Universidad Carlos III de Madrid

Technologies for Text (Semi-)Automatic Semantic Annotation

Prof. Luis Sánchez Fernández

Departamento de Ingeniería Telemática
Universidad Carlos III de Madrid

Contents

Introduction

Generalities

- Goal: extract semantic annotations from free text
- Natural language is complex and ambiguous
- Language dependent
- Domain dependent applications
  - News
  - Literature
  - E-mail
  - Transcriptions of spoken dialogues
- Some useful results can be achieved nowadays

Taxonomy of semantic annotations

– Content based annotations
- Document categorization
- Named entities
- Ontology based domain annotations
  – Concepts and instances identification
  – Relations extraction

NEWS: Technology

SAVE THIS | EMAIL THIS | PRINT THIS | MOST POPULAR

Wash. bans 'violent' game sales

State becomes first in nation to regulate the sale of video games.

May 21, 2003: 10:40 AM EDT
By Chris Morris, staff writer

NEW YORK (CNN/Money) – The state of Washington has become the first in the nation to regulate the sale of video games. Gov. Gary Locke on Monday signed into law a bill banning the sale of certain 'violent' games to anyone under 17.

isGovernor(GaryLocke,WST)

Named Entity (Washington, location)

<rdf:Description rdf:about='WST'> <rdf:type rdf:resource='State'/> </rdf:Description>

<rdf:Description rdf:about='WDC'> <rdf:type rdf:resource='City'/> </rdf:Description>

Meanwhile, other regulation bills are in the works. The largest of these is in Washington, D.C., where Sen. Joe Baca, D.-Calif., has reintroduced his "Protect Children from Video Game Sex and Violence Act". The bill would make it a federal crime to sell or rent "adult video games" to minors – with proposed fines of $5,000 or more. Re-introduced to the House on Feb. 11, the bill is currently in the Subcommittee on Crime, Terrorism, and Homeland Security. The 2002 bill of the same name died in that committee.

basic techniques (i)

- Symbolic NLP
  – Based on the use of lexicons and grammar rules to process text
  – Example: "Barack Obama Elected President"

Lexical Analysis

NP → Barack
NP → Obama
VBT → Elect
VBT → VBT + 'ed'
NN → President

Parsing

S → NP NP* VBT NN

S
NP NP VBT NN

Semantic Analysis

S → NP NP*(X) VBT(Elect) NN(Y)

hasFunction(X, Y)

hasFunction(BarackObama, President)

Basic techniques (ii)

- Statistical NLP
  - Based on counting: finding frequent patterns that make likely the occurrence of certain text feature
  - Use of extensive corpora
  - Example:
    - “Washington” when appearing in the same document with “Hollywood” is likely to represent (Denzel Washington, actor) while Washington” when appearing in the same document with “Obama” is likely to represent (Washington D.C., American capital)
    - We can count the frequency of different meanings of “Washington” when appearing in different <u>contexts</u>

Symbolic NLP

Typical Symbolic NLP process (i)

- Tokenisation
  - Identification of words and punctuation marks
  - Blank spaces ease this task
  - Still, some problems may apppear, for instance with hyphenation

Within the semantic annotation process, one of the key problems that we found in NEWS was the disambiguation of the entities detected by the natural language processing engine. This engine extracts named entities out of the news items, but, in order to allow a fine-grained semantic search for the user of the NEWS system, these entities have to be matched against instances of the NEWS ontology. That is, the natural language processing engine can detect that a cer-tain occurrence of the piece of text *Bush* represents a person, but we also need to deduce that this person is represented in the NEWS ontology by a certain URI like *http://www.news-project.com/2005/1*.

Typical Symbolic NLP process (ii)

- Sentence Segmentation
  - Identification of sentences in free text
  - Based on period and other punctuation marks
  - Context should be taken into account to deal with situations like abbreviations:
    - “… said the director of Russian Bear Ltd. He denied this. But …” (example taken from [1])

[1] A Mikheev, C Grover, M Moens: “Description of the LTG System used for MUC-7“. Seventh Message Understanding Conference, 1998

Typical Symbolic NLP process (iii)

- Lexical analysis
  - Goal: determination of certain features of each word in a piece of text
    - Syntactic category
    - Possible meanings
    - Other: plurals, verbal forms, …
  - Procedure
    - Rules for decomposition of a word in prefixes, root and suffixes
      - Ex. (Spanish) com-er, com-ido, com-eré
  - Procedure (cont.)
    - Other rules can help to identify syntactic categories (i.e. first word capitalized → proper noun)
    - Big lexicons
  - Problems
    - Word/root not in lexicon
      - Continuous creation of new words
    - Misspellings

Typical Symbolic NLP process (iv)

