

Machine Learning applied to Text Analysis

Eric Gaussier (and Nicola Cancedda)

January 27, 2004



Which objects are we interested in?

[The Arundhati Roy Web.html](#)

[ArundhatiRoy.doc](#)

For what purposes?

Extract and present to users information relevant to their needs

- Information Retrieval, Question Answering
- Information Extraction
- Categorisation (routing)
- Browsing high-level (induced) representations
- Text mining

Overview of the presentation

1. Natural language processing

1. Morphological analysis
2. Parsing (syntactic analysis)
3. Semantic analysis

2. *Structural analysis*

3. Textual information access

1. Information retrieval
2. Learning representations
3. Categorisation
4. *Text mining*

4. Multilingual aspects

Machine Learning



Morphological analysis (1)

Morphology: study of word forms and word formation

nationalisation

[nation_N → national_A → nationalise_V → nationalisation]

[nation]_N[al]_A[is]_V[ation]_N

Why is it useful?

1. Allows to normalise variants of same word: inflectional morphology (genre, number, conjugation, declension) – useful in many IR-related contexts

Morphological analysis (2)

2. Allows to normalise variants of same concept: derivational morphology (preserve meaning, not necessarily part-of-speech) – useful in many IR-related contexts (stemmers, a la Porter, combines the twos)
3. Essential component in natural language applications (spelling correction, syntactic analysis, comprehension aid, machine translation, ...)

Why is it difficult?

1. Graphemic proximity does not necessarily reflect semantic proximity

drive → *driver*, *plumb* ✗ → *plumber*

Semantic proximity is difficult to formalise

Morphological analysis (3)

2. Morpho-/phono-graphemic rules for word formation

Ex.: form an adverb from an adjective by suffixing –y

green → *greeny*

But doubling of some consonants

grateful → *gratefully*

And many idiosyncrasies (*conduit*, *conduct*)

3. Specific domains have their own suffixes and word formation rules (e.g. *-itis*, *-in* in the medical domain)

Morphological analysis (4)

Standard approach: hire lexicographers to build morphological resources (tools and methodology in place, 1 m/y for inflectional morphological analyser of a given language)

Analysis of unknown words often minimal.(e.g. CELEX)

Need for automatic techniques to analyse new words, for new domains, from inflectional to derivational morphology

Morphological analysis (5)

Supervised learning (Bosh & Daelemans 99; Dzeroski & Erjavec 97)

- Using CELEX to derive instances, morphological analysis as memory-based learning (both inflectional and derivational morphology)
- Using MULTEXT-East lexical resources to derive examples
surface_form lemma (Noun common masculine singular nominative)
surface_form lemma (Noun common masculine singular accusative)

use inductive logic programming to learn inflectional rules (FOIDL; Mooney 95)

Morphological analysis (6)

Unsupervised learning (Goldsmith 01)

1. Morphology is composed of q list of stems, a list of suffixes, a list of signatures (ways in which the two can be combined) - *NULL.ed.ing.s / print, ask, answer, ...*
2. Learning based on MDL: find morphology M from corpus C such that the length in bits of M plus length of C ($-\log P_M(C)$) is minimal
3. Identification of potential stems, suffixes and signatures in corpus

