procedures

Some may be in, but...

- Hyperplane / probabilistic classifiers
 - Maximum-entropy models, MEMM, CRF
 - Perceptron, Winnow
- But:
 - Classes and feature values are
 - the dimensions of the probability matrices
 - the layers in the network architecture
 - Training (convergence) will be very slow

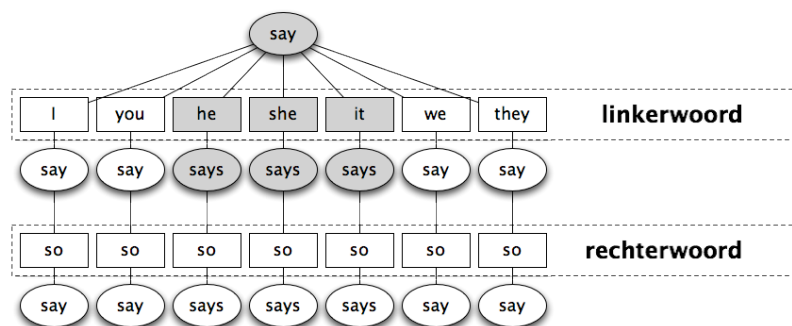
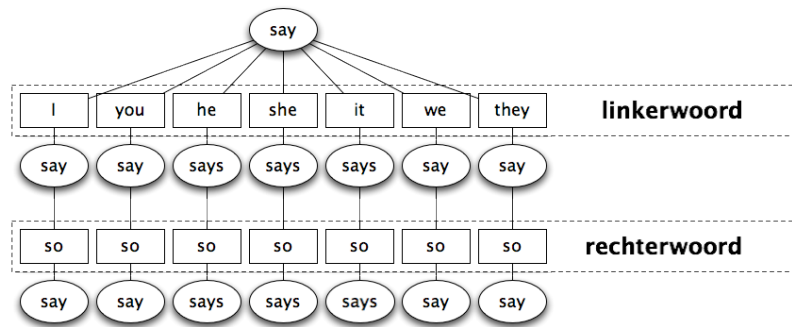
Insensitive to # classes

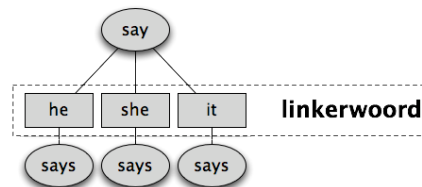
- n -gram models
 - As known from decades of research
- k -Nearest neighbor classification!
 - Totally impervious to number of classes
 - Learning (=storing): $O(n*f)$
 - **But** classification also $O(n*f)$
 - where n = # examples, f = # features + 1 class

Fast approximation of k -NN

- IGTree (Daelemans, Van den Bosch, and Weijters, 1997)
 - Stores instances in *trie* (Knuth, 1973)
 - = decision tree, with fixed order of feature tests (i.e. much simpler than C4.5)
- Efficient and still impervious to number of classes:
 - Storage: $O(n)$ (k-NN: $O(n*f)$)
 - Learning: $O(n \lg(v) f)$ (k-NN: $O(n*f)$)
 - Classification: $O(f \lg(v))$ (k-NN: $O(n*f)$)
 - v : average # arcs fanning out of nodes
 - here: n very large, f small, $\lg(v)$ small

I say so
you say so
he says so
she says so
it says so
we say so
they say so





Means: predict *say*, unless left word is *he*, *she*, or *it*; in that case predict *says*

Relation with n -gram models

- Close relation to back-off smoothing in n -gram models (used for discrete prediction)
 - “Similarity for smoothing”, Zavrel and Daelemans, ACL 1997
 - Use estimates from more general patterns if specific patterns are absent
- n -gram language models
 - generate probability distributions typically measured through *perplexity*
- IGTtree
 - is simply quick, compressed access to discrete outcomes of back-off smoothed n -gram models

Data

Data set	Source	Genre	Number of tokens
Train-Reuters	Reuters Corpus Volume I	newswire	130,396,703
Test-Reuters	Reuters Corpus Volume I	newswire	100,000
Alice	Alice's Adventures in Wonderland	fiction	33,361
Brown	Brown (Penn Treebank)	mixed	453,446

