Statistical Methods for Natural Language Processing

Lluís Padró

January 2009

Natural Language Study & Processing

Study of Natural Language

- What kinds of things do people say? (Structure of language)
- What do these things say/ask/request about the world? (Semantics, pragmatics, discourse)
- Traditional Linguistics assumes:
  - People produce gramatical sentences. *Open the radio*
  - People are monolingual adult speakers. (Learning children, dialects, language changes, ...)

Natural Language Study & Processing

Natural Language Processing

- Field of Computer Science devoted to create machines able to communicate in human language (e.g HAL-9000).
- Human language has long been seen as the touchstone of intelligent behaviour (e.g. Turing's Test)
- NLP is said to be *AI-Complete*

Statistical NLP

Broad multidisciplinary area

- Linguistics to provide models of language
- Psychology to provide models of cognitive processes
- Information theory to provide models of communication
- Mathematics & Statistics to provide tools to analyze and acquire such models
- Computer Science to implement computable models

History. Episode I - The beggining

- **1929** Zipf's laws
- **1940-50** Empiricism is a prominent trend in linguistics. Zellig Harris studies co-occurrences
- **1941** Mosteller & Williams establish authorship of the pseudonymous Federalist Papers using word ocurrence patterns
- **1942-45** World War II: A. Turing works on deciphering German codes (i.e. translating to NL). Good-Turing estimation is developed
- **1948** C. Shannon develops Information Theory: probability of a message being chosen, redundancy, error correction, ...
- **1949** W. Weaver proposes to address translation as a particular case of cryptography
- **1957** J.R. Firth: *"You shall know a word by the company it keeps"*

History. Episode II - Chomsky's advent

1957 N. Chomsky (Harris' student) claims that statistical approaches will always suffer from lack of data, and that language should be analyzed at a deeper level.

Colorless green ideas sleep furiously.
Furiously sleep ideas green colorless.

- Even nowadays, sparse data problem is indeed a serious challenge for statistical NLP
- This change of perspective led to new lines of fundamental multidisciplinary research: e.g. Chomsky hierachy, CFG an NFAs are widely used in computer science and compiler development, Lambek, Montague, and others used λ-Calculus to model the semantics of NL

History. Episode III - Resurrection

1970-80 The empiricists strike back
Speech recognition group at IBM sucessfully uses probabilistic models and HMM. Soon they are applied to other NLP tasks. Evidence from psychology shows than human learning may be statistically-based.

1996 F. Jelinek: *"Every time I fire a linguist, performance goes up"*

1996 S. Abney: *"In 1996, no one can profess to be a computational linguist without a passing knowledge of statistical methods. HMM's are as de rigeur as LR tables, and anyone who cannot at least use the terminology persuasively risks being mistaken for kitchen help at the ACL banquet"*

The future is interdiscipinariety

Problems of the traditional approach (1)

- Language Acquisition:
  Children try and discard syntax rules progressively
- Language Change:
  Language changes along time (*ale* vs. *eel*, *while* as Adv vs. Noun, *near* as Prep vs. Adj)
- Language Variation:
  Dialect continuum (e.g. Inuit)
- Language is a collection of statistical distributions:
  Weights for rules (phonetic, syntactic, etc) change when learning, along time, between communities...

Problems of the traditional approach (2)

- Structural ambiguity

  Our company is training workers
  Our problem is training workers
  Our product is training wheels

  Parker saw Mary
  The a are of I

- Robustness: scaling up

  Up from small and domain specific applications

- Practicallity: Time costly to build systems with good coverage
- Brittleness (metaphors, common sense)
- Instance of IA knowledge Representation problem: requires learning

How Statistics helps

- Disambiguation: Stochastic grammars. *John walks*
- Degrees of grammaticality
- Naturalness: *strong tea, powerful car*
- Structural preferences:
  The emergency crews hate most is domestic violence
- Error tolerance:
  We sleeps *Thanks for all you help*
- Learning on the fly:
  One hectare is a hundred ares
  The are a of I
- Lexical Acquisition.

Zipf's Laws (1929)

- Word frequency is inversely proportional to its rank (speaker/hearer minimum effort) $f \sim 1/r$
- Number of senses is proportional to frequency root $m \sim \sqrt{f}$
- Frequency of intervals between repetitions is inversely proportional to the length of the interval $F \sim 1/I$
- Random generated languages satisfy Zipf's laws
- Frequency based approaches are hard, since most words are rare
  - Most common 5% words account for about 50% of a text
  - 90% least common words account for less than 10% of the text
  - Almost half of the words in a text occurr only once

Usual Objections

Stochastic models are for engineers, not for scientists

- Approximation to handle information impractical to collect in cases where initial conditions cannot be exactly determined (e.g. as queue theory models dynamical systems).
- If the system is not deterministic (i.e. has *emergent* properties), an stochastic account is more insightful than a reductionistic approach (e.g. statistical mechanics)

Chomsky's heritage: Statistics can not capture NL structure

- Techniques to estimate probabilities of unseen events.
- Chomsky's criticisms can be applied to Finite State, N-gram or Markov models, but not to all stochastic models.

Conclusions

- Statistical methods are relevant to language acquisition, change, variation, generation and comprehension.
- Pure algebraic methods are inadequate for understanding many important properties of language, such as the measure of goodness that allows to identify the correct parse among a large candidate set.
- The focus of computational linguistics has been up to now on technology, but the same techniques promise progress at unanswered questions about the nature of language.

Probability Theory

Probability Spaces

- Experiment
- Sample space Ω: discrete/continuous
- Partitions and Parts set $\mathcal{P}(\Omega)$, 2^Ω
- Event $A \subseteq \Omega$. Event space: 2^Ω
- Probability function (or distribution): $P(A)$
  $P : 2^\Omega \rightarrow [0,1]$
  $P(\Omega) = 1$
  $P(\bigcup_{j=1}^{\infty} A_j) = \sum_{j=1}^{\infty} P(A_j)$ (disjoint events)

Conditional Probability and Independence

- Prior/posterior probability
  $$P(A \mid B) = \frac{P(A \cap B)}{P(B)}$$
- Independence
  $$P(A) = P(A \mid B) \qquad P(A \cap B) = P(A)P(B);$$
- Conditional independence
  $$P(A \cap B \mid C) = P(A \mid C)P(B \mid C)$$

Conditional Probability and Independence. Example

English to French preposition translation:

$P(a,b)$	dans	en	à	sur	au-cours-de	pendant	selon	
in	4%	10%	15%	0%	8%	3%	0%	40%
on	6%	25%	10%	15%	0%	0%	4%	60%
total	10%	35%	25%	15%	8%	3%	4%	100%

Exercises:

$P(in) = ?$

$P(sur \vee selon) = ?$

$P(sur|in) = ?$

$P(on|en \vee dans) = ?$

Bayes' Theorem

[Bayes 1763]

$$P(B \mid A) = \frac{P(B \cap A)}{P(A)} \quad \longrightarrow \quad P(B \cap A) = P(B \mid A)P(A)$$

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)} \quad \longrightarrow \quad P(A \cap B) = P(A \mid B)P(B)$$

$$P(B \mid A)P(A) = P(A \mid B)P(B)$$

$$P(B \mid A) = \frac{P(A \mid B)P(B)}{P(A)}$$

Bayes' Theorem. Example

Parasitic Gaps: Uncommon phenomenon (1 in 100,000 sentences)

**One can admire Napoleon without particularly liking ___.*
Napoleon is one of those figures one can admire ___ without particularly liking ___.

- Our recognizer correctly detects a *gap* with $p = 0.95$, and incorrectly detects a gap with $p = 0.005$
- Probability that there is a *gap* (G) when the recognizer says so (T):

$$P(G \mid T) = \frac{P(T \mid G)P(G)}{P(T)}$$

Bayes' Theorem. Example

Parasitic Gaps: Uncommon phenomenon (1 in 100,000 sentences)

**One can admire Napoleon without particularly liking ____.*
Napoleon is one of those figures one can admire ____ without particularly liking ____.

- Our recognizer correctly detects a *gap* with $p = 0.95$, and incorrectly detects a gap with $p = 0.005$
- Probability that there is a *gap* (G) when the recognizer says so (T):

$$P(G \mid T) = \frac{P(T \mid G)P(G)}{P(T \cap G) + P(T \cap \neg G)}$$

Bayes' Theorem. Example

Parasitic Gaps: Uncommon phenomenon (1 in 100,000 sentences)

**One can admire Napoleon without particularly liking ___.*
Napoleon is one of those figures one can admire ___ without particularly liking ___.

- Our recognizer correctly detects a *gap* with $p = 0.95$, and incorrectly detects a gap with $p = 0.005$
- Probability that there is a *gap* (G) when the recognizer says so (T):

$$P(G \mid T) = \frac{P(T \mid G)P(G)}{P(T \mid G)P(G) + P(T \mid \neg G)P(\neg G)}$$

Bayes' Theorem. Example

Parasitic Gaps: Uncommon phenomenon (1 in 100,000 sentences)

* *One can admire Napoleon without particularly liking ____.*
Napoleon is one of those figures one can admire ____ without particularly liking ____.