g o v e r¹ n⁶ m¹ e



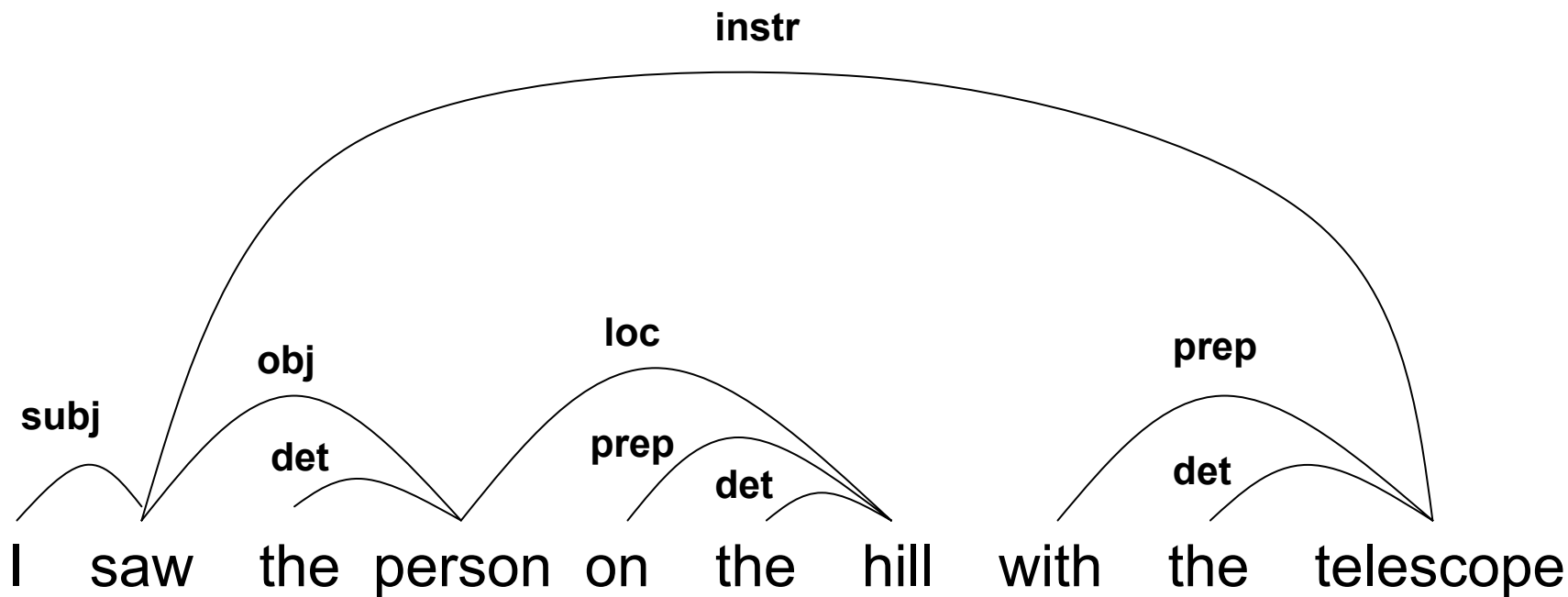
peak of successor frequency

Natural Language Parsing

Problem Statement:

Given a sentence, identify its *phrase structure*...

...or its *dependency structure*

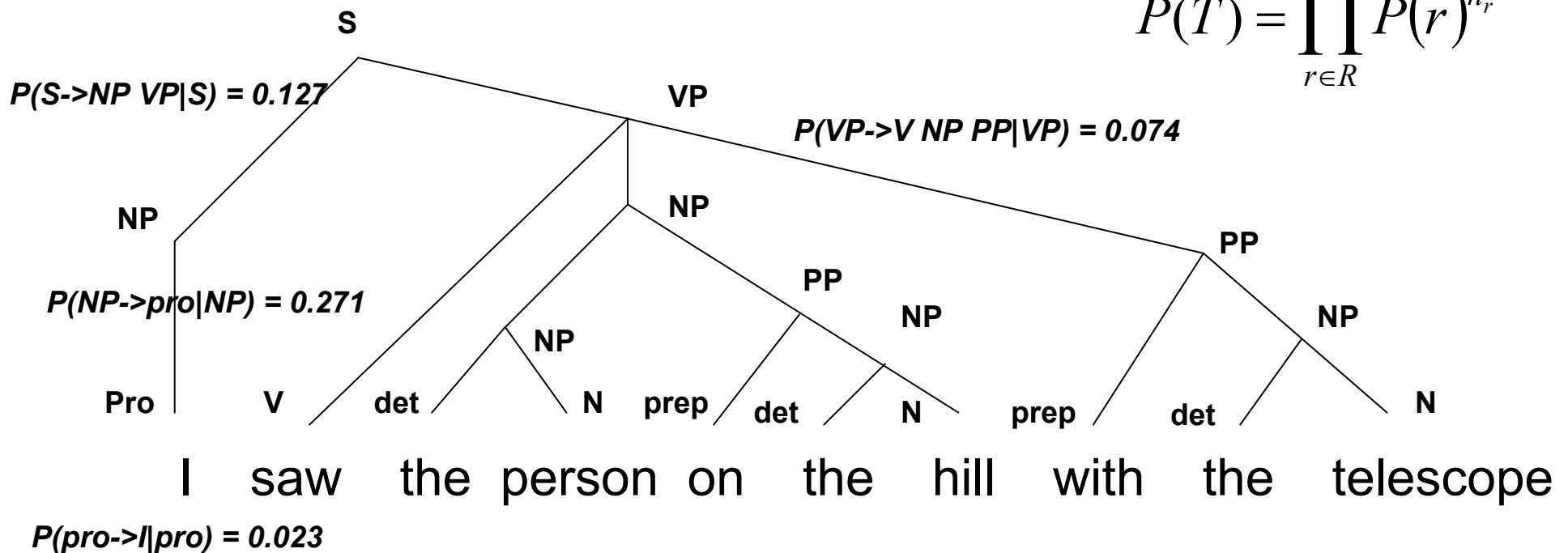


Probabilistic Context-Free Grammars

$P: R \rightarrow [0, 1]$: (Conditional) probability distributions s.t.

