Natural Language Processing
Summary

Miquel Perelló Nieto

November 4, 2012

Contents

Chapter 1

Introduction

Chapter 2

Aplications of Natural Language Processing

Chapter 3

Statistical Models of Language

3.0.1 Probability Theory

3.0.2 Ngram model

1-gram

$P_{MLE}(w) = \frac{C(w)}{|V|}$

2-gram

$P_{MLE}(w_i|w_{i-1}) = \frac{C(w_{i-1}w_i)}{C(w_{i-1})}$

3-gram

$P_{MLE}(w_i|w_{i-1}, w_{i-2}) = \frac{C(w_{i-2}w_{i-1}w_i)}{C(w_{i-2}w_{i-1})}$

3.0.3 Corpora

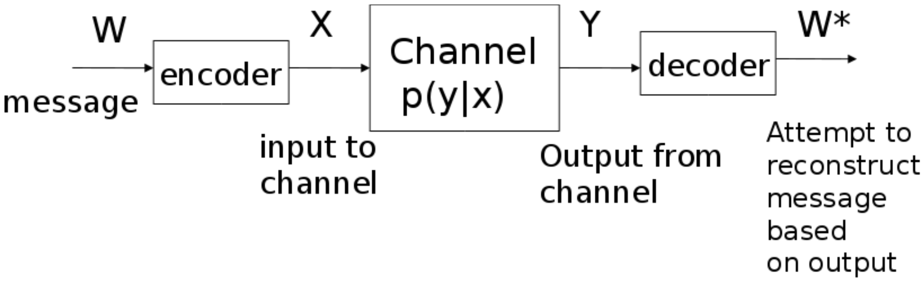
- Colocations
- Argument structure.
- Frecuency information
- Context
- Grammatical Induction
- Probabilistic Analysis.
- Lexical relations
- Examples of use.
- Selectional Restrictions
- Nominal compounds

- Idioms, ...

Training and Test sets

3.0.4 Language models

3.0.5 Noisy channel models



Example: Automatic Speech Recognizer

Example: Machine Translation

3.0.6 Laplace

$P_{laplace}(w_1 \cdots w_n) = \frac{C(w_1 \cdots w_n)+1}{N+B}$

P = probability of an n-gram

C = counting of the n-gram in the training corpus

N = total of n-grams in the training corpus

B = parameters of the model (possible n-grams)]

3.0.7 Lidstone (Laplace generalization)

3.0.8 Held-Out

1. Compute the percentage of the probability mass that has to be reserved for the n-grams unseen in the training corpus
2. We separate from the training corpus a held-out corpus
3. We compute howmany n-grams unseen in the training corpus occur in the held-out corpus
4. An alternative of using a held-out corpus is using Cross-Validation
   - Held-out interpolation
   - Deleted interpolation

$P_{ho}(w_1 \cdots w_n) = \frac{T_r}{N_r N}$

- Let a n-gram w1... wn

- r = C(w1... wn)
- C1(w1... wn) counting of the n-gram in the training set
- C2(w1... wn) counting of the n-gram in the held-out set
- Nr number of n-grams with counting r in the training set

3.0.9 Good-Turing

$r = (r+1)\frac{E(N_{r+1})}{E(N_r)}$ $P_{GT} = \frac{r}{N}$

- r* = adjusted count
- Nr = number of n-gram-types occurring r times
- E(Nr) = expected value
- E(Nr+1) ¡ E(Nr)
- Zipf law

3.0.10 Linear combination (Interpolation)

- Linear combination of de 1-gram, 2-gram, 3-gram, ...
- Estimation of λ using a development corpus

3.0.11 Katz's Backing-Off

1. Start with a n-gram model
2. Back off to n-1 gram for null (or low) counts
3. Proceed recursively

Chapter 4

Information Theory

4.1 Entropy

Related to the coding theory- more efficient code: fewer bits for more frequent messages at the cost of more bits for the less frequent

4.1.1 Ecample

EXAMPLE: You have to send messages about the two occupants in a house every five minutes

Equal probability:

0	no occupants
1	first occupant
2	second occupant
3	Both occupants

Different probability

Situation	Probability	Code
no occupants	.5	0
first occupant	.125	110
second occupant	.125	111
Both occupants	.25	10

- Let X a random variable taking values x1, x2, ..., xn from a domain de according to a probability distribution
- We can define the expected value of X, E(x) as the summatory of the possible values weighted with their probability
- E(X) = p(x1)X(x1) + p(x2)X(x2) + ... p(xn)X(xn)
- A message can thought of as a random variable W that can take one of several values V(W) and a probability distribution P.
- Is there a lower bound on the number of bits neede tod encode a message? Yes, the entropy

- It is possible to get close to the minimum (lower bound)
- It is also a measure of our uncertainty about wht the message says (lot of bits- uncertain, few - certain)

