Trattamento del Linguaggio Naturale Tramite Prolog: un Approccio Promettente per Generare Istituzioni Virtuali da Testi Scritti

Michele Bozzano, Angela Locoro, Maurizio Martelli, Viviana Mascardi

DISI, Dipartimento di Informatica e Scienze dell'Informazione,
Via Dodecaneso 35, 16146, Genova, Italia
michele.bozzano@gmail.com,{angela.locoro, maurizio.martelli, viviana.mascardi}@unige.it

Abstract. Le Istituzioni Virtuali sono un formalismo estremamente potente per descrivere relazioni sociali tra agenti autonomi, ma sono di difficile uso per non-esperti di tali tecnologie. In questo articolo descriviamo un estrattore semi-automatico di ruoli e relazioni tra concetti (e quindi, indirettamente, tra ruoli) a partire da testi scritti. Tale strumento potrà essere utilizzato dagli esperti del dominio per creare Istituzioni Virtuali, senza essere esposti alla complessità del linguaggio con il quale l'Istituzione Virtuale viene descritta.

Keywords: Istituzioni Virtuali, Disambiguazione del Senso di una Parola, Estrazione di Ruoli e Relazioni tra Ruoli.

1 Introduzione

I *Mondi Virtuali Tridimensionali* progettati a scopo non ludico stanno acquisendo sempre più importanza grazie ai numerosi domini in cui trovano applicazione, che vanno dal commercio elettronico al turismo elettronico, alla salvaguardia dei beni culturali. In questi contesti, la possibilità di simulare interazioni tra agenti e di verificarne la conformità rispetto a regole imposte dal mondo stesso riveste un ruolo cruciale poiché consente, ad esempio, di addestrare correttamente un discente sulle possibili azioni ed interazioni ammesse in determinate situazioni, oppure di validare ipotesi sul tipo di relazioni sociali esistenti in società scomparse, oppure ancora di vivere un'esperienza di turismo virtuale estremamente realistica in cui il visitatore umano interagisce con agenti software nell'ambiente e secondo le tradizioni del luogo riprodotto.
Per definire le regole comportamentali di una società di agenti in un ambiente condiviso risultano particolarmente utili le *Istituzioni Elettroniche*, "un modo di implementare convenzioni di interazione tra agenti - umani o software - che possono stabilire obbligazioni in un ambiente aperto" (http://e-institutions.iiia.csic.es/).
Le *Istituzioni Virtuali* [Bog07] rappresentano l'intersezione tra Mondi Virtuali Tridimensionali ed Istituzioni Elettroniche.
La potenzialità delle Istituzioni Virtuali e la trasversalità delle loro applicazioni sono ampiamente riconosciute dalla comunità scientifica, tuttavia, a causa della

complessità del linguaggio per specificare ruoli, norme, protocolli, scene ed altre componenti fondamentali di un'Istituzione Virtuale, la sua progettazione ed implementazione è appannaggio esclusivo di ricercatori con competenze informatiche specifiche su linguaggi dichiarativi ed agenti. In [BPA+09] abbiamo illustrato la progettazione e parziale realizzazione di uno strumento mirato a rendere disponibile contenuto culturale alla grande massa in forma di gioco, ma con basi scientifiche rigorose fondate su Istituzioni Virtuali. Tale strumento dovrà essere utilizzato *in piena autonomia dagli esperti del dominio*, e questo sarà possibile creando un'interfaccia che *mascheri all'esperto del dominio la complessità del linguaggio con il quale l'Istituzione Virtuale viene descritta*. L'Istituzione Virtuale è caratterizzata in primo luogo dai *ruoli* che gli agenti possono ricoprire in essa e dalle *relazioni tra tali ruoli*, che determinano cosa un agente che ricopre un ruolo possa, non possa, debba o non debba fare/dire in determinate situazioni. Uno degli aspetti cruciali per supportare l'esperto è quindi la realizzazione di un estrattore semi-automatico di ruoli e relazioni tra concetti (e quindi, indirettamente, tra ruoli) a partire da testi scritti.
Ad esempio, la generazione automatica della relazione *handle(priest, corpse)* dal testo *"[...] but the priests of the Nile themselves handle the corpse [...]"*, tratto dal secondo libro delle Storie di Erodoto e la individuazione del ruolo *role(clergyman, priest)*, potrebbero fornire un supporto all'esperto di dominio che le visiona e decide se esse sono corrette e se vanno impiegate per generare una specifica nel formato proprio delle Istituzioni Virtuali. L'utilità di disporre di un estrattore di questo tipo è stata evidenziata in più occasioni dall'archeologo che ha partecipato alla stesura di [BPA+09].