$$\forall A \in N: \sum_{A \rightarrow w \in R} P(A \rightarrow w | A) = 1$$

$$P(T) = \prod_{r \in R} P(r)^{n_r}$$

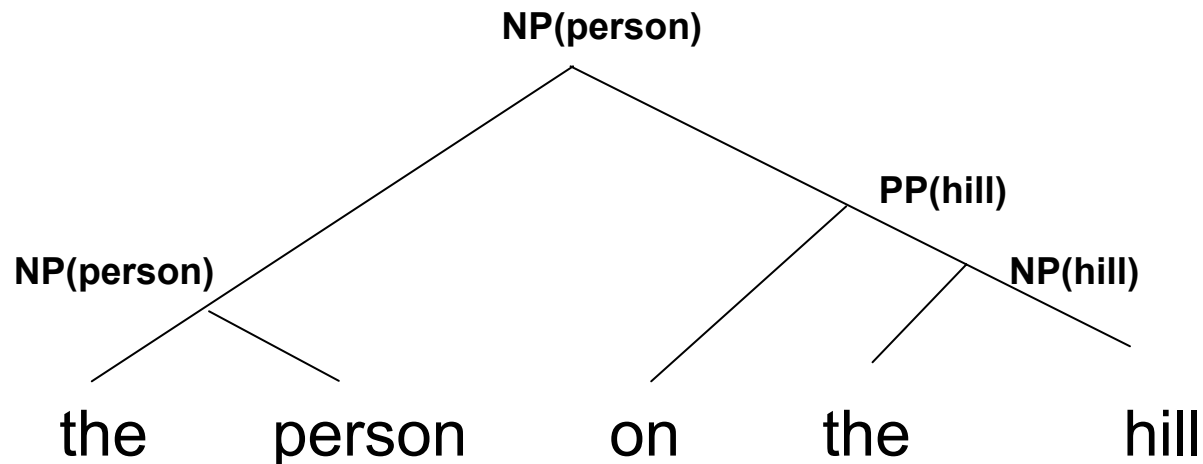


Lexicalised Phrase-Structure Generative Models (Collins '97)

$A(h) \rightarrow L_n(l_n) \dots L_1(l_1) H(h) R_1(r_1) \dots R_m(r_m)$ e.g.: NP(person) \rightarrow NP(person) PP(hill)

$$P(A(h) \rightarrow L_m(l_m) \dots L_1(l_1) H(h) R_1(r_1) \dots R_n(r_n) | A, h) =$$

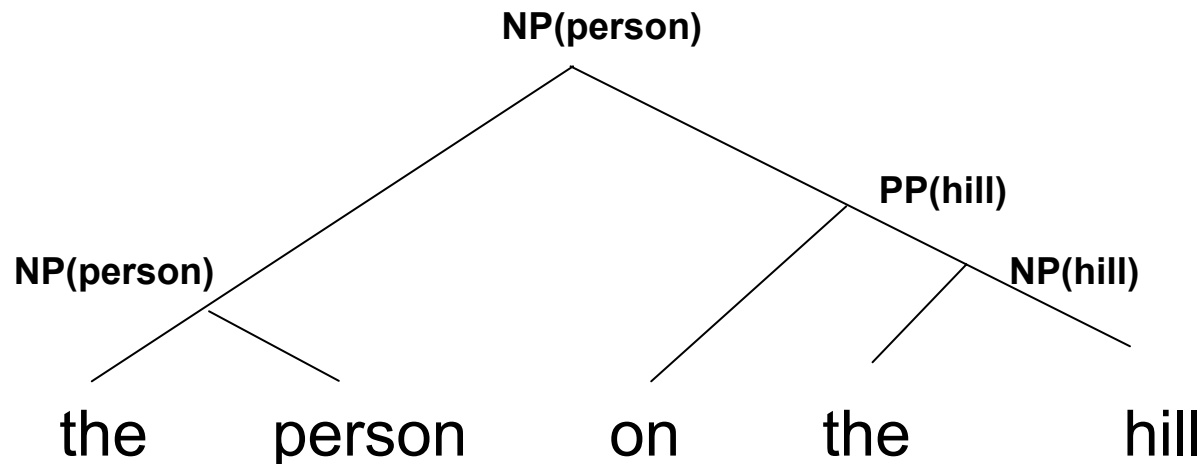
$$= P_H(H | A, h) \prod_{i=1}^{m+1} P_L(L_i(l_i) | A, h, H, \Delta_L(i-1)) \prod_{i=1}^{n+1} P_R(R_i(r_i) | A, h, H, \Delta_R(i-1))$$



