ELABORAZIONE DEL LINGUAGGIO NATURALE (ELN) 2005/2006

NAMED ENTITY RECOGNITION

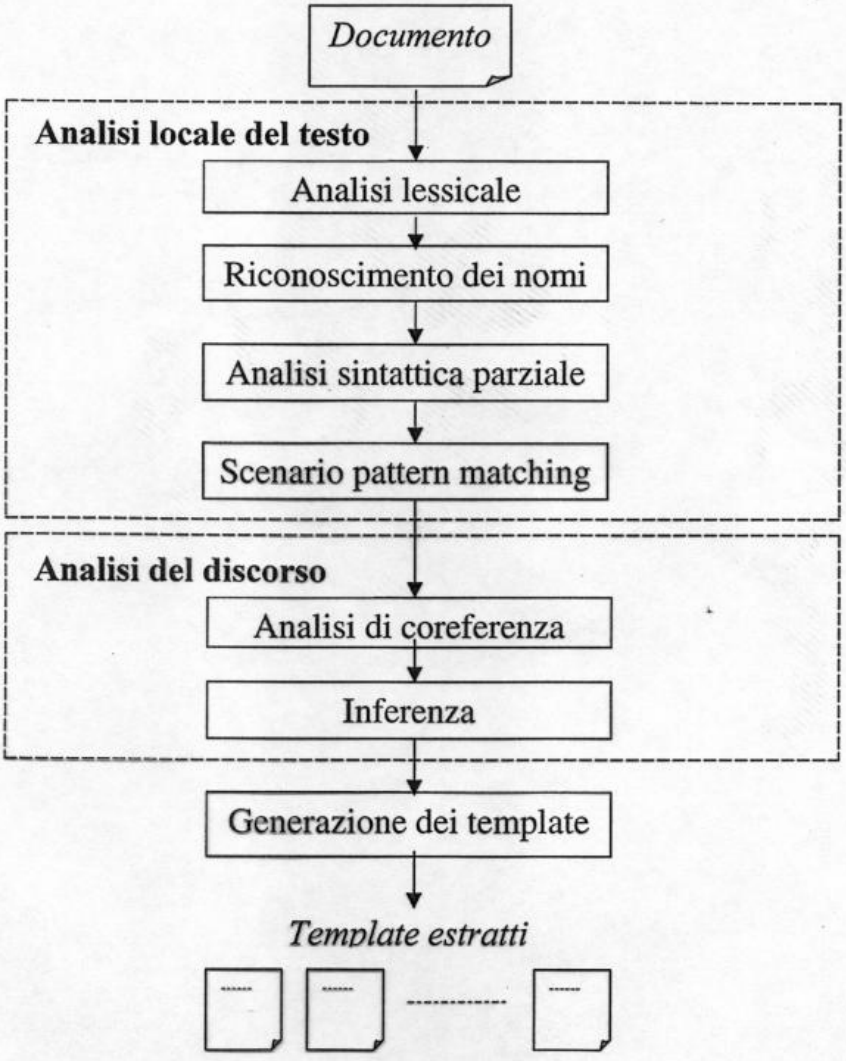
Studente: Tranchina Francesco

Matricola: 242935

Docente: Prof. Amedeo Cappelli

Prima di introdurre la definizione effettiva di Named Entity Recognition dovremmo sapere cos'è l'Information Extraction; questi due concetti sono estremamente collegati tra loro, difatti, come vedremo, il Named Entity Recognition (userò l'abbreviazione NER d'ora in poi) è una fase dell'Information Extraction (di cui userò l'abbreviazione IE). Nel cercare materiale utile per svolgere questa relazione, ho notato che molte persone confondono il termine IE con Information Retrival (IR d'ora in poi). Facciamo subito una distinzione tra questi due termini. Per IE intendiamo l'arte di trovare informazioni utili da un insieme di testi e codificarle in un formato tale da permettere la loro immissione in un database; dato, dunque, un insieme di documenti, un sistema di IE estrae unicamente l'informazione di cui l'utente ha bisogno (l'utente accede, quindi, prima all'informazione ed in seguito al documento in cui essa è contenuta). L'obiettivo principale è dunque l'estrazione d'informazione che consentono di descrivere il contenuto dei documenti. Per IR s'intende, invece, l'arte di selezionare un sottoinsieme rilevante di documenti da un insieme più grande, in risposta a delle query fornite dall'utente. L'IR è spesso utilizzata nei motori di ricerca, sistemi nei quali, inserendo una parola chiave (l'informazione a cui l'utente è interessato), vengono trovati direttamente tutti i documenti in cui tale parola è contenuta. Con il termine informazione intendiamo (definizione tratta dal dizionario interattivo Garzanti) un elemento che ci consente di avere conoscenza di fatti e situazioni. L'IE la possiamo vedere suddivisa in diverse fasi:

STRUTTURA DI UN SISTEMA DI I.E.



Analizziamo velocemente le varie fasi (non è scopo di questa relazione trattare approfonditamente tutte le fasi dell'IE):

Analisi lessicale: Il testo viene prima diviso in frasi e token. Ciascun token viene ricercato all'interno di un dizionario per determinarne i possibili part-of-speech (con questo termine intendiamo la Categoria morfo-sintattiche cui una parola appartiene; le categorie principali sono i nomi, i verbi, aggettivi, avverbi, pronomi, preposizioni…si procede poi ad una fase chiamata POS tagging che procede ad assegnare ad una parola tutti i possibili POS tag, utilizzando un dizionario ed ad applicare le regole di disambiguazione create) ed altre caratteristiche. Generalmente tali dizionari includono una raccolta di nomi di società, abbreviazioni, suffissi di compagnie ed altro.

Riconoscimento dei nomi: La fase successiva del processo identifica i vari tipi di nomi propri ed altre forme speciali, come dati e cifre. E' importate individuare i nomi propri poiché questi ultimi appaiono frequentemente in molti tipi di testi e la loro identificazione e classificazione semplifica le successive fasi di elaborazione. I nomi vengono identificati tramite un set di pattern (espressioni regolari) espresse nei termini del part-of-speech, delle caratteristiche sintattiche e delle caratteristiche ortografiche (ad esempio l'iniziale maiuscola). I nomi propri, per esempio, potrebbero essere identificati da un titolo che precede un nome, *Mr. Herrington David*, da un suffisso, *Snippety Smith Jr.*, o da una iniziale puntata all'interno di una sequenza di nomi, *Humble T. Hopp*. I nomi delle

società possono essere identificate dai loro token finali, ad esempio *Hepplewhite Inc*, *Hepplewhite Corporation*, ecc…, oppure dai loro alias: HP è un alias della Hewlett-Packard Corp. Comunque, esistono dizionari contenenti i nomi delle maggiori società; in tal modo si semplifica notevolmente la fase di riconoscimento dei nomi. L'esempio che segue mostra il risultato dell'applicazione della fase di riconoscimento dei nomi al testo già visto precedentemente.

> [name type: person *Sam Schwarts*] *retired as executive vice president of the famous hot dog manufacturer,* [name type: company*Hupplewhite Inc.*] *He will succeeded by* [name type: person*Harry Himmerlfarb*]

Analisi sintattica: Questa fase identifica i legami sintattici fra i diversi elementi di una frase. Un'analisi sintattica profonda di una frase ha generalmente come risultato una foresta di alberi di derivazione sintattica, ciascuno dei quali fornisce una possibile interpretazione sintatticamente corretta della frase stessa. L'approccio seguito nei sistemi di IE si basa sull'identificazione di legami sintattici elementari fra i diversi elementi della frase. Si rinuncia alla determinazione dell'albero sintattico completo di interpretazione della frase a favore di una interpretazione locale dei sintagmi (cioè, parole o insiemi di parole che abbiano un significato logico) di una porzione della frase stessa. In linea di principio tale analisi può essere assimilata ad un approccio bottom-up nella identificazione di un albero di interpretazione sintattica, interrotto prima di completare l'analisi della frase. Questo approccio al parsing è detto robusto per il fatto che è in grado di fornire una interpretazione sintattica parziale della frase fornita in ingresso anche nel