In questo articolo descriviamo l'implementazione in SWI Prolog, esteso con la libreria ProNTo_Morph [Sch03] e con l'accesso alla versione Prolog di WordNet (http://wordnet.princeton.edu/wordnet/download/), di un estrattore semi-automatico di ruoli di concetti e relazioni tra concetti a partire da testi scritti, finalizzato alla generazione semi-automatica di Istituzioni Virtuali. La semi-automaticità è legata alla possibilità di generare relazioni e ruoli non corretti: l'output dell'estrattore deve sempre essere supervisionato da un esperto. L'estrattore utilizza tecniche di comprensione del linguaggio naturale, in particolare viene utilizzata la disambiguazione del senso delle parole ("Word Sense Disambiguation", WSD [AE06]). Pur essendo il problema affrontato intrinsecamente complesso, gli esperimenti dimostrano che, su alcune tipologie di testo, l'estrattore dà buoni risultati: la relazione tra ruoli e l'assegnazione di un ruolo ad un concetto descritte precedentemente in questa sezione sono state ottenute dall'esecuzione dell'estrattore su una traduzione in inglese del testo di Erodoto.

L'articolo è organizzato nel modo seguente: la Sezione 2 pone le basi scientifiche per comprendere il lavoro proposto; la Sezione 3 descrive l'algoritmo implementato; la Sezione 4 tratta e commenta gli esperimenti condotti; la Sezione 5 conlcude illustrando i lavori collegati e le attività future.

2 Basi scientifiche del lavoro proposto

In questa sezione illustriamo le basi, necessarie a comprendere il resto dell'articolo, su cui il nostro lavoro si poggia: istituzioni elettroniche e disambiguazione del significato di una parola sfruttando WordNet.

2.1 Istituzioni elettroniche

Secondo [Nor90] le interazioni tra esseri umani sono guidate da istituzioni che forniscono la struttura della vita quotidiana, definiscono le regole del gioco in una società ed introducono vincoli formali e informali necessari affinché l'interazione abbia luogo in modo controllato ed efficace. Le istituzioni sono la struttura all'interno della quale avviene l'interazione e definiscono cosa i vari individui possono, devono, non possono fare sotto determinate circostanze. Le istituzioni devono essere create (si pensi alla stesura della Costituzione di uno stato) e possono evolvere (si pensi alla legge ordinaria). Un tratto caratterizzante delle istituzioni è la chiara distinzione tra le regole e gli individui che sottostanno a tali regole. Grazie alle istituzioni è possibile formare organizzazioni ovvero "unità sociali (o gruppi di esseri umani) costituite e ricostituite deliberatamente per perseguire obiettivi specifici" [Etz64].

Il modo in cui le organizzazioni sono create è influenzato dalla struttura istituzionale che a sua volta influenza il modo in cui le organizzazioni evolvono. Le organizzazioni devono conformarsi alle regole della istituzione per essere accettate, legittimate e supportate. In [ERS+01] la metafora delle organizzazioni istituzionalizzate è usata per progettare ed implementare organizzazioni aperte di agenti software, denominate istituzioni elettroniche, in cui un gran numero di agenti umani e software giocano ruoli differenti ed interagiscono tramite atti comunicativi [Sea69].