Lexicalised Phrase-Structure Generative Models (Collins '97)

$A(h) \rightarrow L_n(l_n) \dots L_1(l_1) H(h) R_1(r_1) \dots R_m(r_m)$ e.g.: $NP(\text{person}) \rightarrow NP(\text{person}) PP(\text{hill})$

$$\begin{aligned} &P(NP(\text{person}) \rightarrow NP(\text{person})PP(\text{hill}) \mid NP, \text{person}) = \\ &= P_H(NP \mid NP, \text{person})P_R(PP(\text{hill}) \mid NP, \text{person}, NP, 0) \cdot \\ &\cdot P_R(STOP \mid NP, \text{person}, NP, 3)P_L(STOP \mid NP, \text{person}, NP, 0) \end{aligned}$$



Kernel-based parse re-ranking

(Collins-Duffy 2001, Shen-Joshi 2003)

- Re-ordering the n-best parses from a statistical model
- Discriminative methods relying on an extended set of features
- Collins-Duffy 2001: $\text{Score}(t) = \text{margin of a linear classifier trained on positive examples (treebank) against all the rest, using a "tree kernel"}$.
- Shen-Joshi 2003: Tree-pairs as instances (along Herbrich-et-al 2000), classifier to choose in $\{<, >\}$.

Shallow Parsing

- Chunking as 3-class tagging (Ramshaw-Marcus 1995):

[I] saw [the person] on [the hill] with [the telescope]
| | | | | | | | | |
B O B I O B I O B I

...then use your favourite tagger:

- Transformation-based error-driven parsing (Brill 93)
- SVM (Kudoh-Matsumoto 2000)
- Memory-Based Learning (Tjong Kim Sang 2002)
- HMMs (Molina-Pla 2002)
- Boosting (Careras et al. 01-02), etc.

Semantic analysis

Semantics (Manning & Schutze 99): study of the meaning of words, constructions and utterances (meaning of individual words - lexical semantics - and combination of individual meanings into meaning of sentences)

Lexical semantics: hyperonymy (*animal;cat*), hyponymy (*cat;animal*), antonymy (*hot;cold*), meronymy (*leaf;tree*), holonymy (*tree;leaf*), synonymy (*car;automobile*), homonymy (*bank*), polysemy (*branch: natural subdivision of a plant; separate but dependent part of a central organisation*)

Word sense disambiguation

Assign senses to words in context

he entered the bank ... he was sitted on the river bank

What is a word sense?

- Adopt the sense definitions in dictionary
- Ask subjects to label instances in a corpus
- Inter-annotator agreement (95% - 65%)
 - likely to be high for ambiguous words with skewed distributions
 - highest disagreements with high-frequency words (types vs tokens; vague dictionary definitions)
 - Co-activation: tis would bring competition to the licensed trade (act and people)

Use of pseudowords

Building test sets is time-consuming and tedious

Artificial evaluation data with pseudowords

replace all occurrences of *banana* and *door* with *banana-door*

evaluate ability of systems to disambiguate between the two meanings

Methods for WSD (1)

Supervised disambiguation

Any categorisation method (naive Bayes, Gale et al. 92)

$$P(s | c), P(c | s) = \prod_{v \in c} P(v | s)$$

Dictionary methods

- Disambiguation based on sense definitions (indicators found in word 's dictionary definitions) (Lesk 86)
- Thesaurus-based desambiguation

Methods for WSD (2)

- Thesaurus-based disambiguation (Walker 87; Yarowski 92)
 - Compute a score for each pair of context c (window of n words) and thesaurus category t ($P(t|c)$)
 - Word categorisation $P(w|t)$: proportion of contexts of w in category t (ex: mouse)
 - Disambiguation proper: for all senses (categories) s of w in c

$$P(s | c) \propto P(s) \prod_{v \in c} P(v | s)$$

- Disambiguation based on translations in a second-language corpus

Methods for WSD (3)

One sense per discourse, one sense per collocation (select strongest collocational feature and disambiguate based only on this feature – Yarowski 94)

Unsupervised disambiguation (Brown et al. 91)

$$P(w, c) = \sum_s P(s) P(w, c | s) \approx \sum_s P(s) P(w | s) P(c | s)$$

$$P(c | s) = \prod_{v \in c} P(v | s)$$

EM algorithm to estimate parameters; MAP disambiguation

Overview of the presentation

1. Natural language processing

1. Morphological analysis
2. Parsing (syntactic analysis)
3. Semantic analysis

2. *Structural analysis*

3. Textual information access

1. Information retrieval
2. Learning representations
3. Categorisation
4. *Text mining*

4. Multilingual aspects

Machine Learning



Information Retrieval

Information Retrieval from user need (query), extract and present to user documents from collections deemed relevant to the query (ranking)

... and related activities ...