Task

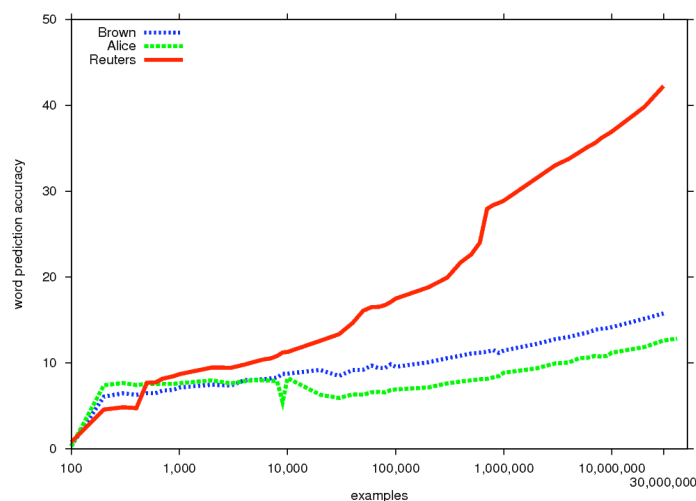
- Predict middle word in local context of
 - 7 words to the left
 - 7 words to the right

...
once or twice she had peeped into ? book her sister was reading , but the
or twice she had peeped into the ? her sister was reading , but it book
twice she had peeped into the book ? sister was reading , but it had her
she had peeped into the book her ? was reading , but it had no sister
had peeped into the book her sister ? reading , but it had no pictures was
...

Experiments

- Learning curve
 - Increasing amount of learning material
 - Direction: from recent to past
- What to expect?
 - Constant improvement with doubling of training examples (Banko and Brill, 2001)?

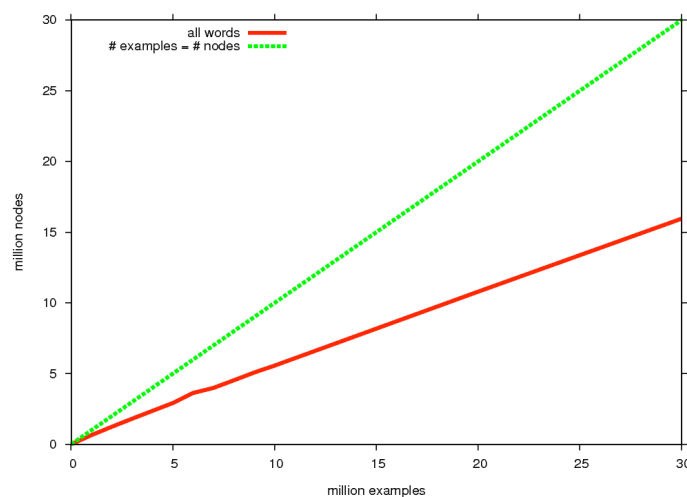
Word prediction accuracy



Results

- Same-genre train-test combination works best (*trivial result*)
 - 42.2% accuracy on Reuters at 30M training examples
 - 12.6% on Alice, 15.6% on Brown
- Roughly log-linear increase
 - On Reuters, every 10-fold yields an extra 8%

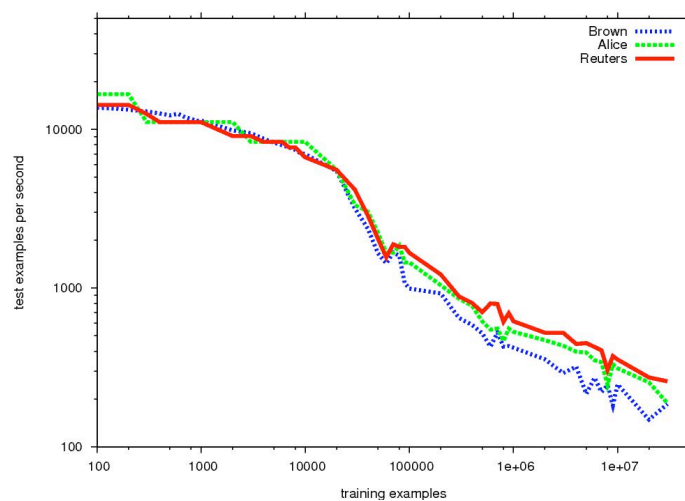
Numbers of nodes



Numbers of nodes

- Less than one node added per example
- At 30M training examples, 16M nodes
- With 20 bytes per node, only 305 Mb