Gli elementi che caratterizzano una istituzione elettronica sono quindi

- *Agenti e ruoli*: gli agenti sono i giocatori all'interno di un'istituzione elettronica, mentre i ruoli sono schemi di comportamento standardizzati. La identificazione e la regolamentazione dei ruoli è una fase rilevante del processo di formalizzazione di ogni organizzazione. Ogni agente in una istituzione elettronica deve adottare uno o più ruoli.
- *Struttura dialogica*: alcuni aspetti di una istituzione quali gli oggetti del mondo ed il linguaggio utilizzato per la comunicazione sono fissati e costituiscono il contesto o struttura dell'interazione tra agenti.
- *Scena*: le interazioni tra agenti avvengono mediante incontri chiamati scene e seguono un protocollo di comunicazione ben definito.
- *Struttura performativa*: le scene possono essere collegate e formare quindi una rete di scene detta struttura performativa.
- *Regole normative:* le azioni degli agenti nel contesto di una istituzione possono avere conseguenze che limitano oppure allargano le successive possibilità di azione. Queste conseguenze imporranno obblighi agli agenti e avranno un impatto sui loro possibili cammini all'interno della struttura performativa.

Per la creazione e verifica di istituzioni elettroniche è possibile usare l'editor ISLANDER (http://e-institutor.iiia.csic.es/islander/islander.html) che consente la specifica delle componenti di un'istituzione elettronica mediante un linguaggio testuale dichiarativo.

Le istituzioni elettroniche 3D, chiamate "Virtual Institutions" [Bog07] sono state utilizzate con successo in diversi domini, dall'apprendimento basato su imitazione alla riproduzione di società antiche e alla condivisione di beni culturali.

2.2 Disambiguazione del significato di una parola

Il problema dell'estrazione di ruoli e relazioni fra ruoli può essere inquadrato nel contesto più generale dell'estrazione dell'informazione o *Information Extraction* (IE) [CL96,Sar08]. In particolare, nel campo dell'elaborazione di linguaggio naturale, un aspetto pregnante del problema riguarda la disambiguazione o *word sense disambiguation* (WSD) [AE06]; essa coinvolge l'associazione ad una parola in un testo del significato che meglio le si addice rispetto a tutti i possibili significati (*sense*, in inglese). Questa associazione può essere ottenuta eseguendo due passi: (i) si recuperano tutti i significati delle parole più rilevanti del testo preso in considerazione; (ii) si assegna ad ogni parola il significato più appropriato.
Per quanto riguarda il recupero dei significati, esistono proposte basate sull'uso di significati predefiniti simili a quelli presenti nei dizionari, su gruppi di associazioni tra parole come i sinonimi, o ancora su accessi a traduzioni in altre lingue. Per ogni significato non esiste tuttavia una definizione accettata da tutti, che cambia per esempio da dizionario a dizionario in base al grado di granularità scelto dall'autore.
Per individuare il giusto significato di una parola, si deve tener conto del contesto in cui la parola è inserita ed è utile sfruttare dati esterni quali risorse lessicali ed enciclopediche.
Le metodologie sono molte, ma raggruppabili in tre aree principali:

- Metodi basati sull'Intelligenza Artificiale
- Metodi basati su corpora
- Metodi basati sulla conoscenza

