

Foundations of Statistical Natural Language Processing

劉成韋

OUTLINE

- Introduction
- Methodological Preliminaries
- Supervised Disambiguation
- Dictionary-based Disambiguation
- Unsupervised Disambiguation
- What is a Word Sense?
- Further Reading

Introduction (1/2)

- Problem
 - A word is assumed to have a finite number of discrete senses.
- Task
 - To make a forced choice between these senses for the meaning of each usage of an ambiguous word.
 - Based on the context of use.
- In fact
 - A word has various somewhat related senses, but it is unclear whether to and where to draw lines between them.

Introduction (2/2)

- However, the senses are not always so well-defined.
- For Example : **bank**
 - The rising ground bordering a lake, river, or sea...(邊坡)
 - As establishment for the custody(保管), loan exchange, or issue of money, for the extension of credit, and for facilitating the transmission of funds.(銀行)

Methodological Preliminaries (1/3)

- Supervised learning
 - Know the actual status for each piece of data on which one train
 - Can usually be seen as a classification task
- Unsupervised learning
 - Don't know the classification of the data in the training example
 - Can thus often be viewed as a clustering task.

Methodological Preliminaries (2/3)

Pseudo-words

- For testing the performance of these algorithms
 - Large number of occurrences has to be disambiguated by hand
 - Time intensive
 - Laborious task
- Pseudo-words
 - Conflating two or more words
 - Such as replaces all banana and door in a corpus by banana-door

Methodological Preliminaries (3/3)

Upper and lower bounds on performance

- The estimation of upper and lower bounds
 - A way to make sense of performance figures
 - A good idea for those which have no standardized evaluation sets for comparing systems.
- The upper bound used is usually human performance
 - We can't expect an automatic procedure to do better
- The lower bound is the performance of the simplest possible algorithm
 - Assign all contexts to the most frequent sense

Supervised Disambiguation

- A disambiguated corpus is available for training
 - There is a training set where each occurrence of the ambiguous word is annotated with a semantic label
- Bayesian classification
 - < Gale et al. 1992 >
 - Treats the context of occurrence as a bag of words without structure
- An information-theoretic approach
 - < Brown et al. 1991 >
 - Looks at only one informative feature in the context, which may be sensitive to text structure

Supervised Disambiguation

Bayesian Classification (1/4)

- Each context word
 - Contributes potentially useful information about which sense of the ambiguous word is likely to be used with it
- A Bayes classifier applies the Bayes decision rule when choosing a class
 - For each cases, choose the class with the highest prob.
 - The rule minimize the probability of error
- Bayes decision rule

$$P(s'|c) > P(s_k | c) \quad \text{for } s_k \neq s'$$

Supervised Disambiguation

Bayesian Classification (2/4)

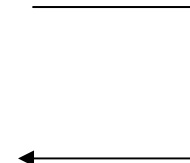
- We want to assign the ambiguous word w to the sense s' , given the context c

$$s' = \arg \max_{s_k} P(s_k | c)$$

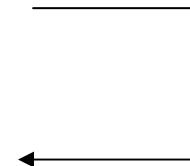
$$= \arg \max_{s_k} \frac{P(c | s_k) P(s_k)}{P(c)}$$

$$= \arg \max_{s_k} P(c | s_k) P(s_k)$$

$$= \arg \max_{s_k} [\log P(c | s_k) + \log P(s_k)]$$



Baye's Rule



log

Supervised Disambiguation

Bayesian Classification (3/4)

- Gale et al.'s classifier, the Naïve Bayes classifier
 - An instance of a particular kind of Bayes classifier
- Naïve Bayes assumption

$$P(c | s_k) = P(\{v_j | v_j \text{ in } c\} | s_k) = \prod_{v_j \text{ in } c} P(v_j | s_k)$$

- All the context and linear ordering of words is ignored
- Each word is independent of another
 - Actually it's not true, such as "president"
- The simplifying assumption makes it more effective

Supervised Disambiguation

Bayesian Classification (4/4)