- Our recognizer correctly detects a *gap* with $p = 0.95$, and incorrectly detects a gap with $p = 0.005$
- Probability that there is a *gap* (G) when the recognizer says so (T):

$$P(G \mid T) = \frac{0.95 \times 10^{-5}}{0.95 \times 10^{-5} + 0.05 \times 0.99999} = 0.002$$

Random Variables: Basics

- Random variable: Function on a stochastic process.
  $X : \Omega \longrightarrow \mathcal{R}$
- Continuous and discrete random variables.
- Probability mass (or density) function, Frequency function:
  $p(x) = P(X = x)$.
  Discrete R.V.: $\sum_x p(x) = 1$
  Continuous R.V: $\int_{-\infty}^{\infty} p(x)dx = 1$
- Distribution function: $F(x) = P(X \leq x)$
- Expectation and variance, standard deviation
  $E(X) = \mu = \sum_x xp(x)$
  $VAR(X) = \sigma^2 = E((X - E(X))^2) = \sum_x (x - \mu)^2 p(x)$

Random Variables: Joint and Conditional Distributions

- Joint probability mass function: $p(x, y)$
- Marginal distribution:

$$p_X(x) = \sum_y p(x, y) \qquad p_{X|Y}(x \mid y) = \frac{p(x,y)}{p_Y(y)}$$
$$p_Y(y) = \sum_x p(x, y)$$

Simplified Polynesian. Sequences of C-V syllabes: Two random variables C,V

P(C,V)	p	t	k	
a	1/16	3/8	1/16	1/2
i	1/16	3/16	0	1/4
u	0	3/16	1/16	1/4
	1/8	3/4	1/8	

$P(p \mid i) = ?$
$P(a \mid t \vee k) = ?$
$P(a \vee i \mid p) = ?$

Determining P

- Relative frequency (MLE)
  - Parametric estimation
  - non-parametric (distribution-free) estimation
- Standard distributions. Discrete:
  - Binomial (e.g. tagger accuracy)
  - Multinomial (e.g. zero-gram PoS model)
- Standard distributions. Continuous:
  - Normal (Gaussian distribution)

Samples and Estimators

- Random samples
- Sample variables:

  Sample mean: $\bar{\mu}_n = \frac{1}{n} \sum_{i=1} n x_i$

  Sample variance: $s_n^2 = \frac{1}{n-1} \sum_{i=1} n (x_i - \bar{\mu}_n)^2$.
- Law of Large Numbers: as n increases, $\bar{\mu}_n$ and s_n^2 converge to μ and σ^2
- Estimators: Sample variables used to estimate real parameters.

Finding good estimators: MLE

Maximum Likelihood Estimation (MLE)

- Choose the alternative that maximizes the probability of the observed outcome.
- $\bar{\mu}_n$ is a MLE for $E(X)$
- s_n^2 is a MLE for σ^2
- Data sparseness problem. Smoothing tecnhiques.

$P(a,b)$	dans	en	à	sur	au-cours-de	pendant	selon	
in	0.04	0.10	0.15	0	0.08	0.03	0	0.40
on	0.06	0.25	0.10	0.15	0	0	0.04	0.60
total	0.10	0.35	0.25	0.15	0.08	0.03	0.04	1.0

Finding good estimators: MEE

Maximum Entropy Estimation (MEE)

- Choose the alternative that maximizes the entropy of the obtained distribution

Observations:

$p(en \vee à) = 0.6$

$P(a,b)$	dans	en	à	sur	au-cours-de	pendant	selon	
in	0.04	0.15	0.15	0.04	0.04	0.04	0.04	
on	0.04	0.15	0.15	0.04	0.04	0.04	0.04	
total		$\underbrace{\qquad\qquad}_{0.6}$						1.0

Finding good estimators: MEE

Maximum Entropy Estimation (MEE)

- Choose the alternative that maximizes the entropy of the obtained distribution

Observations:

$p(en \vee à) = 0.6; \qquad p((en \vee à) \wedge in) = 0.4$

$P(a,b)$	dans	en	à	sur	au-cours-de	pendant	selon	
in	0.04	**0.20**	**0.20**	0.04	0.04	0.04	0.04	
on	0.04	0.10	0.10	0.04	0.04	0.04	0.04	
total		$\underbrace{\qquad\qquad}_{\mathbf{0.6}}$						1.0

Finding good estimators: MEE

Maximum Entropy Estimation (MEE)

- Choose the alternative that maximizes the entropy of the obtained distribution

Observations:

$p(en \vee \grave{a}) = 0.6; \qquad p((en \vee \grave{a}) \wedge in) = 0.4; \qquad p(in) = 0.5$

$P(a,b)$	dans	en	à	sur	au-cours-de	pendant	selon	
in	0.02	**0.20**	**0.20**	0.02	0.02	0.02	0.02	**0.5**
on	0.06	0.10	0.10	0.06	0.06	0.06	0.06	
total		$\underbrace{}_{\mathbf{0.6}}$						1.0

Confidence Intervals

- Risk of error in estimation, risk of a biassed sample.
- Given a parameter γ, and two sample variables ν_1 and ν_2, the value $p = P(\nu_1 < \gamma < \nu_2)$ is the degree of confidence for the interval $[\nu_1, \nu_2]$

Theorems and Properties:

- The sum of squares of n Normal RVs follows a χ^2
- Thus, from s_n^2 definition, $\dfrac{(n-1)s_n^2}{\sigma^2} \sim \chi^2(n-1)$
- If $X \sim N(0,1)$, and $Y \sim \chi^2(r)$, then $\dfrac{X}{\sqrt{Y/r}} \sim t(r)$
- etc.

Confidence Intervals

Example: Ratio of nouns per verb in a text

- Sample variable Y: 1.8, 2.2, 1.1, 1.3, 1.6
- Sample mean $\bar{\mu}_n = 1.6$;   Sample variance $s_n^2 = 0.18$
- C.I. at 95% confidence degree, assuming known $\sigma^2 = 0.2$

$$\bar{Y} \sim N(\mu, \sigma/\sqrt{n}); \qquad \frac{\bar{Y} - \mu}{\sigma}\sqrt{n} \sim N(0, 1)$$

Confidence Intervals

That is, we look for a symmetric interval x_1, x_2 such that:

$$P(x_1 < \frac{\bar{Y} - \mu}{\sigma}\sqrt{n} < x_2) = 0.95$$

which is:

$$P(-1.96 < \frac{\bar{Y} - \mu}{\sigma}\sqrt{n} < 1.96) = 0.95$$

so:

$$P(\bar{Y} - 1.96\frac{\sigma}{\sqrt{n}} < \mu < \bar{Y} + 1.96\frac{\sigma}{\sqrt{n}}) = 0.95$$

Thus, the C.I. is:

$$\bar{Y} - 1.96\frac{\sigma}{\sqrt{n}} < \mu < \bar{Y} + 1.96\frac{\sigma}{\sqrt{n}} \quad \Longrightarrow \quad [1.21, 1.99]$$

Confidence Intervals

Example: Ratio of nouns per verb in a text

- Sample variable Y: 1.8, 2.2, 1.1, 1.3, 1.6
- Sample mean $\bar{\mu}_n = 1.6$; Sample variance $s_n^2 = 0.18$
- C.I. at 95% confidence degree, unknown σ^2

$$\frac{\bar{Y} - \mu}{\sigma}\sqrt{n} \sim N(0,1); \qquad \frac{(n-1)s_n^2}{\sigma^2} \sim \chi^2(n-1)$$

Thus,

$$\frac{\bar{Y} - \mu}{s_n}\sqrt{n} \sim t(n-1)$$

Confidence Intervals

That is, we look for a symmetric interval x_1, x_2 such that:

$$P(x_1 < \frac{\bar{Y} - \mu}{s_n}\sqrt{n} < x_2) = 0.95$$

which is:

$$P(-2.57 < \frac{\bar{Y} - \mu}{s_n}\sqrt{n} < 2.57) = 0.95$$

so:

$$P(\bar{Y} - 2.57\frac{s_n}{\sqrt{n}} < \mu < \bar{Y} + 2.57\frac{s_n}{\sqrt{n}}) = 0.95$$

Thus, the C.I. is:

$$\bar{Y} - 2.57\frac{s_n}{\sqrt{n}} < \mu < \bar{Y} + 2.57\frac{s_n}{\sqrt{n}} \quad \Longrightarrow \quad [1.11, 2.09]$$

Hypothesis Testing

- Use the same idea of C.I. to proof/reject hypothesis: We assume the truth of a null hypothesis H_0, that we want to prove false. Then, we compute the probability of the observed sample under that assumption. If it is below certain threshold, we discard the null hypothesis with confidence degree p.
- If we cannot reject H_0, it doesn't mean it's true. Only that we do not have enough evidence to discard it.

Hypothesis Testing

Example: Given one PoS tagger, check if its accuracy is over 96%
Estimated accuracy on a corpus of 1,000 words: $\bar{T} = 0.97$
The accuracy of a tagger is $\bar{T} \sim bin(n, p)$.
For large values of n, we can assume:

$$\frac{(\bar{T} - p)\sqrt{n}}{\sqrt{p(1-p)}} \sim N(0, 1)$$

Our null hypothesis is $H_0 : T \leq 0.96$, we'll try to reject it, computing the probability of the observation under this hypothesis.