Chapter 5

Lexical Processing

5.1 Superficial level

- Getting the document(s)
  - Accessing a BD
  - Accessing the Web (wrappers)
- Getting the textual fragments of a document
  - Multimedia documents, Web pages, ...
- Filtering out meta-information
  - tags HTML, XML, ...
- Text segmentation into paragraphs or sentences Tokenization
  - orthographic vs grammatical word
  - Multiword terms (dates, formulas, acronyms, abbreviations, quantities (and units), idioms)
  - Named entities (NER, NEC, NERC)
  - Unknown word
- Language identification

Vocabulary size (V)

- Heap's Law
- V = KN
- K depends on the text $10 \leq K \leq 100$
- N total number of words
- $\forall \beta$ depends on the language, for English $0.4 \leq \beta \leq 0.6$
- Vocabulary grows sublinealy but does not saturate

- $\forall\beta$ tends to stabilize for 1Mb of text (150.000w)
- word tokens vs word types
- Statistical distribution of words in a document
  - Obviously non uniform
  - Most common words cover more than 50% of occurrences
  - 50% of the words only occur once
  - $\sim 12\%$ of the document is formed by word occurring less than 4 times.

Zipf law:
We sort the words occurring in a document by their frequency. The product of the frequency of a word (f) by its position (r) is aproximatelly constant

5.2 Lexical level

- Part of Speech (POS)
  - Formal property of a word-type determining its acceptable uses in syntax.
- A POS can be seen as a class of words
- A word-type can own several POS, a word-token only one
- Plain categories
  - open, many elements, neologisms, independent and semantically rich classes
  - N, Adj, Adv, V
- Functional categories
  - closed

Lexicon

- Repository of lexical information for human or computer use
- Two aspects to consider
  - Representation of lexical information
  - Acquisition of lexical information
- Orthografic Transcription
- Phonetic Transcription
- Flexion model
- diathesis alternations, subcategorization frames
  - LOVE VTR (OBJLIST: SN).
  - LOVE

* CAT = VERB
* $SUBCAT = \langle SN, SN \rangle$

- POS
- Argument structure
- Semantic information
    - dictionaries → definition
    - lexicons → semantic types predefined in a hierarchy.
- Lexical Relations
    - derivation
- Equivalence with other languages

Chapter 6

Finite State Models

6.1 Regular expressions

Standard notation for characterizing text sequences, can be implemented by finite-state automaton.

Aplications:

- Morphology
- Phonology
- Lexical generation
- ASR
- POS tagging
- simplification of CFG
- Information Extraction

6.2 Finit State Automaton

$\langle \Sigma, Q, i, F, E \rangle$

Alphabet	Σ
Finite set of states	Q
Initial state	$i \in Q$
Final states set	$F \subseteq Q$
Arc set	$E \subseteq Q \times (\Sigma \cup \{\varepsilon\}) \times Q$
Transition set	$E : \{d \vert d : Q \times (\Sigma \cup \{\varepsilon\}) \to 2^Q\}$

6.2.1 Closure

- Union
- Intersection
- Concatenation

- Complement
- Kleene star(A*)

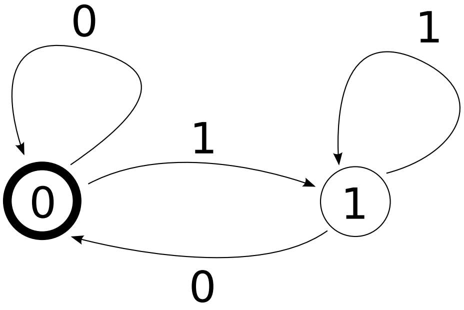


Figure 6.1: Recognize multiples of two

6.3 Finit State Transducer

$\langle \Sigma_1, \Sigma_2, Q, i, F, E \rangle$

Input alphabet	Σ_1
Output alphabet	Σ_2
Finite set of states	Q
Initial state	$i \in Q$
Final states set	$F \subseteq Q$
Arc set	$E \subseteq Q \times (\Sigma_1^\star \times \Sigma_2^\star) \times Q$

6.3.1 Closure

- Union
- Invertion ($Td3^{-1} = Td3$)
- Composition ($Td9 = Td3 \times Td3$)

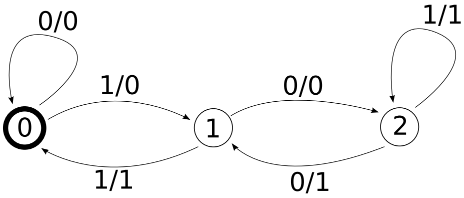


Figure 6.2: Returns the division by three

6.4 FSA associated to a FST

- $FSA\langle \Sigma, Q, i, F, E' \rangle$
- $FST\langle \Sigma_1, \Sigma_2, Q, i, F, E \rangle$
- $\Sigma = \Sigma_1 \times \Sigma_2$
- $(q_1, (a, b), q_2) \in E' \Leftrightarrow (q_1, a, b, q_2) \in E$