I primi fanno riferimento ai tradizionali metodi di rappresentazione della conoscenza come, per esempio, le reti semantiche; i secondi ottengono empiricamente i risultati utilizzando gli esempi forniti dai corpora. I *metodi basati sulla conoscenza*, sui quali ci soffermiamo in quanto sono quelli usati nel nostro approccio, si basano sull'estrazione automatica di informazioni da basi di dati quali dizionari, thesauri e corpora. Essi ebbero nuova linfa con la diffusione su larga scala di queste basi di conoscenza. In questa categoria rientrano lavori compiuti sui "Machine-readable dictionaries", contenenti informazioni lessicali, come gli algoritmi proposti da Lesk [Les86] e da Wilks [Wil90]; lavori compiuti sui thesauri, che contengono relazioni tra categorie di parole e informazioni semantiche, il più famoso dei quali è il Roget's International Thesaurus; lavori compiuti su dizionari semantici, i quali possono essere di tipo enumerativo, in cui i significati sono forniti esplicitamente come WordNet, oppure di tipo generativo, in cui i significati associati alle parole sono derivati da regole di generazione.
WordNet [Mil95] è una base di dati non legata ad un particolare dominio di applicazione, che raggruppa tutte le parole della lingua inglese. Include sia concetti di tipo generale, sia concetti con un maggior grado di specializzazione, collegati non solo da relazioni lessicali ma anche da relazioni semantiche, ed è organizzata in quattro categorie sintattiche: nomi, verbi, aggettivi, avverbi.

All'interno di ogni categoria le parole sono raggruppate in insiemi di sinonimi, detti "synset", per ciascuno dei quali è fornita una definizione, detta "gloss". Ogni parola gestita da WordNet appartiene ad almeno un synset. Le parole che appartengono a più synset sono dette polisemiche e hanno la caratteristica di avere più di un significato ad esse attribuibile.

Le *relazioni semantiche* legano coppie di synset ed includono: *iperonimia*, in cui il secondo synset denota una classe di oggetti più ristretta di quella rappresentata dal primo synset; *iponimia*, relazione inversa dell'iperonimia; *istanza*, in cui il primo synset è un'istanza del secondo synset; *implicazione*, in cui il primo synset rappresenta un'azione che non può verificarsi in mancanza dell'azione rappresentata dal secondo synset; *similarità*, in cui il secondo synset, detto satellite, è legato al primo, detto cluster head, da una somiglianza di significato; *meronimia*, in cui il primo synset è parte del secondo synset oppure ne è membro oppure è una sostanza che si può trovare al suo interno; *causa*, in cui il primo synset rappresenta un'azione scatenante l'azione che rappresenta il secondo synset; *attributo*, in cui un synset, rappresentante un aggettivo, è legato al synset rappresentante il nome di cui esprime il valore.

Le *relazioni lessicali* legano coppie di vocaboli ed includono: *sinonimia*, che esiste tra parole appartenenti a uno stesso synset; *antinomia*, che lega una parola al suo contrario; *classe*, in cui il primo vocabolo può essere classificato come appartenente alla classe indicata dal secondo vocabolo; *derivazione*, che specifica l'esistenza di una derivazione morfologica tra la prima e la seconda parola, tale relazione è riflessiva.

3 Estrazione di ruoli e relazioni tra ruoli da testo scritto

L'algoritmo che proponiamo analizza il testo ed estrae ruoli e relazioni tra ruoli sfruttando WordNet e l'algoritmo di Lesk adattato [BP02], partendo da un testo scritto che descrive in linguaggio naturale la situazione che vogliamo modellare come Istituzione Virtuale.

L'implementazione è stata sviluppata in SWI Prolog esteso con la libreria ProNTo_Morph [Sch03] e con l'accesso alla versione Prolog di WordNet. Prolog è utilizzato con successo per applicazioni di trattamento del linguaggio naturale da circa vent'anni [Cov93,Lag00]; il suo utilizzo si è rivelato particolarmente adatto anche al nostro scopo. Come mostrato in questa sezione mediante alcuni frammenti del codice sviluppato, infatti, il codice prodotto è estremamente compatto e leggibile. I meta-predicati per la raccolta di insiemi di soluzioni si sono rivelati particolarmente utili per l'individuazione e la selezione di tutti gli iperonimi di un concetto, passaggio necessario per assegnare un ruolo ad un concetto, e per l'individuazione delle coppie e terne di parole legate da qualche tipo di relazione all'interno della finestra scorrevole.