Hypothesis Testing

That is, we look for a value x such that:

$$P(\frac{(\bar{T}-p)\sqrt{n}}{\sqrt{p(1-p)}} < x) = 0.95$$

which is: $x = 1.64$
So:

$$P(\bar{T} < p + 1.64\sqrt{\frac{p(1-p)}{n}}) = 0.95$$

under our H_0, $p = 0.96$:

$$P(\bar{T} < 0.96 + 1.64\sqrt{\frac{0.96(1-0.96)}{1,000}}) = P(\bar{T} < 0.9701) = 0.95$$

We cannot reject H_0 (we cannot state that our tagger performs better than 96%)

Hypothesis test on two samples

Example: Given two PoS taggers, check if T_1 is better than T_2
Accuracy on a corpus of $1,000$ words: $\bar{T}_1 = 0.97$; $\bar{T}_2 = 0.96$
$H_0 : T_1 = T_2$

Obs	ok	¬ok	
T_1	970	30	1,000
T_2	960	40	1,000
	1,930	70	2,000

Exp	ok	¬ok	
T_1	965	35	1,000
T_2	965	35	1,000
	1,930	70	2,000

$$\chi^2 = \sum_i \frac{(O_i - E_i)^2}{E_i} = 1.48$$

With 1 d.f. and at 95% confidence, $\chi^2 = 3.84$
Since the obtained value is lower, we cannot reject H_0

Morphology

Morphology

- Deals with the *form* of the words
- Morphological processes
  - Inflection: **[prefixes] + root + suffixes**
    (Root, lemma and form)
  - Derivation:
    Change of category
- Compounds

Grammatical categories, Parts of Speech

- Open categories and Closed (or functional) categories
- Lexicon
- PoS tags

Main Parts of Speech (1)

- Noun
  - Common noun, proper noun
  - Gender, number, case
- Pronoun:
  - Nominative, accusative, possessive, reflexive, interrogative, partitive, ...
  - Anaphora
- Determiner
  - Articles, demonstratives, quantifiers, ...
- Adjective
  - Atributive or adnominal, comparative, superlative, ...

Main Parts of Speech (2)

- Verbs
  - infinitive, gerund, participle
  - number, person, mode, tense (present, past, past perfect, present perfect, future, ...)
  - irregular verbs
  - modal verbs, auxiliary verbs.
- Adverb
  - place, time, manner, degree (qualifiers)
  - Derived / lexical
- Preposition (particles)
- Conjunctions
  - Coordinating, subordinating
- **Agreement**

Syntax and Grammars

- Phrase Structure
  - Word order
  - Syntagma, phrase, constituent
    - NP, VP, AP, head, relative clause, ...
- Grammars
  - Free word order languages. Syntax vs. lexicon
  - Rewrite rules. Context free grammars (CFG):
    - Terminals, no terminals, parse trees...
    - **Recursivity**
  - Bracketing
  - Non-local dependencies
    The women who found the wallet were given a reward.

Structural Ambiguity

- Parse tree $\longrightarrow$ syntactic ambiguity
  - PP-attachment:
    The children ate the cake with a spoon.
    The children ate the cake with a candle.
- Garden paths
  The horse raced past the barn fell.
- Ungrammatical sentences
  **slept children the.*
- Grammatical sentences
  Colorless green ideas sleep furiously.
  The cat barked.

Semantics

- Arguments & Adjuncts
  - Semantic roles: Agent, patient, recipient, instrument, goal
  - Grammatical: Subject, object, indirect object, ...
  - Adjuncts: time, place, manner, ...
  - Active/passive sentences.
- Subcategorization
  - Transitive / intransitive verbs
  - Required/optional Arguments
  - Subcategorization (or diathesis) frames
- Selectional restrictions
  - *The <??> barks*
  - *John eats <??>*

Lexical Semantics

- Relationships between meanings
  - Hypernymy - hiponymy
  - Sinonymy - antonymy
  - Meronymy - holonymy
- Lexical ambiguity
  - Homonymy (bank-bank, *bass-bass*)
  - Homophony (bank-bank, *for-four*)
  - Polysemy (branch-branch)
- Collocations
  - *white hair* *white wine* *white skin*
- Idioms
  - *To pull one's leg* *To kick the bucket*
- Anaphora resolution

Entropy (1)

- Entropy

$$H(p) = H(X) = -\sum_{x \in X} p(x) \log p(x)$$

$$H(X) = \sum_{x \in X} p(x) \log \frac{1}{p(x)} = E(\log \frac{1}{p(X)})$$

- Example: Simplified Polynesian

p 1/8	a 1/4
k 1/8	i 1/8
t 1/4	u 1/8

$$H(P) = -\sum_{i \in \{p,t,k,a,i,u\}} P(i) \log P(i) = 2.5$$

p	t	k	a	i	u
100	00	101	01	110	111

Entropy (2)

- Joint Entropy

$$H(X, Y) = -\sum_{x \in X} \sum_{y \in Y} p(x, y) \log p(x, y)$$

- Conditional Entropy

$$H(X \mid Y) = -\sum_{x \in X} \sum_{y \in Y} p(x, y) \log p(x \mid y)$$

- Chain rule:
$H(X, Y) = H(X) + H(Y \mid X)$
$H(X_1, \ldots, X_n) = H(X_1) + H(X_2 \mid X_1) + \ldots + H(X_n \mid X1, \ldots, X_{n-1})$

Simplified Polynesian Revisited

Sequence of CV syllabes. Two random variables C,V

P(C,V)	p	t	k	
a	1/16	3/8	1/16	1/2
i	1/16	3/16	0	1/4
u	0	3/16	1/16	1/4
	1/8	3/4	1/8	

p	t	k
1/16	3/8	1/16
a	i	u
1/4	1/8	1/8

$$H(C) = -\sum_{c\in\{p,t,k\}} P(c)\log P(c) = -2\frac{1}{8}\log\frac{1}{8} - \frac{3}{4}\log\frac{3}{4} = 1.061$$

$$\begin{aligned} H(V|C) &= -\sum_{c\in\{p,t,k\}} P(C=c)H(V|C=c) = \\ &= \tfrac{1}{8}H(\tfrac{1}{2},\tfrac{1}{2},0) + \tfrac{3}{4}H(\tfrac{1}{2},\tfrac{1}{4},\tfrac{1}{4}) + \tfrac{1}{8}H(\tfrac{1}{2},0,\tfrac{1}{2}) = 1.375 \end{aligned}$$

$$H(C,V) = H(C) + H(V|C) = 2.44 \text{ bits/syllabe} = 1.22 \text{ bits/char}$$

Mutual Information

- Entropy chain rule

$$H(X,Y) = H(X) + H(Y \mid X) = H(Y) + H(X \mid Y)$$

thus, $H(X) - H(X \mid Y) = H(Y) - H(Y \mid X)$, which is defined as $I(X,Y)$.

$$I(X,Y) = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}$$

- Pointwise Mutual Information

$$I(x,y) = \log \frac{p(x,y)}{p(x)p(y)}$$

Entropy of English

- n-gram models (Markov chains)
- Markov assumption (Prob. of a token depends only on the previous k)
- Entropy of English

Model	Cross entropy	
0^{th} order	4.76	(log 27)
1^{st} order	4.03	
2^{nd} order	2.8	
Shannon's experiment	1.34	

Corpus Linguistics. Corpora

- Corpus: Vast sample of language occurrences
- Utility: Corpus linguistics, Statistical NLP.
- Textual corpora, speech corpora (acoustic/transcript)
- Sources
  - LDC, ELRA, ICAME, OTA, etc. (annotated)
  - Newspapers, magazines (raw)
- Criteria
  - Language
  - Genre
  - **Representativeness**. Balanced corpora
- Formatting
  - Markup. Plain *vs.* WYSIWYG
  - Headers, tables, figures... OCR
  - Uppercase/lowercase. Proper nouns, Titles...

Marked up corpora

- Tokens
- Sentences
- Paragraphs
  - Headers, titles, abstracts, ...
- SGML (Standard Generalized Markup Language).
- TEI directives (Text Encoding Initiative)
- XML (eXtensible Markup Language)
- DTD (Document Type Definition)

Marking up Linguistic Information

- PoS Tags.
  - Tag set: *Brown*, *Penn Treebank*, *EAGLES*, self-designed
- Lemmas, stems (IR)
- Syntax
  - Phrase structure, attachments, dependences, ...
- Semantics
  - Word senses, semantic roles, anaphora, correference...
- Markup internal/external to the document

Markup exploitation

- Corpus Linguistics: Evidence for linguistic research.
- NLP
  - Evidence for statistical model estimation
  - Testbench for automatic systems validation