caso in cui essa sia grammaticalmente scorretta. Questa caratteristica deriva dalla possibilità di indagare localmente la struttura della frase senza che la richiesta di raccordo fra le interpretazioni sintattiche locali faccia fallire il metodo. L'identificazione di alcuni aspetti della struttura sintattica semplifica la successiva fase di elaborazione. Infatti, gli argomenti da estrarre spesso corrispondono a frasi di nomi (NP – noun phrase) nel testo, mentre le relazioni di solito corrispondono a relazioni grammaticali. Purtroppo, l'individuazione di strutture sintattiche complete si rivela piuttosto difficile. Alcuni sistemi di IE non hanno una fase separata di analisi sintattica. Altri sistemi tentano di costruire un parsing completo della frase.

Scenario pattern matching: l'obiettivo è l'estrazione di eventi o relazioni rilevanti per lo scenario di interesse.Si fa uso, anche in questa fase, di pattern. Nel caso dello scenario relativo all'evento di successione nel management, vengono usati i pattern:

<person> *retires as* <position>

<person> *is succeded by* <person>

dove <person> e <position> sono elementi che "matchano" con frasi nominali (NP) del tipo da essi indicato. Il risultato di tale fase è un testo marcato con eventi,

[clause event: e7 *Sam Schwartsretired as executive vice president of the famous hot dog manufacturer, Hupplewhite Inc.*] [clause event: e8 *He will succeeded by Harry Himmerlfarb*]

che risulteranno collegati alle entità individuate nella fase precedente.

Analisi di coerenza: Ha come obiettivo la risoluzione dei riferimenti dei pronomi.

Inferenza: Può accadere che informazioni relative ad uno stesso evento siano sparse in diverse frasi. E' necessario, allora, riunire tali informazioni prima della generazione dei template. In altri casi si verifica che siano presenti delle informazioni non esplicitamente indicate nel testo. Per renderle esplicita si fa uso del meccanismo dell'inferenza. Nel dominio degli eventi di successione, nel caso in cui si desideri produrre un template contente informazioni circa chi perde o lascia una particolare posizione, bisognerà determinare cosa implica il predicato "succeed",. Ad esempio, dalla frase "*Sam was president. He was suceeded by Harry*" possiamo inferire che Harry diverrà presidente. Inferenze di questo tipo possono essere implementate tramite i sistemi a produzione. E' chiaro che regole di produzione di questo tipo saranno progettate in modo tale da considerare il lemma del verbo (ottenuto nella fase di part-of-speech tagging) e non tutti i possibili modi di coniugarlo. Si riduce, così, il numero di regole di produzione.

Generazione dei Template: Costruzione dei template di output: strutture a frame con slot da riempire con i valori estratti. Tutte le informazioni finora ricavate dal testo sono sufficienti per l'estrazione dei template. Questi sono frame con slot da riempire con le informazioni richieste. I template estratti, relativi all'evento di *management succession* citato nel testo di esempio sono i seguenti:

EVENT: *leave job*

PERSON: *Sam Schwarts*

POSITION: *excutive vice president*

COMPANY: *Hupplewhite Inc.*

EVENT: *start job*

PERSON: *Harry Himmelfarb*

POSITION: *excutive vice president*

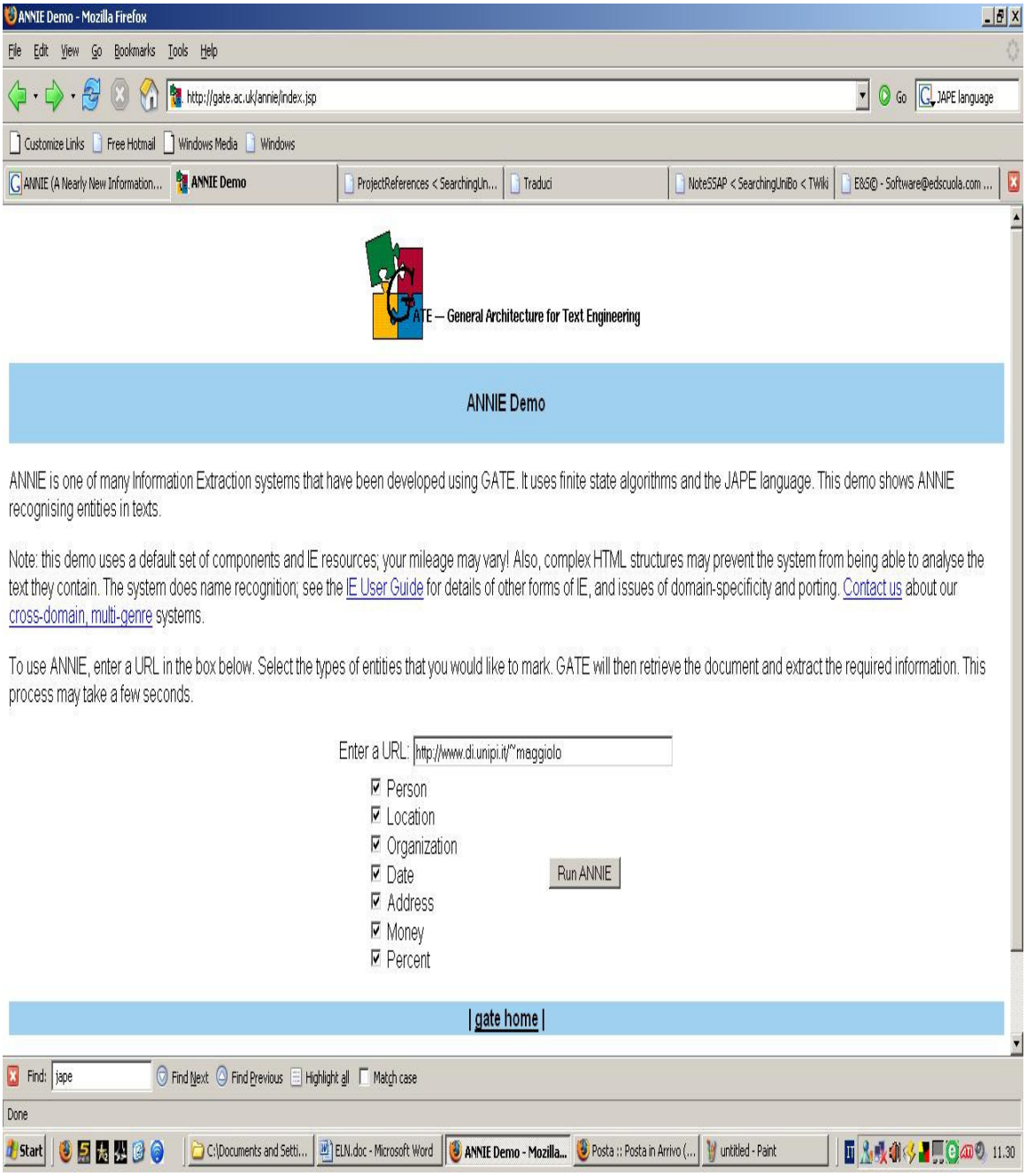
COMPANY: *Hupplewhite Inc.*

E' evidente, quindi, che da una stessa porzione di testo possono essere estratti più template in base al numero di eventi di interesse citati nello stesso.

Soffermiamoci, tralasciando le altre fasi, proprio sul riconoscimento dei nomi, il **Named Entity Recognition**. Tale termine è definito come la fase dell'IE in cui si cerca di classificare gli elementi atomici di un testo in categorie predefinite, quali i nomi delle persone o delle organizzazioni, posizioni, espressioni dei periodi, quantità, valori monetari, percentuali, etc…

Attualmente uno tra i sistemi open-source di rilievo per l'IE è **GATE** (a General Architecture for Text Engineering), sviluppato presso l'Università di Sheffield (UK). GATE è in grado di fornire un'infrastruttura modulare per lo sviluppo di tool per il Language Engineering e per l'analisi dei testi. Di particolare interesse per questa relazione è un modulo all'interno di GATE, chiamato ANNIE (A Nearly New Information Extraction System), il quale mette a disposizione una serie di tecniche per il NER. ANNIE utilizza una serie di language re-souces e di processing resources per riconoscere istanze di entità (Persone, Luoghi, Date, Indirizzi ...) all'interno di un testo. ANNIE utilizza JAPE (Java Annotation Pattern Engine; è un engine basato su

espressioni regolari e regole da applicare alle espressioni; il risultato di JAPE è la corrispondenza delle entità astratte con quelle concrete, ad esempio, Marco = person; in effetti, JAPE è il cuore di ANNIE). Nella figura sottostante vediamo la pagina principale (http://gate.ac.uk/annie/index.jsp) mentre nella successiva è rappresentato il risultato della ricerca delle varie istanze all'interno della pagina http://www.di.unipi.it/~maggiolo:




<B style="color:black;background-color:#FFA0FF"><Person gate:gateId="983" rule="PersonUnknown">Andrea Maggiolo-Schettini(</Person>)</B>'s home page - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://gate.ac.uk/annie/annie.jsp?url=http%3A%2F%2Fwww.di.unipi.it%2F%7Emaggiolo&annotation%5B%5D=Person&annotation%5B%5D=Location&annotation%5B%5
Go
JAPE language
Customize Links Free Hotmail Windows Media Windows
ANNIE (A Nearly New Information...
<B style="color:black;backgr...
ProjectReferences < SearchingUn...
Traduci
NoteSSAP < SearchingUniBo < TWiki
E&S© - Software@edscuola.com ...
ATE – General Architecture for Text Engineering
ANNIE Output for http://www.di.unipi.it/~maggiolo
Annotation Key:
Person Location Organization Date Address Money Percent
Andrea Maggiolo Schettini()
Full Professor at Dipartimento di Informatica , Universita` di Pisa()
In Supraeme Dignitatis
Contact information
Andrea Maggiolo Schettini
() Dipartimento di Informatica
Universita` di Pisa
() Largo Pontecorvo 3
56127 Pisa() - Italy
Room(): 279 D-E
Tel: ++39 050 2212() 759
Fax: ++39 050 2212() 726
Find: jape
Find Next
Find Previous
Highlight all
Match case
Done
Start
C:\Documents and Setti...
ELN.doc - Microsoft Word
<B style="color:blac...
Posta :: Posta in Arrivo (...
untitled - Paint
11.29

Avendo introdotto il termine Language Engineering provvediamo a definirlo; tale termine, però, è spesso collegato con Computational Linguistic e Natural Language Processing. Il Computational Linguistic "è un ramo della linguistica in cui le tecniche computazionali ed i concetti vengono applicati per la spiegazione di problemi linguistici e fonetici" [Crystal, D. 1991. A Dictionary of Linguistics and phonetics. 3rd edtn. Blackwell Publishers, Oxford.], mentre Natural Language Processing "è un ramo dell'informatica che studia sistemi automatici per l'elaborazione del linguaggio naturale. Include lo sviluppo di algoritmi per il parsing, la generazione, e l'acquisizione di conoscenza linguistica, l'indagine sulla complessità spaziale e temporale di tali algoritmi, la progettazione di linguaggi formali computazionalmente utili (come grammatiche e formalismi lessicali) per codificare conoscenza linguistica, l'indagine su architetture software appropriate per i vari compiti del Natural Language Processing e considerazioni sui tipi di conoscenza non linguistica che vengono a contatto con il Natural Language Processing. Il Natural Language Processing è un'area di studio discretamente astratta che non mette particolare impegno nello studio della mente umana, e neppure nel produrre artefatti utili." [Gadzar 1996]. In definitiva, Computational Linguistic è una parte della scienza del linguaggio che usa i computer come strumenti di indagine; Natural Language Processing è quella parte dell'informatica che si occupa dei sistemi computerizzati per l'elaborazione del linguaggio. Il Language Engineering è l'applicazione del Natural Language Processing alla costruzione di sistemi computerizzati che elaborano il linguaggio per qualche compito come la modellazione del linguaggio stesso, o "l'uso strumentale dell'elaborazione del linguaggio, tipicamente come parte di un sistema più

grande con qualche obiettivo pratico, per esempio l'accesso ad un database" [Thompson 1985].