6.5 FST projections

- $FSTT = \langle \Sigma_1, \Sigma_2, Q, i, F, E \rangle$
- First projection
  - $P_1(T)\langle \Sigma_1, Q, i, F, E_{P1} \rangle$
  - $E_{P1} = \{(q, a, q') | (q, a, b, q') \in E\}$
- Second projection
  - $P_2(T)\langle \Sigma_2, Q, i, F, E_{P2} \rangle$
  - $E_{P2} = \{(q, b, q') | (q, a, b, q') \in E\}$

Chapter 7

Morphology

- Morphology
  - Structure of a word as a composition of morphemes
  - Related to word formation rules
  - Functions
    - Flexion
    - Derivation
    - Composition
- Result of morphologic analysis
  - Morphosyntactic categorization (POS)
    - e.g. Parole tagset (VMIP1S0), more than 150 categories for Spanish
    - e.g. Penn Treebank tagset (VBD), about 30 categories for English
  - Morphological features
    - Number, case, gender, lexical functions
- Morphologic analysis
  - Decompose a word into a concatenation of morphemes
  - Usually some of the morphemes contain the meaning
    - One (root or stem) in flexion and derivation
    - More than one in composition
  - The other (affixes) provide morphological features
- Problems
  - Phonological alterations in morpheme concatenation
  - Morphotactics
    - Which morphemes can be concatenated with which others
  - Affixes
    - Suffixes, prefixes, infixes, interfixes

  - flexive Affixes ≠ derivative Affixes
  - Derivation implies sometimes a semantic change not always predictible
    - Meaning extensions
    - Lexical rules
  - A derivative suffix can be followed by a flexive suffix
    - $love \Rightarrow lover \Rightarrow lovers$
  - Flexion does not change POS, sometimes derivation does
  - Flexion affects other words in the sentence
    - agreement
- Morphotactics
  - Word formation rules
  - Valid combinations between morphemes
    - Simple concatenation
    - Complex models root/pattern
    - Regularity language dependent
- Phonological alterations (Morphophonology)
  - Changes when concatenating morphemes
  - Source: Phonology, morphology, orthography
  - variable in number and complexity
  - e.g. vocalic harmony

7.1 Morphemes

1 morpheme : avoid = avoid

2 morphemes : avoidable = avoid + able

3 morphemes : unavoidable = un + avoid + able

4 morphemes : unavoidability = un + avoid + able + ility

7.2 Flexive morphology

number : house, houses

verbal form : walk, walkes, walked, walking

gender : niño, niña (Spanish)

7.3 Derivative morphology

Form : **Prefix** inavoidable

Suffix avoidable

Source : $verb \Rightarrow adjective\ play \Rightarrow playfull$

$verb \Rightarrow noun\ play \Rightarrow player$

$adjective \Rightarrow adjective\ red \Rightarrow reddish$

7.4 Types of morphological analyzers

- Formaries (dictionaries of words forms)
  - efficency
  - languages with few variants
  - extensibility
  - possibility of building and maintenance from a morphological generator
  - languages with high flexive variation
  - derivation, composition
- FS techniques
  - FSA : 1 level analyzers
  - FST : >1 level analyzers

Input (form)	**Output (lemma + morphological features)**
cat	cat + N + sg
cats	cat + N + pl
cities	city + N + pl
merging	merge + V + pres_part
caught	(catch + V + past) or (catch + V + past_part)

Table 7.1: Morphological analysis examples

7.4.1 Example

superficial level : foxes

Spelling rules $FST_1 FST_2 \cdots FST_n$

intermediate level : fox's

Lexical $Lexicon - FST$

lexical level : fox +N +pl

Chapter 8

Hidden Markov models

From Wikipedia, the free encyclopedia

A hidden Markov model (HMM) is a statistical Markov model in which the system being modeled is assumed to be a Markov process with unobserved (hidden) states. An HMM can be considered as the simplest dynamic Bayesian network. The mathematics behind the HMM was developed by L. E. Baum and coworkers.[1, 2, 3, 4, 5] It is closely related to an earlier work on optimal nonlinear filtering problem (stochastic processes) by Ruslan L. Stratonovich,[6] who was the first to describe the forward-backward procedure.

In simpler Markov models (like a Markov chain), the state is directly visible to the observer, and therefore the state transition probabilities are the only parameters. In a hidden Markov model, the state is not directly visible, but output, dependent on the state, is visible. Each state has a probability distribution over the possible output tokens. Therefore the sequence of tokens generated by an HMM gives some information about the sequence of states. Note that the adjective 'hidden' refers to the state sequence through which the model passes, not to the parameters of the model; even if the model parameters are known exactly, the model is still 'hidden'.