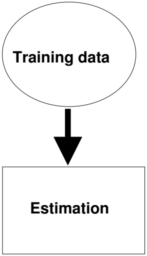
Statistical Models

Goal

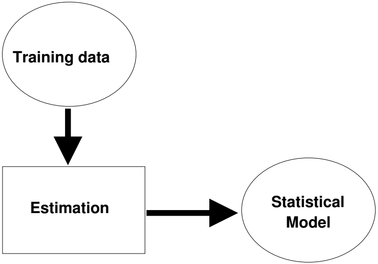
Prediction & Similarity Models

Statistical methods for NLP

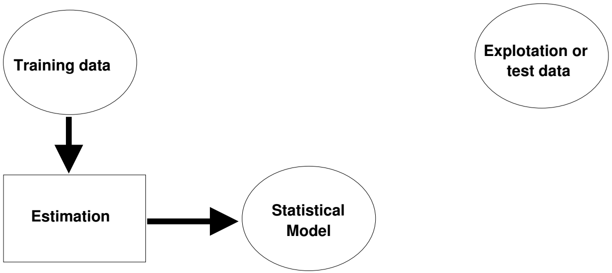
Statistical methods for NLP



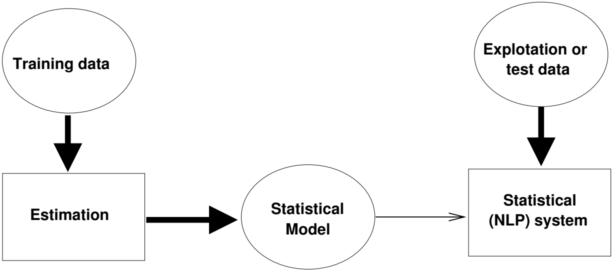
Statistical methods for NLP



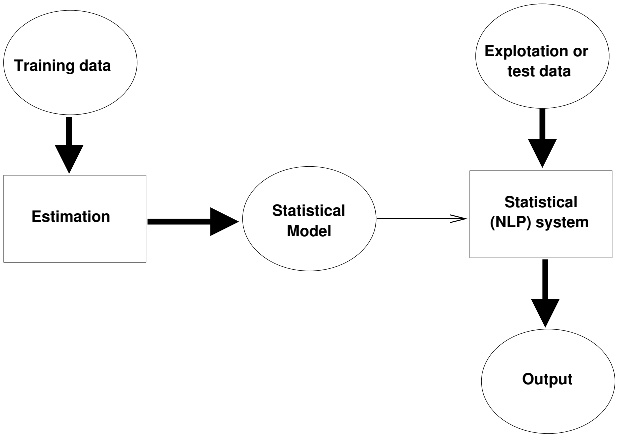
Statistical methods for NLP



Statistical methods for NLP



Statistical methods for NLP



Statistical Models

Goal

Prediction & Similarity Models

Prediction Models & Similarity Models

- Prediction Models: Able to *predict* probabilities of future events, knowing past and present.
- Similarity Models: Able to compute *similarities* between objects (may predict, too).
  - Compare feature-vector/feature-set represented objects.
  - Compare distribution-vector represented objects
  - Used to group objects (clustering, data analysis, pattern discovery, ...)
  - If objects are "present and past" situations, computing similarities may be used as a prediction (memory-based ML techniques).

Prediction Models

Example: Noisy Channel Model (Shannon 48)

NLP Applications

Appl.	Input	Output	$p(i)$	$p(o \mid i)$
MT	L word sequence	M word sequence	$p(L)$	Translation model
OCR	Actual text	Text with mistakes	prob. of language text	model of OCR errors
PoS tagging	PoS tags sequence	word sequence	prob. of PoS sequence	$p(w \mid t)$
Speech recog.	word sequence	speech signal	prob. of word sequence	acoustic model

Similarity Models

Example: Document representation

- Documents are represented as vectors in a high dimensional $\Re^N$ space.
- Dimensions are word forms, lemmas, NEs, ...
- Values may be either binary or real–valued (count, frequency, ...)

$$\vec{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix} \qquad \vec{x}^T = [x_1 \dots x_N] \qquad |\vec{x}| = \sqrt{\sum_{i=1}^{N} x_i^2}$$

Statistical Modeling & Estimation

Inference & Modeling

- Inferring distributions from data
  - Finding good estimators
  - Combining estimators.
- Language Modeling (Shannon game)
- Predictions based on past behaviour
  - Target / classification features $\longrightarrow$ Independence assumptions
  - Equivalence classes (bins). Granularity: discrimination *vs.* statistical reliability

N-gram models

- Predicting the next word in a sequence, given the *history* or *context*. $P(w_n \mid w_1, \ldots, w_{n-1})$
- Markov assumption: Only *local* context (of size $n-1$) is taken into account. $P(w_i \mid w_{i-n+1}, \ldots, w_{i-1})$
- bigrams, trigrams, four-grams ($n = 2, 3, 4$).
  Sue swallowed the large green $<$*?*$>$
- Parameter estimation (number of equivalence classes)
- Parameter reduction via stemming, semantic classes, PoS, ...

Model	Parameters
bigram	$20,000 \times 19,999 \simeq 400 \times 10^6$
trigram	$20,000^2 \times 19,999 \simeq 8 \times 10^{12}$
four-gram	$20,000^3 \times 19,999 \simeq 1,600 \times 10^{15}$

Language model sizes for a 20,000 words vocabulary

Maximum Likelihood Estimation (MLE)

Estimate the probability of the target feature based on observed data. The prediction task can be reduced to having good estimations of the n-gram distribution:

$$P(w_n \mid w_1, \ldots, w_{n-1}) = \frac{P(w_1, \ldots, w_n)}{P(w_1, \ldots, w_{n-1})}$$

- **MLE (Maximum Likelihood Estimation)**
  $P_{MLE}(w_1, \ldots, w_n) = \frac{C(w_1,\ldots,w_n)}{N}$
  $P_{MLE}(w_n \mid w_1, \ldots, w_{n-1}) = \frac{C(w_1,\ldots,w_n)}{C(w_1,\ldots,w_{n-1})}$
  - No probability mass for unseen events
  - Unsuitable for NLP
  - Data sparseness, Zipf's Law

Statistical Modeling & Estimation

Smoothing (1)

- **Laplace's Law** (adding one)

  $$P_{LAP}(w_1, \ldots, w_n) = \frac{C(w_1, \ldots, w_n) + 1}{N + B}$$

  - For large values of B too much probability mass is assigned to unseen events

- **Lidstone's Law**

  $$P_{LID}(w_1, \ldots, w_n) = \frac{C(w_1, \ldots, w_n) + \lambda}{N + B\lambda}$$

  - Usually $\lambda = 0.5$, *Expected Likelihood Estimation*.
  - For $\mu = N/(N + B\lambda)$, we get linear interpolation between MLE and uniform prior,

    $$P_{LID}(w_1, \ldots, w_n) = \mu \frac{C(w_1, \ldots, w_n)}{N} + (1 - \mu)\frac{1}{B}$$

Smoothing (2)

- **Held Out Estimator**
  - Divide the train corpus in two parts (A and B).
  - Let $T_r^{AB} = \sum_{\{\alpha : C_A(\alpha)=r\}} C_B(\alpha)$
  - Let $r = C_A(w_1, \ldots, w_n)$

$$P_{HO}(w_1, \ldots, w_n) = \frac{T_r^{AB}}{N_r^A N}$$

- **Cross Validation** (deleted estimation)

$$P_{DEL}(w_1, \ldots, w_n) = \frac{T_r^{AB} + T_r^{BA}}{(N_r^A + N_r^B)N}$$

Smoothing (3)

- **Absolute Discounting**

$$P_{ABS}(w_1,\ldots,w_n) = \begin{cases} (r-\delta)/N & \textit{if } r>0 \\ \frac{(B-N_0)\delta}{N_0 N} & \textit{otherwise} \end{cases}$$

- **Linear Discounting**

$$P_{LIN}(w_1,\ldots,w_n) = \begin{cases} (1-\alpha)r/N & \textit{if } r>0 \\ \alpha/N_0 & \textit{otherwise} \end{cases}$$

Smoothing (4)

- **Good-Turing Estimation**
  - Let $r = C(w_1, \ldots, w_n)$, observed frequency
  - Let $r^* = (r+1)\frac{E(N_{r+1})}{E(N_r)}$, adjusted frequency ($\approx (r+1)\frac{N_{r+1}}{N_r}$)
  - In practice $r^* = (r+1)\frac{E(N_{r+1})}{E(N_r)}$, where $S(r)$ = Smoothed values for $E(N_r)$.
  - Reserved mass: $\frac{N_1}{N}$

$$P_{GT}(w_1, \ldots, w_n) = \begin{cases} \frac{r^*}{N} & \textit{if } r > 0 \\ \frac{1 - \sum_{r=1}^{\infty} N_r \frac{r^*}{N}}{N_0} \approx \frac{N_1}{N_0 N} & \textit{otherwise} \end{cases}$$

Statistical Modeling & Estimation

Combining Estimators

- **Simple Linear Interpolation**

  $P_{LI}(w_n \mid w_{n-2}, w_{n-1}) =$
  $= \lambda_1 P_1(w_n) + \lambda_2 P_2(w_n \mid w_{n-1}) + \lambda_3 P_3(w_n \mid w_{n-2}, w_{n-1})$

- **Katz's Backing-off**

$$P_{BO}(w_i \mid w_{i-n+1}, \ldots, w_{i-1}) = \begin{cases} (1 - d_{w_{i-n+1},\ldots,w_{i-1}}) \dfrac{C(w_{i-n+1}, \ldots, w_i)}{C(w_{i-n+1}, \ldots, w_{i-1})} \\ \quad \text{if } C(w_{i-n+1}, \ldots, w_i) > 0 \\ \alpha_{w_{i-n+1},\ldots,w_{i-1}} P_{BO}(w_i \mid w_{i-n+2}, \ldots, w_{i-1}) \\ \quad \text{otherwise} \end{cases}$$

- **General Linear Interpolation**

$$P_{LI}(w_n \mid h) = \sum_{i=1}^{k} \lambda_i(h) P_i(w \mid h)$$

Surprise Measures

- Entropy measures uncertainty: If a model captures more of the structure of the language, its entropy will be lower.
- Pointwise Entropy: $H(w \mid h) = -\log m(w \mid h)$
- Cross Entropy:

$$
\begin{aligned}
H(X_{1n}, m) = & -\lim_{n\to\infty} \frac{1}{n} \sum_{x_{1n}} p(x_{1n}) \log m(x_{1n}) = \\
= & -\lim_{n\to\infty} \frac{1}{n} \log m(x_{1n}) \approx -\frac{1}{n} \log m(x_{1n})
\end{aligned}
$$

- Perplexity:
  $Perplexity(x_{1n}, m) = 2^{H(x_{1n}, m)} = m(x_{1n})^{-\frac{1}{n}}$

Train and test data

- Training data
- Overtraining, cross entropy
- Test data
- Splitting training data: *Held out* or *Validation* data.
- Splitting testing data: *Development test* or *Tuning* data
- Mean and variance estimation (cross-validation)
- System comparison: χ^2, t, bayesian decision theory, ...