Speed

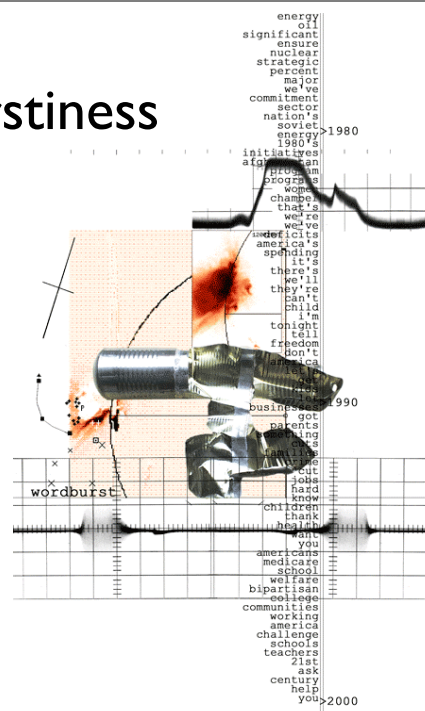


Speed

- Average branching factor (f in $O(f \lg(v))$) is 3.27 at 30M examples; f is 14
- Indeed, classification speeds are OK
 - Still over a hundred predictions per second with largest tree
 - No exponential slowdown
 - Roughly same speed with different genres

Word burstiness

- Even if a word is rare, if it occurs, it tends to re-occur for a while
- Hype words
 - Sign o' the times
 - News
- Domain, register, genre, author, ...



Dirichlet distribution

- Johann P.G. Lejeune Dirichlet (1805-1859)
- Conjugate prior of parameters of multinomial distribution
 - discrete distribution giving the probability of choosing a given collection of m items from a set of n items with repetitions
- Can model word burstiness
 - Applicable to words (Madsen, Kauchak, & Elkan, ICML 2005)
 - Latent Dirichlet Allocation (LDA) (Blei, Ng, & Jordan, JMLR 2003; Blei & Lafferty, ICML 2006)

LDA and Dynamic Topic Modeling (Blei *et al.*)

- LDA: Generative probabilistic model of a natural language corpus
 - Context: text classification, information retrieval, topic segmentation
- Documents are
 - random mixtures over latent topics
 - Following Dirichlet distribution
 - each topic is characterized by a distribution over words

One more example: POS Tagging

- Morphosyntactic disambiguation based on
 - Properties of the word (attested POS tags in training corpus, properties of wordform)
 - Properties of local context
- First step in many text analysis systems
- Considered solved (at 97% accuracy on Penn Treebank)
- But:
 - Bad transfer to other text genres and corpora
 - Essential disambiguation often rather at the 90% level or even below

POS Tagging

	Brown	WSJ	Genia
Brown	96.0	94.8	92.9

A lesson to keep in mind when comparing ML systems

- What is the relevance of a 0.5 to 1% increase in accuracy on a specific dataset when accuracy drops > 3% when moving to a different corpus?
 - As a motivation to use a particular ML algorithm?
 - As a motivation to use a particular representation (feature set)?

Tutorial overview

1. Machine Learning ↔ Natural Language Processing

- State of the art in NLP
- The marriage with ML
- ML driven by NLP

2. Language data challenges

- The curse of modularity
- Very very large corpora
- Zipf and Dirichlet

3. Issues in ML of NL

- Eager - Lazy dimension; Forgetting exceptions is harmful
- Methodological problems
- Search in features x algorithm space

Empirical ML: 2 Flavors

- **Eager**
 - Learning
 - abstract model from data
 - Classification
 - apply abstracted model to new data
- **Lazy**
 - Learning
 - store data in memory
 - Classification
 - compare new data to data in memory

Eager vs Lazy Learning

Eager:

- Decision tree induction
 - CART, C4.5
- Rule induction
 - CN2, Ripper
- Hyperplane discriminators
 - Winnow, perceptron, backprop, SVM
- Probabilistic
 - Naïve Bayes, maximum entropy, HMM

Lazy:

- *k*-Nearest Neighbour
 - MBL, AM
 - Local regression

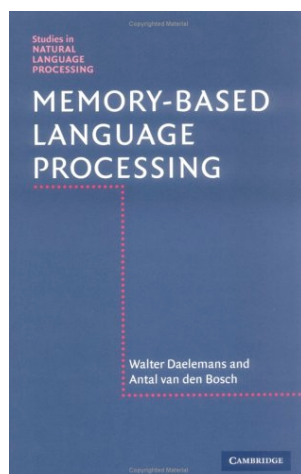
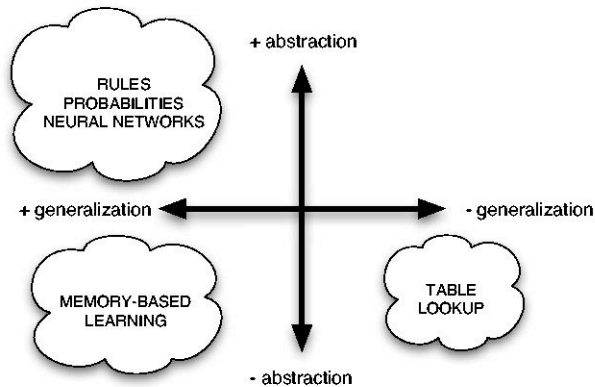
Eager vs Lazy Learning

- **Decision trees**
 - keep the smallest amount of informative decision boundaries (in the spirit of MDL, Rissanen, 1983)
- **Rule induction**
 - keeps smallest number of rules with highest coverage and accuracy (MDL)
- **Hyperplane discriminators**
 - keep just one hyperplane (or vectors that support it)
- **Probabilistic classifiers**
 - convert data to probability matrices
- **k-NN**
 - retains every piece of information available at training time

Eager vs Lazy Learning

- **Minimal Description Length principle:**
 - Ockham's razor
 - Length of abstracted model (covering core)
 - Length of productive exceptions not covered by core (periphery)
 - Sum of sizes of both should be minimal
 - More minimal models are better
- **“Learning = compression” dogma**
- **In ML, length of abstracted model has been focus; not storing periphery**

Eager vs Lazy Learning



- K-nn (also for symbolic features)
- Adaptive similarity metrics
 - Feature and exemplar weighting
 - Value clustering
- Similarity-based smoothing

Properties of NLP tasks ...

- In a mapping between linguistic levels,
 - similar representations at one level correspond to similar representations at the other level
- Zipfian and Dirichlet distributions (burstiness)
 - Sparse data
- Complex interaction of
 - (sub)regularities and exceptions (high disjunctivity, polymorphic concepts, pockets of exceptions)

... fit the bias of MBL

- Similarity-based reasoning
- Uniform modeling of regular / irregular / exceptional
 - keeps all data
- Similarity-based smoothing
- No assumptions about global distributions
 - local learning

History memory-based approach

- Statistical pattern recognition: rule of nearest neighbor 1-NN, k-NN

This “rule of nearest neighbor” has considerable elementary intuitive appeal and probably corresponds to practice in many situations. For example, it is possible that much medical diagnosis is influenced by the doctor's recollection of the subsequent history of an earlier patient whose symptoms resemble in some way those of the current patient. (Fix and Hodges, 1952, p.43)

Memory-based learning and classification

- Learning:
 - Store instances in memory
- Classification:
 - Given new test instance X ,
 - Compare it to all memory instances
 - Compute a distance between X and memory instance Y
 - Update the top k of closest instances (nearest neighbors)
 - When done, take the majority class of the k nearest neighbors as the class of X

Similarity / distance

- Distance determined by
 - Feature weighting (Information Gain, Gain Ratio, Chi Square, Shared Variance, ...)
 - Value weighting (mvdm)
 - Exemplar weighting
 - Distance-weighted class voting

The MVDM distance function

- Estimate a numeric “distance” between pairs of values
 - “e” is more like “i” than like “p” in a phonetic task
 - “book” is more like “document” than like “the” in a parsing task
 - “NNP” is more like “NN” than like VBD in a tagging task