La tecnica che abbiamo sviluppato per estrarre ruoli e relazioni tra ruoli si basa sull'analisi della punteggiatura del testo in input, effettuata considerando il testo avulso da qualsiasi contesto, inteso come dominio o corpus di riferimento. Il testo viene suddiviso in segmenti separati l'uno dall'altro da un segno di punteggiatura forte (punto, punto e virgola, due punti, punto di domanda, punto esclamativo). Ognuno di questi segmenti viene analizzato attraverso una “finestra scorrevole”, spostata di una parola alla volta, la quale contiene al più K parole significative. Nei nostri esperimenti abbiamo posto K = 5 riscontrando empiricamente che con tale valore si ottengono i risultati migliori. Se il segmento di testo contiene meno di K parole significative allora la finestra avrà dimensioni ridotte e coinciderà con tutto il segmento.

Data questa definizione di finestra, è ragionevole assumere che gli aggettivi riferiti ai sostantivi appartenenti alla finestra caschino dentro la finestra stessa, così come le

azioni riferite a questi sostantivi. La vicinanza spaziale risulta altresì importante. Un aggettivo si riferisce ad un sostantivo tanto più gli è vicino nella finestra. Analogamente per i verbi.

Per disambiguare le parole contenute all'interno della finestra abbiamo adottato l'algoritmo di Lesk adattato per tener conto della punteggiatura. Al meglio della nostra conoscenza, sfruttare la punteggiatura per circoscrivere la ricerca di ruoli in relazione è un aspetto originale della nostra proposta. L'idea è quindi quella di far coincidere il contesto con la finestra e applicare a questa l'algoritmo di Lesk.

Una volta terminata la fase di disambiguazione, ad ogni parola viene assegnato un ruolo, ovvero il concetto più generale da cui discende, utilizzando le relazioni semantiche di WordNet. In questa fase di estrazione dei ruoli è importante non perdere la contestualizzazione di ogni termine.

L'estrazione delle relazioni avviene tenendo conto sia delle informazioni prodotte in fase di contestualizzazione che in fase di disambiguazione. Le relazioni vengono estratte associando gli aggettivi ai nomi e gli avverbi ai verbi. Queste associazioni sono guidate dalla distanza tra le parole all'interno della finestra. Successivamente vengono legati verbi e nomi sempre tenendo conto che parole più vicine hanno maggiore probabilità di essere in relazione rispetto a parole più distanti.

Il cuore del nostro algoritmo è implementato dal predicato `doWork/4` che, presa una finestra scorrevole di K parole significative, le disambigua considerando come contesto la finestra stessa e associando ad ogni parola il proprio senso (predicato `disambiguate/3`, il cui primo argomento è la lista di parole da disambiguare, il secondo è il contesto - che coincide con le parole stesse - ed il terzo è la lista di sensi ottenuti dal processo di disambiguazione usando l'algoritmo di Lesk), trova i ruoli ricoperti dalle parole dati i loro significati (predicato `findRoles/2`, il cui primo argomento è la lista di parole con relativo senso disambiguato, ed il secondo è la lista di ruoli associati alle parole), e infine trova le relazioni tra le parole considerando che parte del discorso, in inglese “Part of Speech” abbraviato in POS, ricoprono (predicato `findRelationsByPOS/2`).

```
doWork(W,Senses,Roles,Relations):-
    disambiguate(W,W,Senses),
    findRoles(Senses,Roles),
    findRelationsByPOS(Senses,Relations).
```

Nel seguito illustriamo nel dettaglio le 8 fasi che caratterizzano il nostro algoritmo.

FASE 1: Pre-processing

Scopo di questa fase è individuare, nell’intero testo, le parole che non risultino significative per le fasi successive. Tale obiettivo è raggiunto mediante:

- Creazione di una lista statica di parole comuni (quali aggettivi o proposizioni, in inglese *stopwords*), costruita sulla base del British National Corpus (BNC, http://www.natcorp.ox.ac.uk/), a cui è associata la frequenza.
- Estrazione della lista di *stopwords* dal testo preso in esame e calcolo della loro frequenza nel testo stesso.