Modeling Classification Problems

- Classification problems: Estimate probability that a class a appears with –or given– an event b: $P(a, b)$; $P(a \mid b)$
- ML Estimation problems
  - Corpus sparseness
  - Smoothing
  - Combining evidence
    - Independence assumptions
    - Interpolation

Maximum Entropy Modeling

- Maximum Entropy: alternative estimation technique.
- Able to deal with different kinds of evidence
- ME principle:
  - Do not assume anything about non-observed events.
  - Find the most uniform (maximum entropy) probability distribution that matches the observations.

Simple Example

- Observed facts are constraints for the desired model p.
- Observed fact $p(x,0) + p(y,0) = 0.6$ is implemented as a constraint on the expectation of feature f_1 of model p. That is: $E_p f_1 = 0.6$ where
$E_p f_1 = \sum_{a \in \{x,y\}, b \in \{0,1\}} p(a,b) f_1(a,b)$
$f_1(a,b) = \begin{cases} 1 & \mathit{if}\ \ b = 0 \\ 0 & \mathit{otherwise} \end{cases}$
- Most *uncertain* way to satisfy constraints

$P(a,b)$	0	1	
x	?	?	
y	?	?	
total	0.6		1.0

$P(a,b)$	0	1	
x	0.5	0.1	
y	0.1	0.3	
total	0.6		1.0

$P(a,b)$	0	1	
x	0.3	0.2	
y	0.3	0.2	
total	0.6		1.0

Probability Model

- Infinite probability models consistent with observations:

$$P = \{p \mid E_p f_j = E_{\widetilde{p}} f_j,\ j = 1 \ldots k\}$$

$$E_{\widetilde{p}} f_j = \sum_{a,b} \widetilde{p}(a,b) f_j(a,b)$$

$$E_p f_j = \sum_{a,b} \widetilde{p}(b) p(a \mid b) f_j(a,b)$$

- Maximum entropy model

$$p^* = \arg\max_{p \in P} H(p)$$

$$H(p) = -\sum_{a,b} \widetilde{p}(b) p(a \mid b) \log p(a \mid b)$$

Example 2

Maximum entropy model for *in* translation to French

- No constraints

$P(a,b)$	dans	en	à	au-cours-de	pendant	
	0.2	0.2	0.2	0.2	0.2	
total						1.0

- Subject to constraint: $p(dans) + p(en) = 0.3$

$P(a,b)$	dans	en	à	au-cours-de	pendant	
	0.15	0.15	0.233	0.233	0.233	
total	**0.3**					1.0

- Constraints: $p(dans) + p(en) = 0.3$ and $p(dans) + p(à) = 0.5$

...Not so easy !

Parameter estimation

- Exponential models. (Lagrange multipliers optimization)
  $p(a \mid b) = \frac{1}{Z(b)} \prod_{j=1}^{k} \alpha_j^{f_j(a,b)} \qquad \alpha_j > 0$
  $Z(b) = \sum_a \prod_{i=1}^{k} \alpha_i^{f_i(a,b)}$
- also formuled as
  $p(a \mid b) = \frac{1}{Z(b)} \exp(\sum_{j=1}^{k} \lambda_j f_j(a,b))$
  $\lambda_i = \ln \alpha_i$
- Each model parameter models the influence of a feature.
- Optimal model parameters:
  - GIS. Generalized Iterative Scaling(Darroch & Ratcliff 72)
  - IIS. Improved Iterative Scaling (Della Pietra et al. 96)

Improved Iterative Scaling (IIS)

Input: Feature functions $f_1 \dots f_n$, empirical distribution $\widetilde{p}(a,b)$
Output: λ_i^* parameters for optimal model p^*

Start with $\lambda_i = 0$ for all $i \in \{1 \dots n\}$
Repeat
  For each $i \in \{1 \dots n\}$ **do**
    let $\Delta\lambda_i$ be the solution to
    $\sum_{a,b} \widetilde{p}(b) p(a \mid b) f_i(a,b) \exp(\Delta\lambda_i \sum_{j=1}^{n} f_j(a,b)) = \widetilde{p}(f_i)$
    $\lambda_i \leftarrow \lambda_i + \Delta\lambda_i$
  end for
Until all λ_i have converged

Maximum Entropy Modeling

- Modeling Classification Problems: MLE *vs* MEM
- Building ME Models
- Application to NLP

Application to NLP Tasks

- Speech processing (Rosenfeld 94)
- Machine Translation (Brown et al 90)
- Morphology (Della Pietra et al. 95)
- Clause boundary detection (Reynar & Ratnaparkhi 97)
- PP-attachment (Ratnaparkhi et al 94)
- PoS Tagging (Ratnaparkhi 96, Black et al 99)
- Partial Parsing (Skut & Brants 98)
- Full Parsing (Ratnaparkhi 97, Ratnaparkhi 99)
- Text Categorization (Nigam et al 99)

PoS Tagging (Ratnaparkhi 96)

- Probabilistic model over $H \times T$

$$h_i = (w_i, w_{i+1}, w_{i+2}, w_{i-1}, w_{i-2}, t_{i-1}, t_{i-2})$$

$$f_j(h_i, t) = \begin{cases} 1 & if\ suffix(w_i) =' ing' \wedge t = VBG \\ 0 & otherwise \end{cases}$$

- Compute $p^*(h, t)$ using GIS
- Disambiguation algorithm: *beam search*

$$p(t \mid h) = \frac{p(h, t)}{\sum_{t' \in T} p(h, t')}$$

$$p(t_1 \dots t_n \mid w_1 \dots w_n) = \prod_{i=1}^{n} p(t_i \mid h_i)$$

Text Categorization (Nigam et al 99)

- Probabilistic model over $W \times C$

$$d = (w_1, w_2, \ldots, w_N)$$

$$f_{w,c'}(d,c) = \begin{cases} \frac{N(d,w)}{N(d)} & \quad if\ c = c' \\ 0 & otherwise \end{cases}$$

- Compute $p^*(d,c)$ using IIS
- Disambiguation algorithm: Select class with highest

$$P(c|d) = \frac{1}{Z(d)} exp(\sum_i \lambda_i f_i(d,c))$$

MEM Summary

- Advantages
  - Teoretically well founded
  - Enables combination of random context features
  - Better probabilistic models than MLE (no smoothing needed)
  - General approach (features, events and classes)
- Disadvantages
  - Implicit probabilistic model (joint or conditional probability distribution obtained from model parameters).
  - High computational cost of GIS and IIS.
  - Overfitting in some cases.

Graphical Models

Types of Graphical Model

- **Generative models:**
  - Bayes rule $\Rightarrow$ independence assumptions.
  - Able to *generate* data.
- **Conditional models:**
  - No independence assumptions.
  - Unable to generate data.

Most algorithms of both kinds make assumptions about the nature of the data-generating process, predefining a fixed model structure and only acquiring from data the distributional information.

Examples of Graphical Models

- **Generative models:**
  - HMM (Rabiner 1990), IOHMM (Bengio 1996).
  - Non-graphical: Stochastic Grammars (Lary & Young 1990)
  - Automata-learning algorithms: *No assumptions about model structure*. VLMM (Rissanen 1983), Suffix Trees (Galil & Giancarlo 1988), CSSR (Shalizi & Shalizi 2004).
- **Conditional models:**
  - discriminative MM (Bottou 1991), MEMM (McCallum et al. 2000), CRF (Lafferty et al. 2001).
  - Non-graphical: Maximum Entropy Models (Berger et al 1996).

See (M. Padró 2008) for a brief survey and reference source.