Question Answering from user question, extract from documents precise answers

Information Extraction extract information about pre-specified types of events (entities and their relationships) from documents

Vector space representation

Standard bag-of-words model, and associated vector space model

$$\text{sim}(d^{(i)}, d^{(j)}) = \sum_{k=1}^M \frac{w_k^{(i)} w_k^{(j)}}{\sqrt{w_k^{(i)} w_k^{(i)}} \sqrt{w_k^{(j)} w_k^{(j)}}} = \cos(d^{(i)}, d^{(j)})$$

Misses relations between words – straightforward extensions

- Multi-word expressions (terms “linear discriminants”, entities “City Lights”)
- Syntactic relations (SVO triples “Bush_declare_war’s-end”)
- Structural information (title vs. body)

$$\text{sim}(d^{(i)}, d^{(j)}) = \sum_{i=1}^C \alpha_i \cos_{V_{Si}}(d^{(i)}, d^{(j)})$$

Does not directly address synonymy, polysemy, hyp(er)onymy

Latent semantic indexing (LSI)

Standard LSI (Deerwester et al. 90) singular value decomposition of the term-document matrix D

$$D = U\Sigma V \approx U\Sigma_k V$$

Σ_k rank k optimal for L_2 -norm. Documents are then projected onto the reduced latent space (cosine similarity)

Probabilistic LSI (Hofmann 99,00) $P(d, w) = \sum_z P(z)P(d | z)P(w | z)$

$$P = (P(w_i, d_j))_{i,j}, \hat{U} = (P(w_i, z_k))_{i,k}, \hat{V} = (P(d_j, z_k))_{j,k}, \hat{\Sigma} = \text{diag}(P(z_k))_k$$

$$P = \hat{U}\hat{\Sigma}\hat{V}^t$$

Kullback-Leibler divergence; folding-in or Fisher kernels for similarity

Latent semantic indexing (2)

Semantic kernels (Siolas & d'Alché-Buch 00)

Term-term similarity matrix describing semantic proximity (inverse of topological distance in a semantic network, e.g. Wordnet)

$$k(d_1, d_2) = d_1^t P^2 d_2^t, \|Pd_1 - Pd_2\|$$

Used in a Gaussian kernel.

Existing semantic resources vs. acquired ones?

Distributional semantics IR (DSIR, Romaric et al. 03)

Latent semantic kernels (Cristianini et al. 01)

General formulation of similarity between documents in vector spaces; LSI in feature space

Probabilistic retrieval

Large literature on probabilistic approaches

Harter, 75; Bookstein & Swanson, 76; Robertson & Sparck-Jones, 77;
Croft & Harper, 79; Fuhr, 89; Robertson & Walker, 94;
Ponte & Croft, 98; Berger & Lafferty, 99

Binary independence model

$$RSV(d, q) = \log\left(\frac{P(d | R)}{P(d | -R)}\right) = \sum_{j=1}^M I(t_j, d) \left(\log\left(\frac{P(t_j | R)}{P(-t_j | R)}\right) + \log\left(\frac{P(-t_j | -R)}{P(t_j | -R)}\right) \right)$$

Iterative IR through relevance feedback, re-estimation of parameters

$$P(t_j | R) = \frac{r(t_j)}{r}, P(t_j | -R) = \frac{DF(t_j) - r(t_j)}{N - r}$$

Language modeling approach

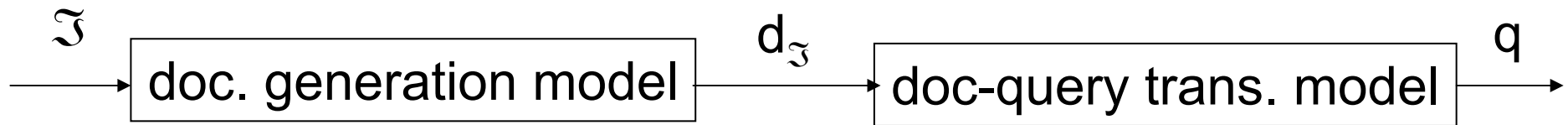
- Focus on query generation probability (not probability of relevance), adequation between indexing and retrieval
- No parametric assumption