- With the Naïve Bayes assumption :
 - Decision rule for Naïve Bayes
 - decide s' if

$$s' = \operatorname{argmax}_{s_k} [\log P(s_k) + \sum_{v_j \text{ in } c} \log P(v_j | s_k)]$$

$$P(v_j | s_k) = \frac{C(v_j, s_k)}{C(s_k)} \quad P(s_k) = \frac{C(s_k)}{C(w)}$$

- Choose $s' = \operatorname{argmax}_{s_k} \text{score}(s_k)$

Supervised Disambiguation

An information-theoretic approach (1/5)

- It tries to find a single contextual feature that reliably indicates which sense of the ambiguous word is being used.
 - Instead of use information from all words in the context, such as Bayes classifier

Ambiguous word	Indicator	Examples: value → sense
prendre	object	<i>measure</i> → <i>to take</i> <i>décision</i> → <i>to make</i>
vouloir	tense	present → <i>to want</i> conditional → <i>to like</i>
cent	word to the left	<i>per</i> → % number → <i>c.</i> [money]

Table 7.3 Highly informative indicators for three ambiguous French words.

Supervised Disambiguation

An information-theoretic approach (2/5)

- Two senses of the word prendre
 - Prendre une mesure → take a measure
 - Prendre une decision → make a decision
- Flip-Flop algorithm <Brown et al.>
 - Let $\{t_1, \dots, t_m\}$ be the translations of the ambiguous word
 - Let $\{x_1, \dots, x_m\}$ be the possible values of the indicator
 - For prendre $\{t_1, \dots, t_m\}$ → {take, make, rise, speak}
 - For prendre $\{x_1, \dots, x_m\}$ → {measure, note, exemple, decision, parole}

Supervised Disambiguation

An information-theoretic approach (3/5)

- Flip-Flop Algorithm :
 - find a random partition $P = \{P_1, P_2\}$ for $\{t_1, \dots, t_m\}$
 - while (improving) do
 - find partition $Q = \{Q_1, Q_2\}$ of $\{x_1, \dots, x_n\}$
 - that maximizes $I(P; Q)$
 - find partition $P = \{P_1, P_2\}$ of $\{t_1, \dots, t_m\}$
 - that maximizes $I(P; Q)$
 - end
- Each iteration of the algorithm increases the mutual information $I(P; Q)$ monotonically.

$$I(P; Q) = \sum_{t \in P} \sum_{x \in Q} p(t, x) \log \frac{p(t, x)}{p(t)p(x)}$$

Supervised Disambiguation

An information-theoretic approach (4/5)

- The initiation partition p
 - $P1 = \{\text{take, rise}\}$ $P2 = \{\text{make, speak}\}$
- Let's assume prendre is translated by take, so
 - $Q1 = \{\text{measure, note, exemple}\}$
 - $Q2 = \{\text{decision, parole}\}$
 - Since this partition will maximize $I(P;Q)$
- The 2nd partition p
 - $P1 = \{\text{take}\}$ $P2 = \{\text{male, speak, rise}\}$

Supervised Disambiguation

An information-theoretic approach (5/5)

- Disambiguation
 - For the occurrence of the ambiguous word, determine the value of the indicator
 - If the value is in Q_1 , assign the occurrence to sense 1
if the value is in Q_2 , assign the occurrence to sense 2

Dictionary-Based Disambiguation

- If we have no information about the sense categorization of a word
 - Relying on the senses in dictionaries and thesauri.
- Disambiguation based on sense definitions
- Thesaurus-based disambiguation
- Disambiguation based on translations in a second-language corpus
- One sense per discourse, one sense per collocation

Dictionary-Based Disambiguation

disambiguation-based on sense definitions (1/3)

- D_1, \dots, D_k the dictionary definitions of the senses S_1, \dots, S_k of the ambiguous word w , represented as the bag of words occurring definition.
- v_j is the word occurring in the context c of w
- E_{v_j} is the dictionary definition of v_j (union of all the sense definitions of v_j)

Dictionary-Based Disambiguation

disambiguation-based on sense definitions (2/3)

- **The algorithm:**

- Given a context c for a word w

- For all senses s_1, \dots, s_k of w do

- $score(s_k) = overlap(D_k, \bigcup_{v_j \text{ in } c} E_{v_j})$

that is, overlap (word set of dictionary definition of sense S_k ,
word set of dictionary definition of V_j in context c)

- end

- Choose the sense with highest score.