Graphical Models

Markov Models and Hidden Markov Models

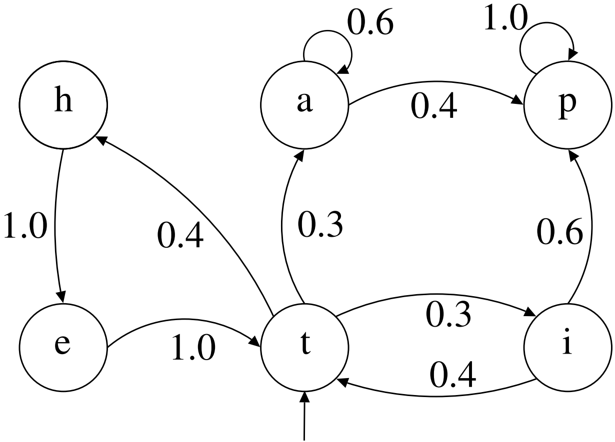
HMM Fundamental Questions

1. Observation Probability
2. Best State Sequence
3. Parameter Estimation

Markov Models

- $X = (X_1, \dots, X_T)$ sequence of random variables taking values in $s = \{s_1, \dots, s_N\}$
- Markov Properties
  - Limited Horizon:
    $P(X_{t+1} = s_k \mid X_1, \dots, X_t) = P(X_{t+1} = s_k \mid X_t)$
  - Time Invariant (Stationary):
    $P(X_{t+1} = s_k \mid X_t) = P(X_2 = s_k \mid X_1)$
- Transition matrix: $a_{ij} = P(X_{t+1} = s_j \mid X_t = s_i)$
- Initial probabilities: $\pi_i = P(X_1 = s_i)$
- Sequence probability
  $P(X_1, .., X_T) = P(X_1)P(X_2 \mid X_1)P(X_3 \mid X_1 X_2) \cdots P(X_T \mid X_1..X_{T-1}) =$
  $= P(X_1)P(X_2 \mid X_1)P(X_3 \mid X_2) \cdots P(X_T \mid X_{T-1}) = \pi_{X_1} \prod_{t=1}^{T-1} a_{X_t X_{t+1}}$

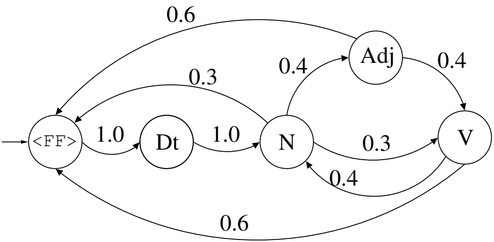
MM Example



Hidden Markov Models (HMM)

- States and Observations
- Emission Probability:
$$P(O_t = k \mid X_t = s_i, X_{t+1} = s_j) = b_{ijk}$$
- Used when underlying events probabilistically generate surface events
  - PoS tagging (hidden states: PoS tags, observations: words)
- Trainable with unannotated data. Expectation Maximization (EM) algorithm.
- arc-emission *vs* state-emission

Example: PoS Tagging



Emission probabilities	.	el	la	gato	niña	come	corre	pescado	fresco	pequeña	grande
<FF>	1.0										
Dt		0.6	0.4								
N				0.6	0.1			0.3			
V						0.7	0.3				
Adj									0.3	0.3	0.4

Graphical Models

Markov Models and Hidden Markov Models

HMM Fundamental Questions

1. Observation Probability
2. Best State Sequence
3. Parameter Estimation

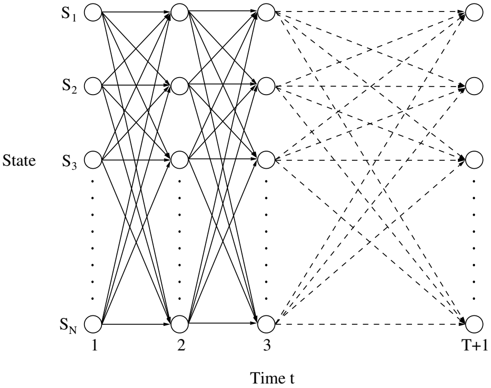
HMM Fundamental Questions

1. **Observation probability (decoding):** Given a model $\mu = (A, B, \pi)$, how we do efficiently compute how likely is a certain observation ? That is, $P(O \mid \mu)$
2. **Classification:** Given an observed sequence O and a model μ, how do we choose the state sequence $(X_1, \ldots, X_{T+1})$ that best explains the observations?
3. **Parameter estimation:** Given an observed sequence O and a space of possible models, each with different parameters (A, B, π), how do we find the model that best explains the observed data

Question 1. Observation probability

- Let $O = (o_1, \ldots, o_T)$ observation sequence.
  For any state sequence $X = (X_1, \ldots, X_T)$, we have:
  $$P(O \mid X, \mu) = \prod_{t=1}^{T} P(o_t \mid X_t, X_{t+1}, \mu) =$$
  $$= b_{X_1 X_2 o_1} b_{X_2 X_3 o_2} \cdots b_{X_T X_{T+1} o_T}$$
  $$P(X \mid \mu) = \pi_{X_1} a_{X_1 X_2} a_{X_2 X_3} \cdots a_{X_T X_{T+1}}$$
  Since $P(O, X \mid \mu) = P(O \mid X, \mu) P(X \mid \mu)$, thus
  $$P(O \mid \mu) = \sum_X P(O \mid X, \mu) P(X \mid \mu) =$$
  $$= \sum_{X_1 \cdots X_{T+1}} \pi_{X_1} \prod_{t=1}^{T} a_{X_t X_{t+1}} b_{X_t X_{t+1} o_t}$$
  Complexity: $\mathcal{O}(TN^T)$
- Dynammic Programming. Trellis, lattices.

Trellis



Fully connected HMM where one can move to any state to any other at each step. A node $\{s_i, t\}$ of the trellis stores information about state sequences which include $X_t = i$.

Forward & Backward (1)

- Forward procedure: $\alpha_i(t) = P(o_1 o_2 \cdots o_{t-1}, X_t = i \mid \mu)$
  1. Inicialization: $\alpha_i(1) = \pi_i \qquad 1 \leq i \leq N$
  2. Induction: $\alpha_j(t+1) = \sum_{i=1}^{N} \alpha_i(t) a_{ij} b_{ijo_t}$
     $1 \leq t \leq T, 1 \leq j \leq N$
  3. Total: $P(O \mid \mu) = \sum_{i=1}^{N} \alpha_i(T+1)$

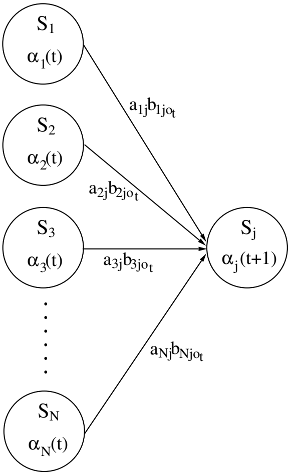
  Complexity: $\mathcal{O}(N^2 T)$
- Backward procedure: $\beta_i(t) = P(o_t \cdots o_T \mid X_t = i, \mu)$
  1. Inicialization: $\beta_i(T+1) = 1 \qquad 1 \leq i \leq N$
  2. Induction: $\beta_i(t) = \sum_{j=1}^{N} a_{ij} b_{ijo_t} \beta_j(t+1)$
     $1 \leq t \leq T, 1 \leq i \leq N$
  3. Total: $P(O \mid \mu) = \sum_{i=1}^{N} \pi_i \beta_i(1)$

Forward & Backward (2)

- Combination
  $P(O, X_t = i \mid \mu) =$
  $= P(o_1 \cdots o_{t-1}, X_t = i, o_t \cdots o_T \mid \mu) = \alpha_i(t)\beta_i(t)$

  thus, $P(O \mid \mu) = \sum_{i=1}^{N} \alpha_i(t)\beta_i(t) \qquad 1 \leq t \leq T+1$

Forward calculations



Closeup of the computation of forward probabilities at one node. The forward probability $\alpha_j(t+1)$ is calculated by summing the product of the probabilities on each incoming arc with the forward probability of the originating node.

Question 2. Best state sequence

- Most likely path.
- Compute $\arg\max_X P(X \mid O, \mu)$
  - For a given O, compute $\arg\max_X P(X, O \mid \mu)$
  - Let $\delta_j(t) = \max_{X_1 \cdots X_{t-1}} P(X_1 \cdots X_{t-1}, o_1 \cdots o_{t-1}, X_t = j|\mu)$
- Viterbi algorithm
  1. Initialization. $\delta_j(1) = \pi_j \qquad 1 \leq j \leq N$
  2. Induction. $\delta_j(t+1) = \max_{1 \leq i \leq N} \delta_i(t) a_{ij} b_{ijo_t} \qquad 1 \leq j \leq N$
  3. Store backtrace. $\psi_j(t+1) = \arg\max_{1 \leq i \leq N} \delta_i(t) a_{ij} b_{ijo_t} \quad 1 \leq j \leq N$
  4. Termination path readout (backwards)
     - 4.1 $\hat{X}_{T+1} = \arg\max_{1 \leq i \leq N} \delta_i(T+1)$
     - 4.2 $\hat{X}_t = \psi_{\hat{X}_{t+1}}(t+1)$
     - 4.3 $P(\hat{X}) = \max_{1 \leq i \leq N} \delta_i(T+1)$

Question 3. Parameter Estimation

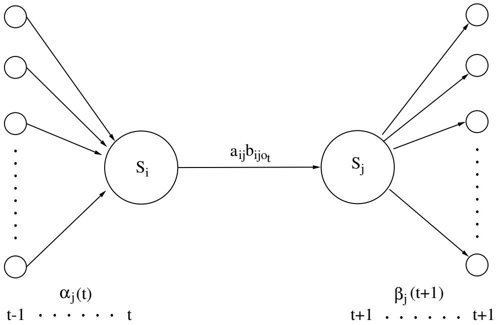
- Obtain model parameters $\mu = (A, B, \pi)$ given observation: $\arg\max_\mu P(O_{train} \mid \mu)$
- Baum-Welch (Forward-Backward) algorithm. Iterative hill-climbing. Special case of Expectation Maximization.
- Let $p_t(i,j) =$

$$= P(X_t = i, X_{t+1} = j \mid O, \mu) = \frac{P(X_t = i, X_{t+1} = j, O \mid \mu)}{P(O \mid \mu)} =$$

$$= \frac{\alpha_i(t) a_{ij} b_{ijo_t} \beta_j(t+1)}{\sum_{m=1}^N \alpha_m(t)\beta_m(t)} = \frac{\alpha_i(t) a_{ij} b_{ijo_t} \beta_j(t+1)}{\sum_{m=1}^N \sum_{n=1}^N \alpha_m(t) a_{mn} b_{mno_t} \beta_n(t+1)}$$

- Let $\gamma_i(t) = \sum_{j=1}^N p_t(i,j)$, thus
  $\sum_{t=1}^T \gamma_i(t) =$ expected # of transitions from state i in O.
  $\sum_{t=1}^T p_t(i,j) =$ expected # of transitions from state i to j in O.