The MVDM distance function

$$\Delta(X,Y) = \sum_{i=1}^n w_i \delta(x_i, y_i)$$

$$\delta(x_i, y_i) = \sum_{j=1}^n |P(C_j | x_i) - P(C_j | y_i)|$$

Distance weighted class voting

- Increasing the value of k is similar to smoothing
- Subtle extension: making more distant neighbors count less in the class vote
 - Linear inverse of distance (w.r.t. max)
 - Inverse of distance
 - Exponential decay

Comparative experiments

- **Ripper**
 - *Cohen, 95*
 - Rule Induction
 - Algorithm parameters: different class ordering principles; negative conditions or not; loss ratio values; cover parameter values
- **TiMBL**
 - *Daelemans/Zavrell/van der Sloot/van den Bosch, 98*
 - Memory-Based Learning
 - Algorithm parameters: overlap, mvdm; 5 feature weighting methods; 4 distance weighting methods; values of k

Datasets

- **GPLURAL**
 - Formation of plural of German nouns
 - Kind (child) → Kind-er
 - Example = word
 - Features
 - Syllable structure of last two syllables
 - Gender
 - Class
 - 8 plural formation types (includes Umlaut)
 - 50%-50% split train - test

Datasets

- DIMIN

- Formation of diminutive of Dutch nouns
- ring (ring) → ringetje
- Example = word
- Features
 - Syllable structure of last three syllables (with stress marker)
- Class
 - 5 diminutive formation types
- 90%-10% split train - test

Datasets

- MORPH

- Morphological analysis of Dutch words
- [abnormaal]_A [iteit]_{A_→N} [en]_{plural}
(abnormalities)
- Example = window over word
- Features
 - Spelling symbols
- Class
 - 3831 symbols indicating complex segmentation / pos tagging / spelling variation decisions at positions in word
- 90%-10% split train - test

Datasets

- PP
 - English (WSJ), prepositional phrase attachment (Ratnaparkhi data)
 - Eat pizza with sister → V
 - Example = *VP NP PP*
 - Features
 - V N1 P N2
 - Class
 - Binary (V or N attachment)
 - Ratnaparkhi split

Datasets

- CHUNK
 - English (WSJ), chunking data (CoNLL shared task)
 - [He]_{NP} [reckons]_{VP} [the current account deficit]_{NP}
[will narrow]_{VP} [to]_{PP} [only \$ 1.8 billion]_{NP} [in]_{PP}
[September]_{NP} .
 - Example = window of words and tags
 - Features
 - Words and tags
 - Class
 - Extended IOB tag (Inside, Between or Outside XP)
 - CoNLL shared task data

Datasets

- NER
 - English (Reuters) named entity recognition (CoNLL shared task)
 - [U.N.]_{organization} official [Ekeus]_{person} heads for [Baghdad]_{location}.
 - Example = window of words and tags
 - Features
 - Words and tags
 - Class
 - Extended IOB tag (Inside, Between or Outside NER-type)
 - CoNLL shared task data

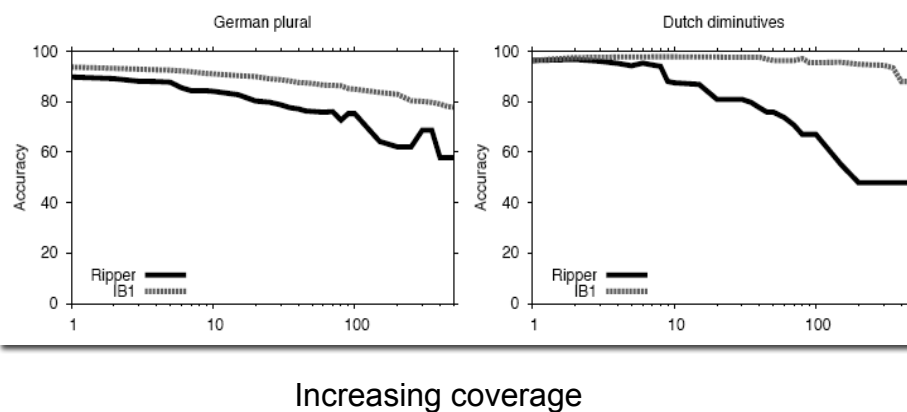
DATA properties

Task	Number of Examples	Number of Features	Range of number of values	Number of classes
GPLURAL	12,584	7	8 – 81	8
DIMIN	2,999	12	2 – 69	5
MORPH	2,888,255	7	49 – 55	3,831
PP	20,801	4	66 – 5,451	2
CHUNK	211,727	14	44 – 19,122	22
NER	203,621	14	45 – 23,623	8