$$P(q | d) = \prod_{t \in q} P(t | d) \prod_{t \notin q} (1 - P(t | d))$$

$P(t|d)$ smoothed version of MLE $\hat{P}(t | d)$

Does not directly address synonymy, polysemy, hyp(er)onymy

Noisy channel approach



$$\rho_d(d) = P(q | d, U)P(d | U)$$

$P(d|U)$ measures quality of document wrt user's preferences (length, languages). A first model, model 1 (cf. Brown et al. 93), consists in:

$$P(q | d) = P(m | d) \prod_{j=1}^m (n(n+1)^{-1} \sum_w P(q_j | w)P(w | d) + (n+1)^{-1} P(q_j | \langle null \rangle))$$

Estimation of the translation probabilities $P(q_j|w)$. Lacking sufficient training data, use of synthetic queries to generate training corpus, then learn parameters with EM.

Topography of collections

- Documents and collections more and more structured
- Need to rely on models able to take this information into account (Google; VSM misses relations between elements)
 - Piwowarski & Gallinari 03
 - Denoyer et al. 03 (classification)
 - Lalmas 00
 - Fuhr & Rölleke 98

INEX (Initiative for the Evaluation of XML Retrieval)

<http://inex.is.informatik.uni-duisburg.de:2003/>

Learning document representations (1)

- Story segmentation, topic detection and tracking (Fiscus et al. 98) - topic is seminal event or activity, with related events or activities (TDT)
 - story segmentation: segmenting stream of data into cohesive topics
 - topic detection: identify new topics
 - topic tracking: associate incoming stories with known topics

Applications in IR, speech recognition (topic-based language modeling)

Hearst 97 (TextTiling)

Beeferman et al. 00 (long- and short-range language models)

Brants et al. 02 (PLSA)

Learning document representations (2)

1. Defining elementary blocks in documents: sentences
2. Identify features and associated decision function to determine whether or not segment boundary between blocks
 - topicality features (word distributions)
 - cue-word features (Mr. in WSJ)

W. E. Coyote, president of ACME ... [.] Mr. Coyote

Learning document representations (3)

- Inferring topic/thematic hierarchies: induce themes (and their dependencies) from document collections
 - Applications: IR, browsing functionalities, SR, language models
 - Approach
 - identify elementary units (document)
 - hierarchical clustering of units to identify themes and their dependencies
 - from hierarchical clusters to category systems/taxonomies/ontologies
- (Kashyap et al. 04; Gaussier et al. 02; M. Keller 04; Illouz & Jardino, 01; D. Lin 98; H. Nikje-Fotzo 04; Sanderson 99)

Categorisation (1)

Specifics properties of texts

1. Words are ordered
2. Mainly short-range dependencies (syntactic and semantic)
3. Few irrelevant features

Are linear or polynomial kernels appropriate: fail to account for points 1 & 2

Word-sequence kernels - string kernels (Watkins 99; Lodhi et al. 01-02; Cancedda et al. 03)

$$K_n(s, t) = \langle \Phi(s), \Phi(t) \rangle, \Phi_u(s) = \sum_{i:s[i]=u} \lambda^{l(i)}$$

u ordered subsequence of Σ^n , λ decay factor (non-contiguity), $l(i)$ length spanned by $s[i]$ in s

Categorisation (2)

Word-sequence kernels on the Reuters collection - Results similar to linear and polynomial kernels

1. Word order has very little effect on performance
2. Locality has no impact

Major problem for machine learning in textual information access: **the data annotation bottleneck**

Example: biological entity recognition. Need biologists annotate protein, gene, RNA names in texts

- task is time consuming
- inter-annotator agreement low (iterative annotation)

Categorisation (3)

Need to rely on methods able to leverage small amounts of annotated data with large amounts of unannotated data

- mixture models with partially labeled data (Miller & Uyar 97)
- transductive inference (Joachims 99)
- co-training (Blum & Mitchell 98)
- application to biological entity recognition (Goutte et al. 02-04)
- application to coreference resolution (Ng & Cardie 03)

Overview of the presentation

1. Natural language processing

1. Morphological analysis
2. Parsing (syntactic analysis)
3. Semantic analysis

2. *Structural analysis*

3. Textual information access

1. Information retrieval
2. Learning representations
3. Categorisation
4. *Text mining*

4. Multilingual aspects

Machine Learning



Why process multilingual collections?