Arc probability



The probability of traversing an arc. Given an observation sequence and a model, we can work out the probability that the Markov process went from state s_i to s_j at time t.

Reestimation

- Iterative reestimation

$$\hat{\pi}_i = \gamma_i(1)$$

$$\hat{a}_{ij} = \frac{\sum_{t=1}^{T} p_t(i,j)}{\sum_{t=1}^{T} \gamma_i(t)}$$

$$\hat{b}_{ijk} = \frac{\sum_{\{t:o_t=k,1\leq t\leq T\}} p_t(i,j)}{\sum_{t=1}^{T} p_t(i,j)}$$

- EM Property: $P(O \mid \hat{\mu}) \geq P(O \mid \mu)$
- Iterative improving. Local maxima

Clustering

- Partition a set of objects into clusters.
- Objects: features and values
- Similarity measure
- Utilities:
  - Exploratory Data Analysis (EDA).
  - Generalization (*learning*). Ex: *on Monday*, *on Sunday*, *? Friday*
- Supervised *vs* unsupervised classification
- Object assignment to clusters
  - Hard. *one cluster per object.*
  - Soft. *distribution* $P(c_i \mid x_j)$. *Degree of membership.*

Clustering

- Produced structures
  - Hierarchical (set of clusters + relationships)
    - Good for detailed data analysis
    - Provides more information
    - Less efficient
    - No single best algorithm
  - Flat / Non-hierarchical (set of clusters)
    - Preferable if efficiency is required or large data sets
    - K-means: Simple method, sufficient starting point.
    - K-means assumes euclidean space, if is not the case, EM may be used.
- Cluster representative
  - Centroid $\overrightarrow{\mu} = \frac{1}{|c|} \sum_{\overrightarrow{x} \in c} \overrightarrow{x}$

The Concept of Similarity

- *Similarity, proximity, affinity, distance, difference, divergence*
- We use *distance* when metric properties hold:
  - $d(x, x) = 0$
  - $d(x, y) \geq 0$ when $x \neq y$
  - $d(x, y) = d(y, x)$ (simmetry)
  - $d(x, z) \leq d(x, y) + d(y, z)$ (triangular inequation)
- We use *similarity* in the general case
  - Function: $sim : A \times B \longrightarrow S$ (where S is often $[0, 1]$)
  - Homogeneous: $sim : A \times A \longrightarrow S$ (e.g. word-to-word)
  - Heterogeneous: $sim : A \times B \longrightarrow S$ (e.g. word-to-document)
  - Not necessarily symmetric, or holding triangular inequation.

The Concept of Similarity

- If A is a metric space, the distance in A may be used.
  - $D_{euclidean}(\vec{x},\vec{y}) = |\vec{x}-\vec{y}| = \sqrt{\sum_i (x_i - y_i)^2}$
  - $D(d^i,d^j) = \sqrt{\sum_{k=1}^{N} (d^i_k - d^j_k)^2}$
- Similarity and distance
  - $sim_D(A,B) = \frac{1}{1+D(A,B)}$
  - monotonic: $min\{sim(x,y), sim(x,z)\} \geq sim(x, y \cup z)$

Applications

- Clustering, case-based reasoning, IR, ...
- Discovering related words - Distributional similarity
- Resolving syntactic ambiguity - Taxonomic similarity
- Acquiring selectional restrictions

Relevant Information

- Content (information about compared units)
  - Words: form, morphology, PoS, ...
  - Senses: synset, topic, domain, ...
  - Syntax: parse trees, syntactic roles, ...
  - Documents: words, collocations, NEs, ...
- Context (information about the situation in which simmilarity is computed)
  - Window–based *vs.* Syntactic–based
- External Knowledge
  - Monolingual/bilingual dictionaries, ontologies, corpora

Vectorial methods (1)

- L_1 norm, Manhattan distance, taxi-cab distance, city-block distance
$$L_1(\vec{x},\vec{y}) = \sum_{i=1}^{N} |x_i - y_i|$$
- L_2 norm, Euclidean distance
$$L_2(\vec{x},\vec{y}) = |\vec{x} - \vec{y}| = \sqrt{\sum_{i=1}^{N} (x_i - y_i)^2}$$
- Cosine distance
$$cos(\vec{x},\vec{y}) = \frac{\vec{x}\cdot\vec{y}}{|\vec{x}|\cdot|\vec{y}|} = \frac{\sum_i x_i y_i}{\sqrt{\sum_i x_i^2}\cdot\sqrt{sum_i y_i^2}}$$

Vectorial methods (2)

- L_1 and L_2 norms are particular cases of Minkowsky measure

$$D_{minkowsky}(\vec{x},\vec{y}) = L_r(\vec{x},\vec{y}) = \left(\sum_{i=1}^{N}(x_i - y_i)^r\right)^{\frac{1}{r}}$$

- Camberra distance

$$D_{camberra}(\vec{x},\vec{y}) = \sum_{i=1}^{N}\frac{|x_i - y_i|}{|x_i + y_i|}$$

- Chebychev distance

$$D_{chebychev}(\vec{x},\vec{y}) = \max_{i=1}^{N}|x_i - y_i|$$

Set-oriented methods (3): Binary–valued vectors seen as sets

- Matching coefficient. $D_{mc}(X,Y) = |X \cap Y|$
- Dice. $D_{dice}(X,Y) = \dfrac{2 \cdot |X \cap Y|}{|X| + |Y|}$
- Jaccard. $D_{jaccard}(X,Y) = \dfrac{|X \cap Y|}{|X| \cup |Y|}$
- Overlap. $D_{overlap}(X,Y) = \dfrac{|X \cap Y|}{\min(|X|,|Y|)}$
- Cosine. $cos(X,Y) = \dfrac{|X \cap Y|}{\sqrt{|X| \times |Y|}}$

Set-oriented methods (4): Agreement contingency table

$$
\begin{array}{cc|c|c|c}
 & & \multicolumn{2}{c}{\text{Object } i} & \\
 & & 1 & 0 & \\
\hline
\text{Object } j & 1 & a & b & a+b \\
\hline
 & 0 & c & d & c+d \\
\hline
 & & a+c & b+d & p
\end{array}
$$

- Matching coefficient. $D_{mc}(i,j) = \dfrac{a+d}{p}$
- Jaccard. $D_{jaccard}(X,Y) = \dfrac{a}{a+b+c}$

Distributional Similarity

- Particular case of vectorial representation where attributes are probability distributions

  $$\vec{x}^T = [x_1 \dots x_N] \text{ such that } \forall i, 0 \leq x_i \leq 1 \text{ and } \sum_{i=1}^{N} x_i = 1$$

- Kullback-Leibler Divergence (Relative Entropy)

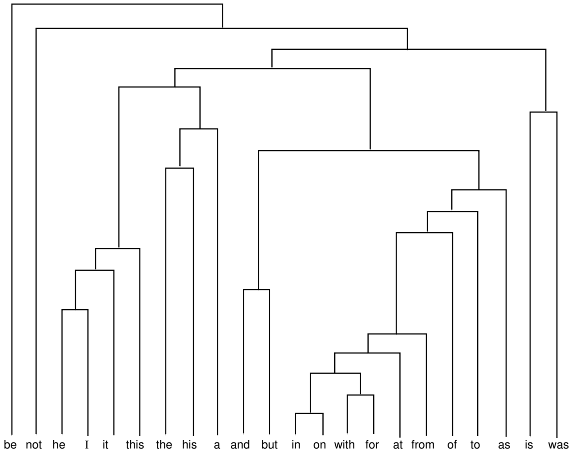
  $$D(q||r) = \sum_{y \in Y} q(y) \log \frac{q(y)}{r(y)} \qquad \text{(non symmetrical)}$$

- Mutual Information

  $$I(A,B) = D(h||f \cdot g) = \sum_{a \in A} \sum_{b \in B} h(a,b) \log \frac{h(a,b)}{f(a) \cdot g(b)}$$

  (KL-divergence between joint and product distribution)

Dendogram



Single-link clustering of 22 frequent English words represented as a dendogram.

Hierarchical Clustering

- Bottom-up (Agglomerative Clustering)
  Start with individual objects, iteratively group the most similar.
- Top-down (Divisive Clustering)
  Start with all the objects, iteratively divide them maximizing within-group similarity.