- Parsing
  - Goal: identify the syntactic structure of a sentence
  - Based on grammars
  - Very difficult
    - Natural language structure is very complex (much more than that of programming languages); in practice, it is impossible that a grammar reflects all possible correct sentence structures
    - Humans are breaking rules all the time
  - “relaxed” rules and statistical algorithms
- Semantic analysis
  - Goal: process the syntax tree to identify logic statements

Typical Symbolic NLP process (v)

I visited Paris
I bought you some expensive cologne
I went to London
I bought a coat
Then I flew home

I visited Paris
I bought a coat
I went to London
I bought you some expensive cologne
Then I flew home

- Discourse understanding
  - Goal: try to understand the meaning of a piece of text
  - The meaning of a sentence is influenced by the global meaning of the text where the sentence appear

Ambiguity issues

- Major difficulty when processing free text
- Dealt with by means of context
- Examples:
  - Part of speech
    - “can” can be either a verb or a noun
  - Lexical
    - George Bush, Washington, bank
  - Syntactic
    - I ate spaghetti with meatballs
    - I ate spaghetti with a fork
  - Referencial
    - It

Statistical NLP

Overview

- Goal: estimate the value of certain feature of a piece of text
  - Word, sentence, document
- Applications: text categorization, instance recognition, part of speech tagging, …
- Procedure
  - Define certain context to be used
  - Define a mathematical model of that context
  - Use a training set
    - For each possible value the feature can take identify which documents in the training set have such value
    - Compute the mathematical representation of the defined context for each document in the training set
  - Compute the mathematical representation of the defined context for the piece of text we want to annotate
  - Use some algorithm to compare the mathematical representations to estimate the feature value

Contexts

- Whole document (i.e. categorization) vs. local context (i.e. part of speech tagging)
- Common words vs. named entities (i.e. instance recognition)
- Which common words?
  - All, only nouns, all but stop words
  - Removing words is not a good idea if we are interested in part-of-speech chains patterns
- Previous annotations
  - For instance, use categorization annotations for instance recognition

Use of ontologies in the annotation process

- Ontologies can be used in several disambiguation tasks
  - Use instance class and instance name to find instance candidates (instance recognition)
  - Use domain and range to match candidate relations (relation extraction)
  - Use datatype properties to match the literal object value with the context (instance recognition)
  - Use object properties to match 2 named entities in the text we want to annotate (instance recognition)
  - Use object properties to match previously recognized instances in the text we want to annotate (instance recognition)
  - …

Algorithms

- Probability models
  - Try to estimate the likelihood of a piece of text in a given context having a certain feature value
  - Example: Naive Bayes
- Vector models
  - Represent the context as a vector
    - Example: TF-IDF
  - Try to compare the vector that represent the context of the piece of text to be annotated with the vectors of the training set
    - Example: cosine similarity

Naive Bayes (i)

- Tries to find the most likely feature value, taking into account the context
- A_i is one of the possible feature values
- B represents the context

$$P(A_i \mid B)$$

Naive Bayes (ii)

- Each B_j represents that the context of the piece of text to be annotated fulfills certain condition
- Examples:
  - "The named entity X appears in the context"
  - "The previous word to the one to be annotated is a definite article"

$$B = B_1 \cap \cdots \cap B_n$$

Bayes Theorem

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

$$P(A_i \mid B) = P(A_i \mid B_1 \cap \cdots \cap B_n) = \frac{P(B_1 \cap \cdots \cap B_n \mid A_i)P(A_i)}{P(B_1 \cap \cdots \cap B_n)}$$

Naive Bayes (iii)

- $P(B_1 \cap \ldots \cap B_n)$ does not depend on Ai, so we discard it
- Assuming that for all j,k, $(B_j|A_i)$ and $(B_k|A_i)$ are independent events

$$\frac{P(B_1 \cap \cdots \cap B_n \mid A_i)P(A_i)}{P(B_1 \cap \cdots \cap B_n)} = $$

$$= \frac{P(A_i)\prod_{l=1}^{n} P(B_l \mid A_i)}{P(B_1 \cap \cdots \cap B_n)}$$

TF-IDF

- Define the context (global or local)
- Select some terms that will be used to represent the context content
  - For instance, x nouns most frequent in the training set
- Each term will correspond to an element of the vector
  - Its value will be the TF-IDF value of that term

TF

- tf: term frequency
  - The most frequent a term is in a document the most related is that document with such term
  - $\mathtt{freq}_{i,j}$= number of ocurrences of term i in document j

$$tf_{i,j} = \frac{freq_{i,j}}{\max_l freq_{l,j}}$$

IDF

- idf: inverse document frequency
  - The less frequent is a term the more information that term provides
  - ni= number of documents in corpus where term i occurs
  - N= total number of documents in corpus

$$idf_i = \log \frac{\mathrm{N}}{\mathrm{n_i}}$$

Cosine similarity

- The cosine of the angle of 2 vectors can be used to compare how similar are the vectors excluding the vector modules

$$sim\left(\vec{v}_1, \vec{v}_2\right) = \frac{\sum_{i=1}^{n} v_{1i} v_{2i}}{\sqrt{\sum_{i=1}^{n} v_{1i}^2}\sqrt{\sum_{i=1}^{n} v_{2i}^2}}$$