- Confronto di quest'ultima lista con la lista statica, basato sulla soglia $FSqrt < M * FreqSqrt$ dove $FSqrt = \sqrt{F}$, con F frequenza della parola Word nel testo preso in esame, $FreqSqrt = \sqrt{Freq}$ e $Freq$ frequenza della parola Word nella lingua, M fattore moltiplicativo fissato empiricamente a 10.

Dagli esperimenti condotti è risultato che l'utilizzo della radice quadrata e l'introduzione del fattore M sono necessari per meglio approssimare i valori di frequenza. L'analisi di un testo composto da poche frasi porterebbe senz'altro ad avere, per ogni parola, una frequenza molto superiore alle parole della lista statica. La scelta di introdurre radice quadrata e fattore M non è convalidata in letteratura, ma è stata indotta dai test svolti.

FASE 2: Selezione del segmento di testo

La fase si segmentazione del testo è basata sulla punteggiatura forte. Come riportato nella Sezione 5, questo approccio potrà essere completato da una procedura di controllo che distingua tra segni di punteggiatura che terminano il periodo rispetto a quelli presenti all'interno di parole abbreviate.

Le frasi tra parentesi sono considerate come segmenti a se stanti, assumendone l'indipendenza rispetto a quelle esterne ad esse. Se durante lo scorrimento del testo si incontra una parentesi aperta, si va alla ricerca della corrispondente parentesi chiusa e si salva la frase tra parentesi per una successiva iterazione dell'algoritmo.

Viene inoltre trattato il caso in cui ci siano parentesi annidate.

FASE 3: Individuazione delle espressioni polirematiche

Lo scopo di questa fase è quello di individuare le unità lessicali del linguaggio, ovvero le espressioni polirematiche, composte da più termini che identificano un unico concetto. Se ad essere prese in esame fossero le singole parole dell'espressione una alla volta, le informazioni ricavate sarebbero incomplete e fuorvianti rispetto al significato inteso dall'utilizzo dell'espressione stessa. Per esempio il concetto 'credit card' ha molto più senso se considerato come un tutt'uno piuttosto che suddiviso nelle singole parole 'credit' e 'card'.

Per poter quindi assegnare la corretta interpretazione semantica l'operazione viene svolta andando a confrontare sequenze di parole successive con i vocaboli in WordNet. Se le corrispondenze possibili sono più di una, viene scelta la sequenza di parole più lunga, perché sperabilmente più significativa. Ad esempio le espressioni 'professional tennis' e 'professional tennis player' sono entrambe contenute in WordNet. Se il testo contenesse l'espressione *"[...] professional tennis player [...]"* questa verrebbe unificata con il concetto formato da tutte e tre le parole, perché portatore di maggiori informazioni rispetto al concetto formato dalle sole prime due.

Nell'implementazione dell'algoritmo si ricercano sequenze con non più di 4 parole.

FASE 4: Rimozione delle parole non significative

Questa fase è strettamente correlata a quella di pre-processing. E' infatti questo lo stadio in cui le parole non significative individuate in quella fase vengono rimosse.

Ciò avviene in un secondo momento rispetto al calcolo delle frequenze e alla fase di individuazione delle espressioni polirematiche, per permettere anche l'unificazione

di polirematiche quali 'to it' o 'a few' che altrimenti, rimuomendo sia 'to' che 'it' dalla prima e 'a' dalla seconda, non verrebbero individuate.

In questa fase vengono rimosse solo le singole parole che non risultano far parte di un'espressione polirematica e tutte le parole che non sono presenti in WordNet e che non sarebbero quindi gestibili in alcun modo.

L'assenza in WordNet delle coniugazioni dei verbi e delle parole plurali rende necessario ricondurre queste parole alla loro forma normale per poterle recuperare correttamente.

Lo strumento utilizzato per compiere questo lavoro è ProNTo_Morph, un morphological analysis tool che consente di spezzare una parola in radice e suffisso.