Anche **Falcon**, sistema per Question-Answering Systems (è un sistema di recupero automatico delle informazioni, destinato a rispondere alle domande che gli sono poste nel linguaggio naturale), sfrutta metodi per Named Entity Recognition per la fase di *Expected answer type*; è il passaggio in Falcon in cui viene individuato il modello di risposta attesa. Il sistema è dotato di un riconoscitore di nomi che copre 18 categorie, chiamate Top Categories, ognuna delle quali è collegata a numerose classi di parole contenute in WordNet, la più nota ontologia esistente (l'ontologia è il tentativo di formulare uno schema concettuale esaustivo e rigoroso nell'ambito di un dato dominio; si tratta generalmente di una struttura dati gerarchica che contiene tutte le entità rilevanti, le relazioni esistenti fra di esse, le regole, gli assiomi, ed i vincoli specifici del dominio. L'uso del termine "ontologia" nell'informatica è derivato dal precedente uso dello stesso termine in filosofia, dove ha il significato dello studio dell'essere o dell'esistere, così come le fondamentali categorie e delle relazioni tra esse). Wordnet tenta di descrivere i concetti (sinonimi, contrari e relazioni tra concetti) contenuti nella lingua inglese. Questo sistema è stato sviluppato dal linguista George Miller presso l'Università di Princeton.

WordNet e' una base di conoscenza lessicale per l'inglese, disponibile gratuitamente, su supporto elettronico. In origine il progetto si e' ispirato alle correnti teorie psicolinguistiche sulla memoria lessicale umana. Nomi, verbi, aggettivi ed avverbi sono organizzati in insiemi di sinonimi, ciascuno dei quali rappresenta un concetto lessicale. Questi insiemi di sinonimi sono collegati tra di loro tramite un certo numero di relazioni

ed organizzati in tassonomie. Nella attuale versione di WordNet sono presenti 95.600 forme lessicali organizzate in 70.100 significati (o synsets). Le corrispondenze tra forme lessicali e significati vengono mantenute tramite una matrice bidimensionale, nella quale ciascun synset è inteso essere un designatore non ambiguo del significato di una parola. Spesso (circa il 70%) ad un synset viene associata anche una breve definizione (gloss). WordNet distingue due tipi di relazioni: relazioni lessicali, quali la sinonimia, la antinomia e la polisemia, e relazioni concettuali, quali l'iponimia e la meronimia.

La relazione lessicale più importante per WordNet è la similarità di significato, dal momento che la capacità di riconoscere sinonimia tra parole è un prerequisito per la costruzione dei synsets e quindi per la rappresentazione dei significati nella matrice lessicale. Due espressioni sono sinonime se vale il principio di sostitutività (in altre parole se la sostituzione di una con l'altra non cambia il valore di verità di una frase). In realtà risulta più utile una definizione più debole, relativizzata ad un contesto. Due espressioni sono sinonime in un contesto linguistico C se la sostituzione di una con l'altra in C non cambia il valore di verità. È importante notare che la definizione di sinonimia in termini di sostitutività rende necessario partizionare WordNet in nomi, verbi, aggettivi e avverbi. Ovviamente l'appartenenza di una parola a più di un synset dà un'indicazione della sua polisemia. La relazione di antinomia fornisce invece il principio organizzativo centrale per aggettivi ed avverbi. WordNet è dunque una rete semantica di concetti connessi gli uni agli altri e contenuti in una lista organizzata sottoforma di grafi di relazioni.

Anche nei sistemi di elaborazione del Linguaggio Naturale (in sigla, NLP, Natural Language Processing) è prevista una fase di NER, prendiamo come esempio **LingPipe**, che è un software JAVA per NLP. LingPipe riconosce sia istanze di persone, organizzazioni, ma anche istanze di nomi del campo biomedico (per esempio, genomi, enzimi, geni, etc…). Nel tutorial del programma, vediamo come LingPipe effettua la fase di riconoscimento di una frase del campo biomedico (la frase è: p53 regola l'insulina dell'essere umano come espressione del gene di fattore II di sviluppo attraverso il promotore attivo P4 in cellule di rhabdomyosarcoma. La frase in inglese, utilizzata nel tutorial è: p53 regulates human insulin-like growth factor II gene expression through active P4 promoter in rhabdomyosarcoma cells.); per far ciò lingPipe sfrutta il programma Confidence Named Entity Chunking (questo programma dirà con quale percentuale un'entità è un genoma). Vediamo l'output:

Phrase: p53 regulates human insulin-like growth factor II gene expression through active P4 promoter in rhabdomyosarcoma cells

Rank	Conf	Span	Type	Phrase
0	**0.9999**	(0, 3)	GENE	p53
1	**0.7328**	(81, 92)	GENE	P4 promoter
2	**0.6055**	(20, 54)	GENE	insulin-like growth factor II gene
3	**0.3817**	(14, 54)	GENE	human insulin-like growth factor II gene
4	**0.1395**	(74, 92)	GENE	active P4 promoter
5	**0.0916**	(81, 83)	GENE	P4
6	**0.0088**	(74, 83)	GENE	active P4
7	**0.0070**	(20, 49)	GENE	insulin-like growth factor II
8	**0.0044**	(14, 49)	GENE	human insulin-like growth factor II

Ulteriori approfondimenti nella home page del sistema: http://www.alias-i.com/lingpipe/index.html

Come abbiamo visto, il Named Entity Recognition è una fase fondamentale dell'Information Extraction, utilissima inoltre anche per sistemi di Question - Answering e di Natural Language Processing.

Bibliografia & Sitografia:

1. http://www.alias-i.com/lingpipe/demos/tutorial/ne/read-me.html il sito di lingpipe (questo link ci conduce al tutorial per la fase di Named Entity)
2. Tesi di De Paulis Paolo, 2004/2005 (www.ing.unisi.it/~biblio/tesi.htm)
3. Tesi di Davide Dattilo (http://digilander.libero.it/davidemichele/)
4. Tesi di Roberta Benassi, (www.dbgroup.unimo.it/tesi/benassi.pdf)
5. Tesi di Deborah Vinciguerra, 2003/2004 (medeaserver.isa.cnr.it/dacierno/tesipdf/**vinciguerra**.pdf)
6. Tesi di Francesco Bertagna, 2005/2006, (etd.adm.unipi.it/theses/available/etd-05102006-085731/unrestricted/Bertagna_tesi_final.pdf)
7. Tesi di Ernesto di Iorio, 2004/2005, (www.ing.unisi.it/~biblio/tesi.htm)
8. Articolo: Named-Entity Recognition in Novel Domains with External Lexical Knowledge di Massimiliano Ciaramita e Yasemin Altun (www.loa-cnr.it/Papers/ciaramita_altun_structlearn.pdf)
9. Articolo: Named Entity Recognition with Character-Level Models di Dan Klein, Joseph Smarr, Huy Nguyen e Christopher D. Manning (www.cs.berkeley.edu/~klein/papers/character-ner.pdf)
10. Articolo: Using WordNet Predicates for Multilingual Named Entity Recognition di Matteo Negri e Bernardo Magnini (http://www.fi.muni.cz/gwc2004/proc/102.pdf)
11. Articolo: Named Entity Recognition from Diverse Text Types di Diana Maynard, Valentin Tablan, Cristian Ursu, Hamish Cunningham e Yorick Wilks (http://gate.ac.uk/sale/ranlp2001/maynard-etal.pdf)

12. http://www.gate.ac.uk/ sito di GATE
13. Lucidi tratti dal professor Giovanni Semeraro, docente del corso di Gestione della conoscenza dell'azienda presso la facoltà di Bari , dipartimento di informatica (www.di.uniba.it/~semeraro/GCI/ProgrammaPreliminare_GCI_2005_2006.pdf)
14. Lucidi del corso di Database della facoltà di Ingegneria di Reggio-Emilia e Modena (http://www.dbgroup.unimo.it/tesi/ner.html)
15. Articolo: Costruzione di una base di conoscenza lessicale per l'italiano basata su WordNet di Bernardo Magnini e Carlo Strapparava (multiwordnet.itc.it/pa**per**/wordnet-sli.pdf)
16. Thompson, H. 1985. Natural language processing: a critical analysis of the structure of the field, with some implications for parsing. In K. Sparck-Jones and Y. Wilks, editors, *Automatic Natural Language Parsing*. Ellis Horwood.