Typical treatment for different types of semantic annotations

Text Categorization (i)

- We start with a document *training set*
  - The class of each document is manually annotated
  - For each document a mathematical representation of its content is produced
    - For instance, we can select the terms more frequent in the training set, excluding *stop words*
    - Then each document is represented by a vector whose components are the number of occurrences of certain term

Term vector: (president, campaign, healthcare, republican, biracial, economy)

TIME

No, when historians analyze the 2008 campaign, they're going to remember that the two-term Republican President had 20% approval ratings, that the economy was in meltdown, and that Americans didn't want another Republican President. They'll also remember that Obama was a change candidate in a change election. And of course they'll remember that America elected a biracial leader less than a half-century after Jim Crow. But that's just about all they'll remember. Politics is a lot simpler than the pundits pretend.

Document content representation: (2, 1, 0, 2, 1, 1)

Text Categorization (ii)

- To categorize a document first obtain its mathematical content representation
- Then compare it using some classification algorithm with the vectors of the documents in the training set and select the appropriate category
- Please note that incertain applications/domains a document can belong to several categories

Text Categorization (iii)

- For instance, you can use cosine similarity and select the class of the most similar document in the training set
- Alternative:
  - Find the k documents in the training set most similar to document to be categorized
  - Decide by majority

Named entity recognition

- Based on the combination of
  - Rules that recognize certain entity types
    - For instance, the rule “in the LOC area” can recognize in the sentence “in the Washington area” that Washington is an entity of type location
  - Lexicons with names of cities, contries, organizations, persons, etc.
- It is unusual that the same entity occurs in a document with different meanings

Instance recognition

- Setp 1: Named entity recognition
- Step 2: Selection of instance candidates: which instances in the knowledge base match with the name and type of the entity
- Step 3: Use the context in which the named entity occurs to identify the instance
  - Typical contexts:
    - N words before and after the named entity
      - Sentence is a limit?
      - Which words (i.e. type-of-speech restrictions)?
    - Other named entities
    - Document categories
  - Use of a training set

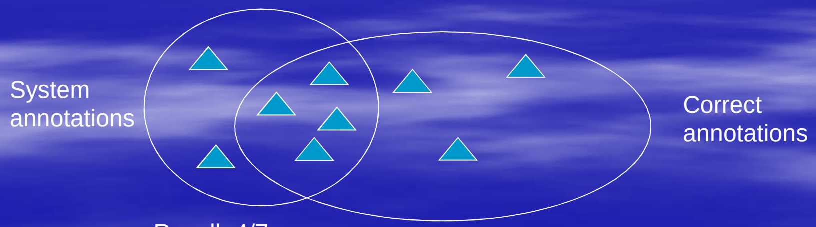
Relation extraction

- Step 1 and 2: parsing and instance recognition
- Step 3 (semantic analysis): process the syntax tree to identify relations (defined by properties in the ontology)
  - Ontological context (for instance domain and range of properties) is very relevant here
- The most difficult semiautomatic annotation task
  - Language structure complexity
  - Ambiguity
  - Discourse context
  - Irony

Quality Measures

$$\text{Recall} = \frac{\text{Correct system annotations}}{\text{Total correct annotations}}$$

$$\text{Precission} = \frac{\text{Correct system annotations}}{\text{Total system annotations}}$$



Recall=4/7
Precission=4/6

Quality Measures

- In many semiautomatic annotation tasks it is difficult to produce good results both in precission and recall
- F-measure allows to compare different systems or system configurations combining both values
  - α=0.5 is typical

$$F = \frac{1}{\alpha \frac{1}{\text{Precission}} + (1-\alpha)\frac{1}{\text{Recall}}}$$

$$F_{\alpha=0.5} = \frac{2 \text{ x Precission x Recall}}{\text{Precission} + \text{Recall}}$$

Quality Measures

- Some semiautomatic annotation tasks can achieve good quality results if the system is tuned for a particular language and domain
  - Text categorization: F-measure above 95%
  - Named entity recognition: F-measure above 90%
  - Instance recognition: F-measure above 80%

References

- Stuart Russell and Peter Norvig. Artificial Intelligence. A modern approach. Prentice-Hall, 1995.
- Christopher D. Manning and Hinrich Schütze. Foundations of Statistical Natural Language Processing. The MIT Press, 1999.
- GATE, A General Architecture for Text Engineering. http://gate.ac.uk/