Dictionary-Based Disambiguation

disambiguation-based on sense definitions (3/3)

■ **Example** (Two Senses of *ash*):

■ Senses	Definition
■ S1 tree	a tree of the olive family
■ S2 burned stuff	the solid residue left when
■	combustible material is burned

■ Score	Context
■ S1 S2	
■ 0 1	This cigar burns slowly and creates a stiff ash
■ 1 0	The ash is one of the last trees to com into leaf.

Dictionary-Based Disambiguation

Thesaurus based disambiguation (1/2)

- This exploits the semantic categorization provided by a thesaurus like Roget's.
- Semantic categories of the words in a context
 - decide the semantic category of the context
 - then decide which word sense are used
- (Walker, 1987) : Each word is assigned one or more subject codes which corresponds to its different meanings.
- For each subject code, we count the number of words (from the context) having the same subject code.
- We select the subject code corresponding to the highest count.

Dictionary-Based Disambiguation

Thesaurus based disambiguation (2/2)

- Walker's Algorithm

comment: given context c

for all senses s_k of w do

$$\text{score}(s_k) = \sum_{v_j \text{ in } c} \zeta(t(s_k), v_j)$$

end

choose s' s.t. $s' = \arg \max_{s_k} \text{score}(s_k)$

- The unit value is either 1 or 0

Dictionary-Based Disambiguation

disambiguation-based on translations in a second-language
(1/3)

- This method makes use of word correspondences in a bilingual dictionary.
- First language
 - The one for which we want to do disambiguation
- Second language
 - Target language in the bilingual dictionary
- For example, if we want to disambiguate English based on German corpus, then English is the 1st language, and the German is the 2nd language.

Dictionary-Based Disambiguation

disambiguation-based on translations in a second-language
(2/3)

- For the word “interest” :

■	Sense1	Sense2
■ Definition	legal share	attention, concern
■ Translation	Beteiligung	Interesse
■ Collocation	acquire an interest	show interest
■ Translation	Beteiligung erwerben	Interesse zeigen

Dictionary-Based Disambiguation

disambiguation-based on translations in a second-language
(3/3)

- For disambiguation (for example {interest, show})
 - Step1
 - Count the number of times that translations of the two senses of interest occur with translations of show in the second language corpus
 - Setp2
 - Compare the counts of the two different senses
 - Step3
 - Choose the sense that has the higher counts as a corresponding sense

Dictionary-Based Disambiguation

one sense per discourse, one sense per collocation
(1/2)

- Most dictionary-based algorithms process each occurrence separately.
- There are constraints between different occurrences that can be exploited for disambiguation.
- One sense per discourse
 - The sense of a target word is highly consistent within any given document.
- One sense per collocation
 - Nearby words provide strong and consistent clues to the sense of a target word. (word sense depends on context)

Dictionary-Based Disambiguation

one sense per discourse, one sense per collocation
(2/2)

- The first constraint is especially useable when
 - The material to be disambiguated is a collection of small documents
 - Or can be divided into short discourses
- For example
 - Discourse initial label context
 - D1 living the existence of *plant* and animal life
 - D1 living classified as either *plant* of animal
 - D1 ? Although bacterial and *plant* cells are...

Unsupervised Disambiguation (1/3)

- (Schutze, 1998)
 - Disambiguate word senses without having resource to supporting tools such as dictionaries and thesauri and in the absence of labeled text.
 - Simply cluster the contexts of an ambiguous word into a number of groups and discriminate between these groups without labeling them.
 - The probabilistic model is the same Bayesian model as the one used for supervised classification, but the $P(v_j | s_k)$ are estimated using the EM algorithm.

