



Tracce di approfondimento per il capitolo 4 Lo studio della lingua e la linguistica computazionale

Disambiguazione di parole ambigue utilizzando WordNet (difficoltà: *)

La ricerca su web si basa in larga misura sull'assunzione che i bisogni dell'utente sono esprimibili in linguaggio naturale (principalmente attraverso "parole chiave") così come il contenuto dei documenti da trovare. Se cerchiamo parole semanticamente ambigue (es. cuore, pianta, cane, mano...) recupereremo documenti che includono tutti i possibili significati della parola. Un recente trend dei motori di ricerca (ad esempio Bing della Microsoft) è quello di cercare di categorizzare i risultati di query "ambigue". Ad esempio interrogando un futuribile motore di ricerca con la parola "fegato" potremmo aspettarci documenti che trattano di epatiti e documenti che parlano di imprese coraggiose suddivisi in due aree ben distinte.

Immaginiamo di voler creare il nostro filtro per categorizzare in modo automatico i risultati standard di una query su un normale motore di ricerca. Questo è possibile ad esempio arricchendo le nostre ricerche con termini estratti da Wordnet.

Ecco come procedere:

1. cercare alcune parole ambigue (es. cuore, aereo, seguire...) su ItalWordNet (http://www.ilc.cnr.it/iwndb_php);
2. una volta identificata almeno una coppia di sensi chiaramente distinguibili, provare a fare una ricerca con il termine ambiguo (es. "cuore") su Google e annotarsi i primi 10-20 documenti trovati, classificandoli rispetto al diverso senso individuato;
3. usare alcuni iperonimi (uno per volta) in aggiunta al termine ambiguo e rilanciare la ricerca su Google (es. "cuore organo" vs "cuore luogo") e annotare i nuovi risultati;
4. usare alcuni iponimi (sempre uno per volta) in aggiunta al termine ambiguo e rilanciare la ricerca su Google (es. "cuore muscolo" vs "cuore centro") e annotare i nuovi risultati;
5. usare meronimi, olonimi, termini coordinati... e registrare i risultati della ricerca in termini di sensi individuati;

Con i risultati ottenuti usando un numero sufficiente di termini ambigui (almeno 10-20) provare a rispondere alle seguenti domande:

1. cosa specifica meglio la categoria semantica del termine ambiguo (il suo iperonimo, un suo iponimo, un meronimo, un sinonimo... più cose combinate insieme...)?
2. quanto è affidabile WordNet per disambiguare i risultati di una query ambigua (funziona meglio con i nomi, con i verbi o con gli aggettivi)?
3. quanto si modificano copertura e precisione (vedi [paragrafo 3.3](#))?

Parlare con il calcolatore (difficoltà: **)

Come abbiamo visto nel [paragrafo 3.2](#), progettare un sistema in grado di dialogare con gli esseri umani è piuttosto semplice. Proviamo quindi ad implementare un chatbot vero e proprio utilizzando un semplice linguaggio di etichettatura: l'AIML (Artificial Intelligence Markup Language) <http://www.alicebot.org/aiml.html>.

Scegliere quindi un tema da affrontare (ad esempio un corso di cucina, un tema di filosofia morale, un servizio su cui fornire supporto tecnico...), studiare un prototipo di dialogo su questo tema

Marco Lazzari, Alessandra Bianchi, Mauro Cadei,
Cristiano Chesi, Sonia Maffei

Informatica umanistica



(magari raccogliendo dati sfruttando il dialogo con degli amici) ed implementare alcune semplici regole seguendo lo standard AIML. Per vedere il chatbot in funzione e per testarlo, utilizzare uno dei molti programmi freeware disponibili all'indirizzo <http://www.alicebot.org/downloads/programs.html>.

Riflettere su pregi e limiti di questo approccio al dialogo con la macchina.

Creazione e interrogazione di un piccolo corpus non annotato (difficoltà: ***)

Come discusso nel paragrafo 2.3, i corpora sono strumenti molto utili per descrivere una lingua. Si invita il lettore a raccogliere ed interrogare un piccolo corpus (10.000-100.000 parole) seguendo questi passi:

1. scelta della tipologia del corpus (es. produzioni spontanee di bambini che parlano la loro lingua madre, articoli di giornale, conversazioni in dialetto...)
2. identificazione della popolazione da analizzare e criteri di selezione (es. bambini dai 24 ai 36 mesi, articoli di giornale di cronaca del 2010, parlanti dialetto nativi di 50-70 anni...)
3. definizione della struttura del corpus (es. file di testo, codifica UTF-8, un file per conversazione, un file per articolo... nel (nome del) file sono incluse le informazioni di provenienza del testo, la tipologia ecc.)
4. campionamento del materiale e trascrizione (es. registrazione conversazioni di 30 minuti circa, raccolta di articoli di giornale di 1000 battute...)

Ricordiamoci che un corpus deve rappresentare un campione realistico della lingua che si vuole studiare: deve essere sufficientemente esteso, deve essere raccolto in un contesto "ecologico" (ad esempio, se si studia la lingua parlata in un determinato contesto, bisogna registrare i parlanti in normali conversazioni e trascrivere esattamente quello che dicono), si deve adottare uno "standard" di codifica delle informazioni che si intendono registrare (ad esempio errori di pronuncia se si ritiene che questi siano importanti, annotazioni del contesto...).

Una volta costruito il corpus si cerchi di esplorarlo con programmi per identificare le concordanze (ad esempio TextStat, <http://neon.niederlandistik.fu-berlin.de/textstat/>) e/o per l'interrogazione del testo attraverso l'uso di espressioni regolari (ad esempio WinGrep, <http://www.wingrep.com/>)

Per un approfondimento si vedano alcuni corpora da prendere come esempio e i seguenti testi di approfondimento:

- CHILDES (<http://childes.psy.cmu.edu/>);
- CoLFIS (http://www.istc.cnr.it/material/database/colfis/index_eng.shtml)
- Lenci, Montemagni e Pirrelli (2005) Testo e computer. Elementi di linguistica computazionale.

Creazione ed interrogazione di un piccolo corpus annotato (difficoltà: ****)

Raccogliere un piccolo corpus come descritto nella traccia precedente.

Una volta creati i vari file di testo, scegliere uno standard di annotazione morfosintattica (ad esempio il Siena University Treebank, SUT, <http://www.ciscl.unisi.it/progetti/sut/>, il Turin University Treebank, TUT, <http://www.di.unito.it/~tutreeb/>, o il Penn Treebank, <http://www.cis.upenn.edu/~treebank/>) e, usando un software di annotazione (ad esempio XMLTreeTagger per lo standard SUT: <http://www.ciscl.unisi.it/progetti/sut/>) taggare il corpus prodotto.

Esplorare infine il corpus utilizzando software di interrogazione adeguati (ad esempio TGrep se si sceglie uno standard di annotazione Penn-compatible; notare che esistono filtri di esportazione che

Marco Lazzari, Alessandra Bianchi, Mauro Cadei,
Cristiano Chesi, Sonia Maffei

Informatica umanistica



convertono sia file in SUT che in TUT verso lo standard Penn).

Per un approfondimento si vedano:

- Chesi, Lebani, Pallottino (2008) A Bilingual Treebank (ITA-LIS) suitable for Machine Translation: what Cartography and Minimalism teach us. *StIL Vol. 2*
- Lesmo, Lombardo, Bosco (2002) Treebank Development: the TUT Approach. In *Proceedings of ICON 2002, Mumbai, India, 2002*
- Marcus, Santorini, Marcinkiewicz (1994) Building a large annotated corpus of English: The Penn Treebank. *ACL Proceedings*.