Hidden Markov models are especially known for their application in temporal pattern recognition such as speech, handwriting, gesture recognition,[7] part-of-speech tagging, musical score following,[8] partial discharges[9] and bioinformatics.

A hidden Markov model can be considered a generalization of a mixture model where the hidden variables (or latent variables), which control the mixture component to be selected for each observation, are related through a Markov process rather than independent of each other.

Chapter 9

Tagging

The goal of a POS tagger is to assign each word the most likely within a context.

Sequence of words : $W = w_1 w_2 \cdots w_n$

Sequence of POS tags : $T = t_1 t_2 \cdots t_n$

For each word wi only some of the tags can be assigned (except the unknown words). We can get them from a lexicon or a morphological analyzer.
$f : W \rightarrow T = f(W)$

9.1 Rule-based taggers

- Knowledge-driven taggers
- Usually rules built manually
- Limited amount of rules (≈ 1000)
- LM and smoothing explicitely defined.
- pros
  - Linguistically motivated rules
  - High precision (99.5%)
- cons
  - High development cost
  - Not transportable
  - Time cost of tagging

9.2 Statistical POS taggers

- LM and smoothing automatically learned from tagged corpora (supervised learning).
- Data-driven taggers
- Statistical inference
- Techniques coming from speech processing
- pros
  - Well founded theoretical framework
  - Simple models.
  - Acceptable precision ($>97\%$)
  - Language independent
- cons
  - Learning the model (Sparseness)
  - less precision

9.2.1 N-grams

$$argmaxP(t_1, \ldots, t_n | w_1, \ldots, w_n) \approx \prod_{k=1}^{n} P(t_k | t_{k-2}, t_{k-1}) \times P(w_k | t_k)$$

$P(t_k | t_{k-2}, t_{k-1})$ Contextual probability (trigrams)

$P(w_k | t_k)$ Lexical probability

9.2.2 Hidden Markov Models tagger

- Hidden States associateds to n-grams
- Transition probabilities restricted to valid transitions ($*BC \rightarrow BC*$)
- Emision probabilities restricted by lexicons

9.2.3 Hybrid systems

- Transformation-based, error-driven (Brill, 1995)(Roche, Schabes, 1995)
  - Based on rules automatically acquired
- Maximum Entropy (Ratnaparkhi, 1998)(Rosenfeld, 1994)(Ristad, 1997)
  - Combination of several knowledge sources
  - No independence is assumed
  - A high number of parameters is allowed

Brill's system

- Based on transformation rules that corerect errors produced by an initial HMM tagger
- rule
  - change label A into label B when ...
  - Each rule corresponds to the instantiation of a template
- templates
  - The previous (following) word is tagged with Z
  - One of the two previous (following) words is tagged with Z
  - The previous word is tagged with Z and the following with W
  - ...
- Learning of the variables A,B,Z,W through an iterative process That choose at each iteration the rule (the instanciation) correcting more errors.

Bibliography

[1] Baum, L. E.; Petrie, T. (1966). "Statistical Inference for Probabilistic Functions of Finite State Markov Chains". The Annals of Mathematical Statistics 37 (6): 1554–1563. doi:10.1214/aoms/1177699147. Retrieved 28 November 2011.

[2] Baum, L. E.; Eagon, J. A. (1967). "An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology". Bulletin of the American Mathematical Society 73 (3): 360. doi:10.1090/S0002-9904-1967-11751-8. edit

[3] Baum, L. E.; Sell, G. R. (1968). "Growth transformations for functions on manifolds". Pacific Journal of Mathematics 27 (2): 211–227. Retrieved 28 November 2011.

[4] Baum, L. E.; Petrie, T.; Soules, G.; Weiss, N. (1970). "A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains". The Annals of Mathematical Statistics 41: 164. doi:10.1214/aoms/1177697196. edit

[5] Baum, L.E. (1972). "An Inequality and Associated Maximization Technique in Statistical Estimation of Probabilistic Functions of a Markov Process". Inequalities 3: 1–8.

[6] a b Stratonovich, R.L. (1960). "Conditional Markov Processes". Theory of Probability and its Applications 5: 156–178.

[7] Thad Starner, Alex Pentland. Real-Time American Sign Language Visual Recognition From Video Using Hidden Markov Models. Master's Thesis, MIT, Feb 1995, Program in Media Arts

[8] B. Pardo and W. Birmingham. Modeling Form for On-line Following of Musical Performances. AAAI-05 Proc., July 2005.

[9] Satish L, Gururaj BI (April 2003). "Use of hidden Markov models for partial discharge pattern classification". IEEE Transactions on Dielectrics and Electrical Insulation.