Unsupervised Disambiguation (2/3)

■ EM algorithm

- Initialize $p(v_j | s_k) \rightarrow$ random

- Compute likelihood $l(C | \mu)$, and $P(c_i) = \sum_{k=1}^K P(c_i | s_k) P(s_k)$

$$l(C | \mu) = \log \prod_{i=1}^I \sum_{k=1}^K p(c_i | s_k) p(s_k) = \sum_{i=1}^I \log \sum_{k=1}^K p(c_i | s_k) p(s_k)$$

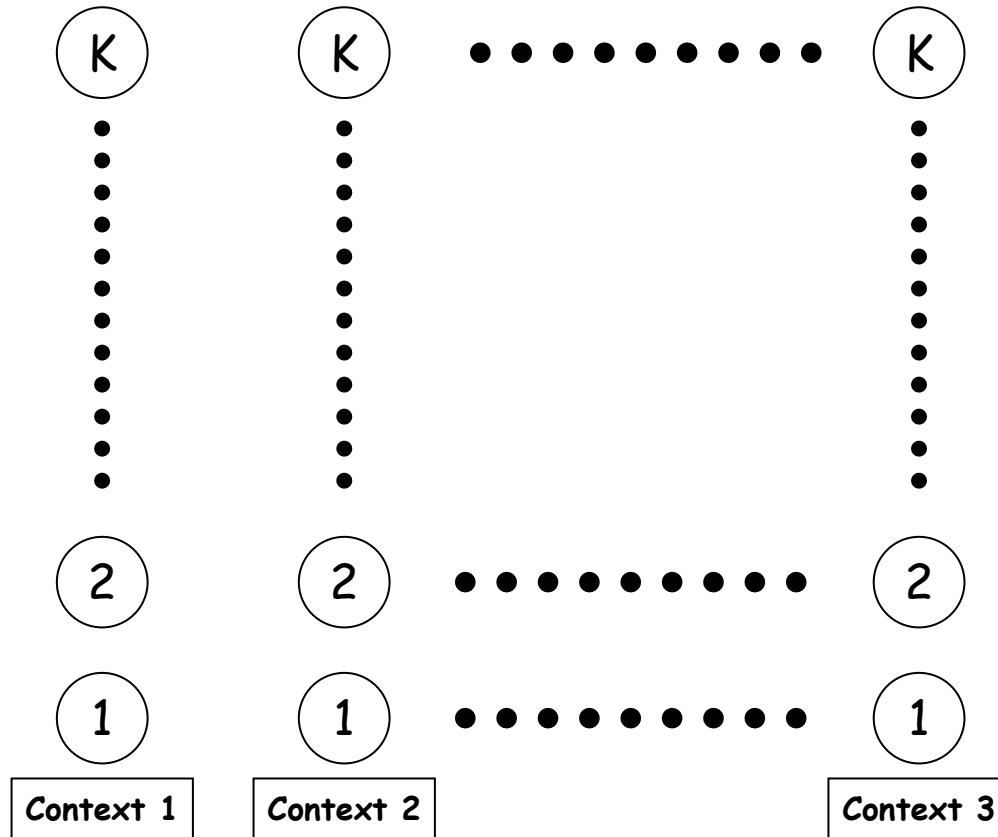
- While $l(C | \mu)$ is improving repeat: $p(c_i | s_k) = \prod_{v_j \in c_i} p(v_j | s_k)$

- E step : $h_{i,k} = \frac{p(c_i | s_k)}{\sum_{k=1}^K p(c_i | s_k)}$

- M step : Re-estimate

$$p(v_j | s_k) = \frac{\sum_{\{c_i: v_j \in c_i\}} h_{i,k}}{\sum_{k=1}^K \sum_{\{c_i: v_j \in c_i\}} h_{i,k}} \quad p(s_k) = \frac{\sum_{i=1}^I h_{i,k}}{\sum_{k=1}^K \sum_{i=1}^I h_{i,k}}$$

Unsupervised Disambiguation (3/3)



■ The End