Agglomerative Clustering (Bottom-up)

Input: A set $\mathcal{X} = \{x_1, \ldots, x_n\}$ of objects
A function sim: $\mathcal{P}(\mathcal{X}) \times \mathcal{P}(\mathcal{X}) \longrightarrow \mathcal{R}$
Output: A cluster hierarchy

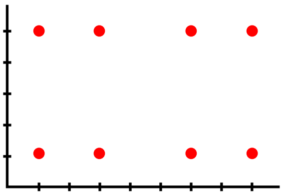
for $i{:=}1$ **to** n **do** $c_i{:=}\{x_i\}$ **end**
$C{:=}\{c_1, \ldots, c_n\}; \;\; j{:=}n+1$
while $C > 1$ **do**
$\quad (c_{n_1}, c_{n_2}){:=}\arg\max_{(c_u,c_v)\in C\times C} \text{sim}(c_u, c_v)$
$\quad c_j = c_{n_1} \cup c_{n_2}$
$\quad C{:=}C \setminus \{c_{n_1}, c_{n_2}\} \cup \{c_j\}$
$\quad j{:=}j+1$
end–while

Cluster Similarity

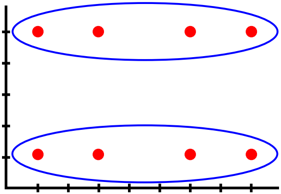
Similarity measure families

- Single link: Similarity of two most similar members
  - Local coherence (close objects are in the same cluster)
  - Elongated clusters (chaining effect)
- Complete link: Similarity of two least similar members
  - Global coherence, avoids elongated clusters
  - Better (?) clusters
- Group average: Average similarity between members
  - Trade-off between global coherence and efficiency

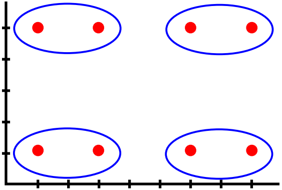
Examples



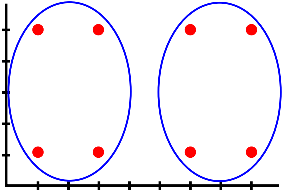
A cloud of points in a plane



Single-link clustering



Intermediate clustering



Complete-link clustering

Divisive Clustering (Top-down)

Input: A set $\mathcal{X} = \{x_1, \ldots, x_n\}$ of objects

A function coh: $\mathcal{P}(\mathcal{X}) \longrightarrow \mathcal{R}$

A function split: $\mathcal{P}(\mathcal{X}) \longrightarrow \mathcal{P}(\mathcal{X}) \times \mathcal{P}(\mathcal{X})$

Output: A cluster hierarchy

$C{:=}\{\mathcal{X}\};\ \ c_1{:=}\mathcal{X};\ \ j{:=}1$

while $\exists c_i \in C$ s.t. $|c_i| > 1$ **do**

$\quad c_u{:=}\arg\min_{c_v \in C}\ \mathrm{coh}(c_v)$

$\quad (c_{j+1}, c_{j+2}) = \mathrm{split}(c_u)$

$\quad C{:=}C \setminus \{c_u\} \cup \{c_{j+1}, c_{j+2}\}$

$\quad j{:=}j + 2$

end–while

Top-down clustering

- Cluster splitting: Finding two sub-clusters
- Split clusters with lower *coherence*:
  - Single-link, Complete-link, Group-average
  - Splitting is a sub-clustering task:
    - Non-hierarchical clustering
    - Bottom-up clustering
- Example: Distributional noun clustering (Pereira et al., 93)
  - Clustering nouns with similar verb probability distributions
  - KL divergence as distance between distributions

$$D(p||q) = \sum_{x \in X} p(x) \log \frac{p(x)}{q(x)}$$

  - Bottom-up clustering not applicable due to some $q(x) = 0$

Non-hierarchical clustering

- Start with a partition based on random seeds
- Iteratively refine partition by means of *reallocating* objects
- Stop when cluster quality doesn't improve further
  - group-average similarity
  - mutual information between adjacent clusters
  - likelihood of data given cluster model
- Number of desired clusters ?
  - Testing different values
  - Minimum Description Length: the goodness function includes information about the number of clusters

K-means

- Clusters are represented by centers of mass (centroids) or a prototypical member (medoid)
- Euclidean distance
- Sensitive to outliers
- Hard clustering
- $\mathcal{O}(n)$

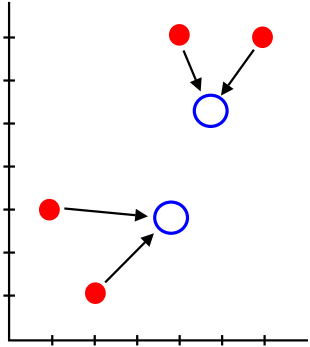
K-means algorithm

Input: A set $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subseteq \mathcal{R}^m$
A distance measure $d : \mathcal{R}^m \times \mathcal{R}^m \longrightarrow \mathcal{R}$
A function for computing the mean $\mu : \mathcal{P}(\mathcal{R}) \longrightarrow \mathcal{R}^m$

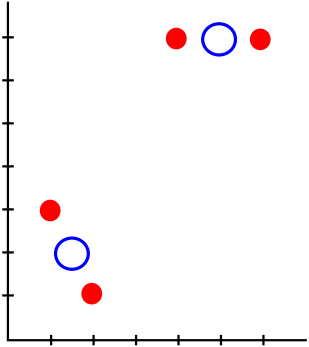
Output: A partition of $\mathcal{X}$ in clusters

Select k initial centers $\mathbf{f}_1, \ldots, \mathbf{f}_k$
while stopping criterion is not true **do**
  for all clusters c_j **do**
    $c_j := \{\mathbf{x}_i \mid \forall \mathbf{f}_l \; d(\mathbf{x}_i, \mathbf{f}_j) \leq d(\mathbf{x}_i, \mathbf{f}_l)\}$
  for all means $\mathbf{f}_j$ **do**
    $\mathbf{f}_j := \mu(c_j)$
end–while

K-means example



Assignment



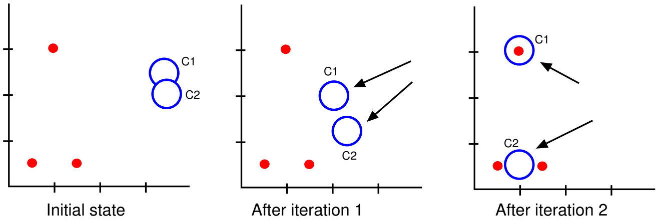
Recomputation of means

EM algorithm

- Estimate the (hidden) parameters of a model given the data
- Estimation–Maximization deadlock
  - Estimation: If we knew the parameters, we could compute the expected values of the hidden structure of the model.
  - Maximization: If we knew the expected values of the hidden structure of the model, we could compute the MLE of the parameters.
- NLP applications
  - Forward-Backward algorithm (Baum-Welch reestimation).
  - Inside-Outside algorithm.
  - Unsupervised WSD

EM example

- Can be seen as a *soft* version of K-means
- Random initial centroids
- Soft assignments
- Recompute (averaged) centroids



An example of using the EM algorithm for soft clustering

Clustering evaluation

- Related to a reference clustering: Purity and Inverse Purity.

$$P = \frac{1}{|D|}\sum_{c} \max_{x} |c \cap x|$$

$$IP = \frac{1}{|D|}\sum_{x} \max_{c} |c \cap x|$$

Where:

$c =$ obtained clusters

$x =$ expected clusters

- Without reference clustering: *Cluster quality* measures: Coherence, average distance, etc.

References

References (1)

Statistics and Linguistics

- S. Abney, **Statistical Methods and Linguistics** In *The Balancing Act: Combining Symbolic and Statistical Approaches to Language*. The MIT Press, Cambridge, MA, 1996.
- L. Lee, **"I'm sorry Dave, I'm afraid I can't do that": Linguistics, Statistics, and Natural Language Processing**. National Research Council study on Fundamentals of Computer Science, 2003.

References (2)

Statistical basics, applications to NLP

- T. Cover & J. Thomas, **Elements of Information Theory**. John Wiley & Sons, 1991.
- C. Manning & H. Schütze, **Foundations of Statistical Natural Language Processing**. The MIT Press. Cambridge, MA: May 1999.
- B. Krenn & C. Samuelsson, **The Linguist's Guide to Statistics (DON'T PANIC)**. Universität des Saarlandes. Saarbrücken, Germany, 1997

References (3)

Maximum Entropy Modeling

- A. Ratnaparkhi, **Maximum Entropy Models for Natural Language Ambiguity Resolution**. Ph.D Thesis. University of Pennsylvania, 1998
- A. Berger, S.A. Della Pietra & V.J. Della Pietra, **A Maximum Entropy Approach to Natural Language Processing**. Computational Linguistics, 22(1):39-71, 1996

References (4)

Graphical Models

- M. Padró **Applying Causal-State Splitting Reconstruction Algorithm to Natural Language Processing Tasks**. Ph.D Thesis. Universitat Politècnica de Catalunya, 2008.
- S.L. Lauritzen, **Graphical Models**. Oxford University Press, 1996

References (5)

Similarity, Clustering

- H. Rodriguez, **Some notes on using similarity (and distance) measures in Computational Linguistics**. Curso de Industrias de la Lengua. FDS, Soria, 2002.
  `http://www.lsi.upc.edu/~horacio/varios/soria2002.zip`
- L. Lee, **Similarity-Based Approaches to Natural Language Processing**. PhD Thesis, Harvard University Tech. Report TR-11-97. Harvard, 1997.
  `http://www.cs.cornell.edu/home/llee/papers/thesis.pdf`
- C. D. Manning, P. Raghavan & H. Schütze, **Introduction to Information Retrieval**. Cambridge University Press, 2008.
  `http://www-csli.stanford.edu/ hinrich/information-retri`