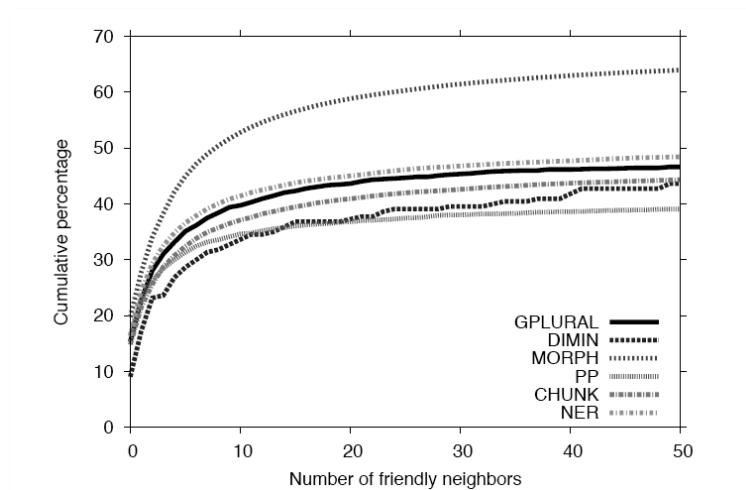
Eager (ripper) never outperforms Lazy (timbl)

Task	Performance metric	Generalization performance (%)		
		IB1	IGTREE	RIPPER
GPLURAL	accuracy	94.0	94.3	91.0
DIMIN	accuracy	97.6	96.6	96.7
MORPH	F-score	70.1	69.9	38.4
PP	accuracy	80.7	76.7	76.1
CHUNK	F-score	91.9	87.6	89.5
NER	F-score	77.2	66.6	55.5

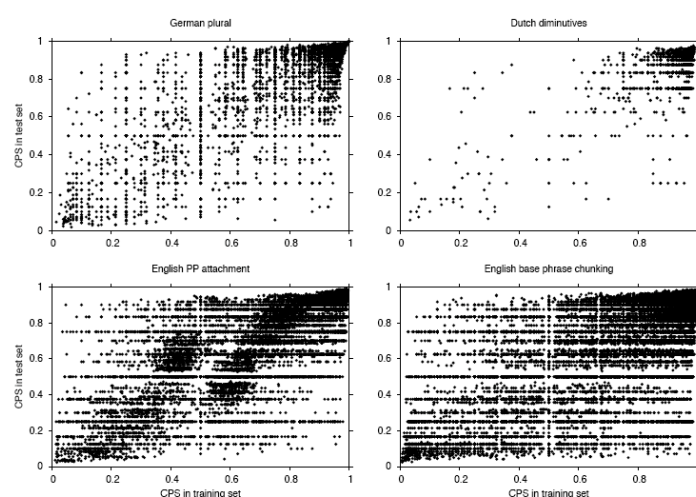
Abstraction hurts



How can we measure disjunctivity?

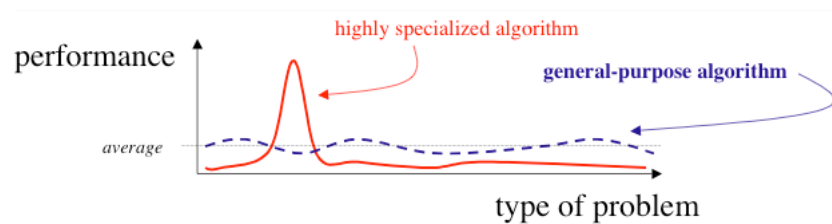


How useful are exceptional examples? (Class Prediction Strength)