Textual information in large, international companies and organizations is multilingual

EPO, JPO, EC, Canada, Switzerland, Xerox, ...

Competition is international, and so is technology watch

Need to store, categorize, filter, search, browse, mine such collections

Delphi report: IT users spend up to 30% of their time looking for information that should have been easier to find

Comprehension aids/translation of (selected) documents

General Characteristics

Parallel vs comparable collection

- Translation memories vs newspapers articles
- Most general case: partly parallel, partly comparable (Nie 1999)

Words, terms, entities (proper names, biological names): Synonymy, polysemy, hyponymy, hyperonymy

light, green light, City Lights

Information access functionalities

- Performance equivalence across languages
- Consistency across languages
- Benefit from other languages (little training data)

Feature extraction in multilingual collections (1)

Illustration from an example: adjacent pairs

text mining (*fouille de donnees*)

→ Explicit extraction of terms, *NN*, *Adj N*, and say some *SUBJ/VERB* and *VERB/OBJ* relations

→ French equivalent: *N Adj*, *N (Adj) prep (det) (Adj) N (prep = a, de, dans, en ,par)*, plus *SUBJ/VERB* and *VERB/OBJ* relations

The two sets of patterns lead to different units (level of permissible interference – *niveau de brouillage admissible*)

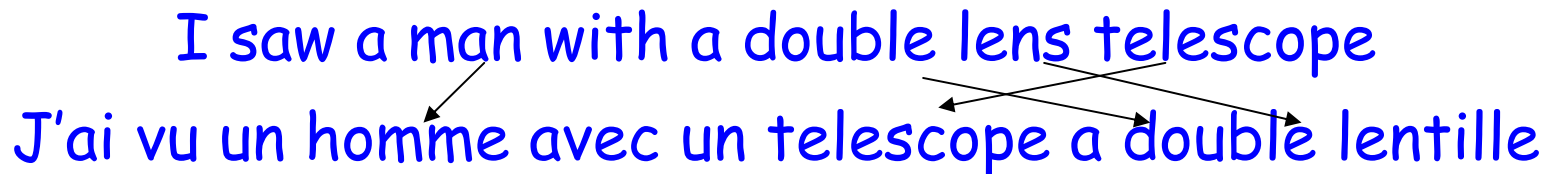
→ No complete equivalence between tools performing the same task on different languages (few exceptions)

Feature extraction in multilingual collections (2)

Parallel corpora

Extended parse-parse (Debili, 95; Hull 99); One way parsing (Gaussier 98); Parallel parsing (Wu 97)

I saw a man with a double lens telescope
J'ai vu un homme avec un telescope a double lentille



Comparable corpora

Dimension of parameter space increases wrt parallel corpora

Word to word alignment not satisfactory yet

What if no tools for a given language: translingual inference through parallel alignment

Crossing the language barrier (1)

Parallel collection in IR

- “Artificial documents”, indexed by two languages
- Interlingual latent semantic analysis – Littman et al. 00 (interlingual probabilistic latent semantic analysis)

$$Sim(d, q) = d^t U I_k U^t q$$

- Monolingual mapping to language independent space (GVSM, Yang et al. 98; LSI, bilingual mapping (Jiang & Littman 00))

$$Sim(d, q) = d^t U_A I_k U_B^t q$$

Crossing the language barrier (2)

Parallel collection in IR

- Inferring a cross-lingual semantic representation (kernel canonical correlation analysis – Vinokourov et al. 02)

$$Sim(d, q) = d^t P_1 P_2^t q, (c_1^1, c_1^2) \dots (c_r^1, c_r^2)$$

- Extracting bilingual corpora from parallel corpora (Veronis 00). Use in CLIR (Brown et al. 00)
 - additional constraints (parallelism at the sentence level)
 - bilingual lexicons perform both query translation and expansion
 - best performance (k-CCA never compared)

Crossing the language barrier (3)

Comparable collection

- Combination with lexicons extracted from parallel corpora (Renders et al. 03)
- Extension of previous methods to comparable corpora?
- Bilingual lexicon extraction from comparable corpora not satisfying yet

Bilingual extraction from comparable corpora

Hyp: if two words are translation of each other, then their collocates are likely to be translations of each other as well

(collocation defined as co-occurrence relation; implicit assumption: domain specific word co-occur with general words (hence possibility to use general bilingual dictionaries))

Step 1: build context vectors for source and target word (based on co-occurrence)

Leber: Transplantation, 138; Resektion, 53; Metastase, 41; Arterie, 38; Cirrhose, 26; ...