Comparative experiments

- “No free lunch” theorems and the problem of induction (Hume)
 - No inductive algorithm is universally better than any other
 - A posteriori justification needed (empirical)



(From Wikipedia)

Methodology

- Evaluate appropriateness of the bias of different ML methods for some NLP task
 - SVM or CRF for NER?
- Evaluate role of different information sources, training data sizes, ...
 - Keywords or only local context for WSD?
- Supported by
 - CoNLL shared tasks, Pascal challenges, NIST competitions etc.
- Often contradictory and unreliable results

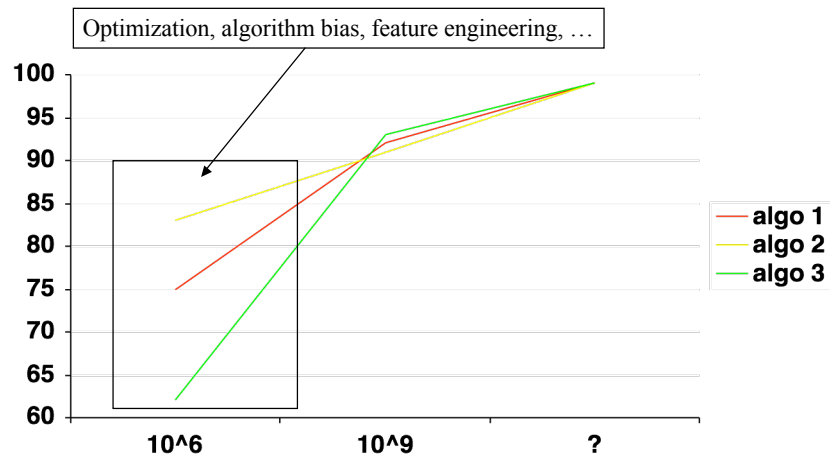
Standard (good) ML methodology

- Cross-validation
- Confusion matrix based evaluation metrics
 - accuracy, precision, recall, F-score, ROC, AUC
- Appropriate statistical significance testing
 - McNemar, paired t-tests
- Learning curve experiments
- Feature selection
- Algorithm parameter optimization
- ...

What leads to “reversals” in ML experiments?

- | | |
|--|--|
| <ul style="list-style-type: none">• Information sources<ul style="list-style-type: none">– feature selection– feature representation (data transforms)• Algorithm parameters• Training data<ul style="list-style-type: none">– sample selection– sample size (Banko & Brill, Van den Bosch & Buchholz) | <ul style="list-style-type: none">• Interactions<ul style="list-style-type: none">– Algorithm parameters and sample selection– Algorithm parameters and feature representation– Feature representation and sample selection– Sample size and feature selection– Feature selection and algorithm parameters– ... |
|--|--|

The Eric Brill model



Eager vs. Lazy: Dutch Diminutive: reversal

	Ripper	TiMBL
Default	96.3	96.0
Feature selection	96.7 (-11)	97.2 (-30)
Parameter optimization	97.3 (-27)	97.8 (-45)
Joint	97.6 (-35)	97.9 (-48)

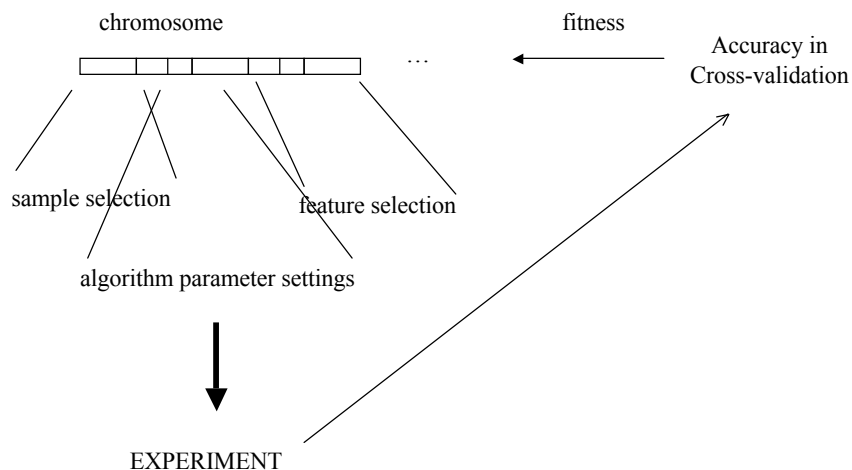
Comparative Methodology problem

- Cannot completely be solved and is also inherent in other fields
 - e.g. psycholinguistics
- More careful methodology is feasible
 - Incorporating more optimization in comparative methodology
 - wrapped progressive sampling (paramsearch) / GA
 - Claims about learning algorithm should be supported by results on many datasets (CoNLL benchmark test)
 - Claims about information sources should be demonstrated with different learning methods