Step 2: translate context vectors (say from target to source language)

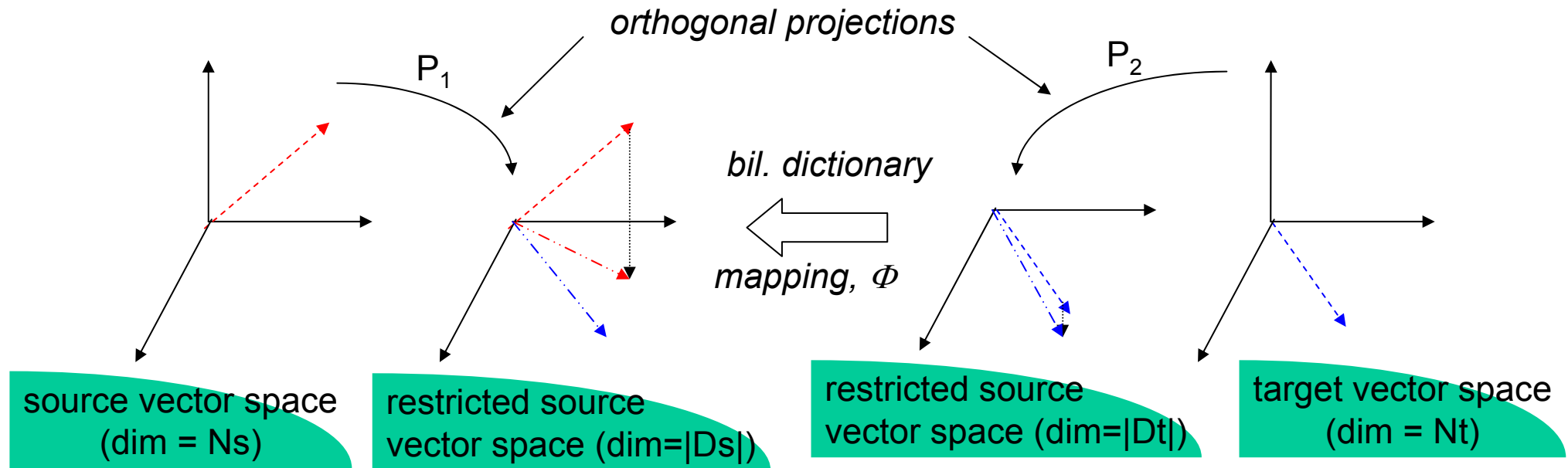
Leber: transplant, 138; metastase, 41; artery, 38; cirrhosis, 26; ...

Step 3: compute similarity between source and translated target context vectors

Liver: transplant, 140; tumour, 100; cirrhosis, 70; artery, 40; ...

Leber: transplant, 138; metastase, 41; artery, 38; cirrhosis, 26; ...

Geometric view



$$\text{sim}(s,t) = f(\langle P_1 s, \Phi(P_2 t) \rangle)$$

Remark: (a) projection of s (for s in D_s) may signif. differ from its corresponding axis; (b) synonymy and polysemy problems

Statistical machine translation (1)

From existing translations (parallel corpora), learn a translation model
(Brown et al. 93; Och et al. 99)

The spirit is willing, but the flesh is weak.
L'esprit est fort, mais la chair est faible. ∅

Given a French sentence of length m , find the English string e for which $P(e|f)$ is maximal, i.e. for which $P(e)(f|e)$ is maximal

$$P(f | e) = \sum_a P(f, a | e), P(f, a | e) = P(m | e) \prod_{j=1}^m P(f_j | e_{a_j})$$

Statistical machine translation (2)

Estimation via EM algorithm (unique maximum for model 1 above)

Model 2 to 5 refine the translation model by introducing alignment probabilities, fertility and distortion probabilities)

1-to-n assumption; removed in recent models (alignment template)

Once parameters have been estimated, translating amounts to decoding: huge search space, several heuristics possible

Conclusion & Perspectives

Most ML work on text analysis adopt a BOW representation: is it sufficient? can we do better?

Most tasks addressed involve/require knowledge with minimal formalisation

- Question answering require advanced knowledge, and reasoning capabilities
- Semantic web

Annotation and inter-annotator agreement

- Acute need for annotated data, but: data annotation bottleneck, lack of inter-annotator agreement for semantic-related tasks

Evaluation is of crucial importance (new evaluation paradigms needed)

Conclusion

Machine (aided) translation (Holy Grail of Computational Linguistics)

No ifs, thens or but(**tl**er)s, we want the truth.

Cessons de tourner autour du pot ... **de chambre**.

buttler - valet de chambre

chamber pot - pot de chambre