GA for Optimisation in ML of NL

- Use chromosome to encode
 - Algorithm parameters
 - Sample selection
 - Feature selection
- Every individual is an n-fold cv experiment design
- Define fitness of individual in terms of efficiency, feature construction cost, accuracy or a weighted combination of these
 - Don't try this with CRFs (yet) :-)

Genetic Algorithms



Real Case Study: Word Sense Disambiguation

- Decide on the contextually appropriate word sense given local information
 - collocations, keywords, pos tags, syntactic structure, ...
- Supervised ML methods outperform knowledge-based and unsupervised learning approaches
 - Senseval-1, Senseval-2 lexical sample and all-word tasks, different languages
- Which information sources?
- Which machine learning method?

Comparative research

- *Mooney, EMNLP-96*
 - NB & perceptron > DL > MBL ~ Default
 - Only one word, no algorithm parameter optimization, no feature selection, no MBL feature weighting, ...
- *Ng, EMNLP-97*
 - MBL > NB
 - No cross-validation
- *Escudero, Marquez, & Rigau, ECAI-00*
 - MBL > NB
 - No feature selection
- *Escudero, Marquez, Rigau, CoNLL-00*
 - LazyBoosting > NB, MBL, SNoW, DL

Lee & Ng, EMNLP-02

State-of-the-art comparative research
Studies different knowledge sources and different learning algorithms and their interaction
Senseval-1 and senseval-2 data (lexical sample, English)
All knowledge sources better than any 1
SVM > Adb, NB, DT

BUT:

No algorithm parameter optimization
No interleaved feature selection and algorithm parameter optimization

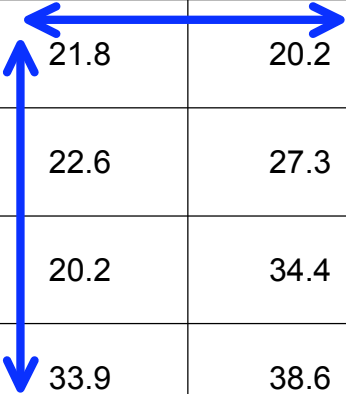
Experiment I

- 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 Word Sense Disambiguation

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



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

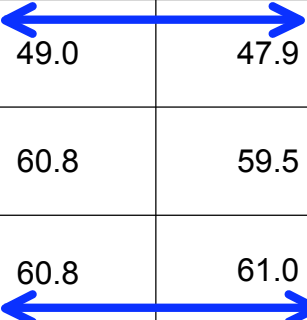
Experiment 2

- 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



Interpretation?

- Exhaustive interleaved algorithm parameter optimization and feature selection is in general computationally intractable
- There seem to be no generally useful heuristics to prune the experimental search space
- In addition, there may be interaction with sample selection, sample size, feature representation, etc.
- Are we taking comparative machine learning experiments too seriously? (*compare results Banko & Brill for large datasets*)

General Conclusions

- The use of inductive techniques in NLP has been a breakthrough toward practical application
 - A lot more progress still needed (semantics)
- Properties of language tasks (extraordinary distributions) have to be acknowledged better
 - Can ML solve problem of adaptation to domains?
- Comparative experiments can help finding learning algorithms with the right bias for language
 - However, comparison is inherently unreliable
 - GAs are useful for optimisation of other ML techniques

More information

- SIGNLL
 - <http://www.aclweb.org/signll>
- Conll shared task datasets
 - <http://ilps.science.uva.nl/~erikt/signll/conll/>
- CNTS
 - <http://www.cnts.ua.ac.be>
- ILK
 - <http://ilk.uvt.nl>