Per esempio la parola 'played' è composta dai due morfemi 'play' e '-ed'. ProNTo_Morph si avvale di regole di spelling generali quale ad esempio il suffisso '-s' per il riconoscimento di parole plurali e coniugazioni di verbi nella terza persona singolare presente. Fa inoltre utilizzo di una lista di parole irregolari per il riconoscimento di nomi, verbi, aggettivi e avverbi che non possono essere analizzati secondo le regole grammaticali generali.

FASE 5: Creazione della finestra scorrevole

Sui concetti rimanenti del segmento di testo viene implementata una finestra scorrevole. Essa viene inizialmente posta sui primi K concetti del segmento di testo e successivamente viene fatta scorrere di un concetto alla volta fino alla fine del segmento. Durante ogni spostamento l'algoritmo può vedere attraverso la finestra solamente i K concetti contenuti in quel momento, e solo su quelli può lavorare.

Nel caso di un segmento composto da meno di K concetti, la finestra viene ridotta alle dimensioni del numero di concetti presenti.

FASE 6: Disambiguazione

La disambiguazione avviene considerando la finestra come il contesto sul quale applicare l'algoritmo di Lesk adattato.

L'algoritmo è basato sull'assunzione che le parole contenute in un certo intorno condividano un argomento comune e consiste sostanzialmente nei seguenti passi:

1. scegliere coppie di parole ambigue entro un ristretto raggio
2. controllare le loro definizioni in un vocabolario
3. scegliere i significati che massimizzano il numero di termini comuni nelle definizioni delle due parole.

Le parole da disambiguare sono tutte quelle del contesto appartenenti a più di un synset. Per ogni significato della parola da disambiguare:

- si recuperano le parole della definizione del synset (glossa) a cui essa appartiene
- si concatena la parola con la lista delle parole contenute nelle glosse dei synset in relazione diretta di iponimia, iperonimia, meronimia, olonimia e similarità con essa. Chiamiamo la lista risultante L.
- si associa, ad ogni parola del contesto, la lista delle parole contenute nella glossa di ogni synset a cui appartiene un significato della parola, concatenata con la lista delle parole contenute nelle definizioni dei synset in relazione diretta di iponimia, iperonimia, meronimia, olonimia e similarità con in vari synset cui la parola è abbinata. Tutte queste liste vengono concatenate in una unica che chiamiamo LL.

La bontà di un significato è poi calcolata a partire da questi valori:

1. il numero di occorrenze delle parole di L che compaiono in LL;
2. il numero di occorrenze delle parole del contesto che compaiono in L;
3. il numero di occorrenze delle parole di L che compaiono nel contesto.

Ognuno di questi tre valori ha associato un coefficiente modificabile che permette di variarne il peso nella valutazione finale. Di default la misura dei coefficienti è posta a 1 per il primo valore, 50 per il secondo, 25 per il terzo. Questo è dovuto al fatto che il secondo valore ha maggiore peso nella valutazione rispetto agli altri due.

La valutazione finale del significato viene calcolata come somma dei tre prodotti e tra i significati disponibili viene scelto quello con la valutazione più alta. Il risultato della disambiguazione è l'etichettatura di ogni parola disambiguata con tre informazioni: il synset, la parte del discorso, il valore di bontà del significato scelto, che sarà quello che ha ottenuto il punteggio maggiore come valutazione.

FASE 7: Estrazione dei ruoli

Una volta assegnato il presunto significato alla parola l'obbiettivo diventa estrarre il suo ruolo, che in questo contesto definiamo come “il super-concetto che abbia il miglior rapporto tra generalità e dettaglio”. Il problema sta nell'individuare quale, tra i super-concetti, presenta il giusto grado di astrazione per candidarsi a diventare un “ruolo”.

La catena dei super-concetti di un concetto C si trova tramite le relazioni semantiche di WordNet, in particolare attraverso le relazioni di istanza e di iperonimia. Per quel che riguarda i nomi, WordNet è organizzato in un albero con radice il synset a cui appartiene la parola 'entity'. Ogni altro synset è figlio diretto o indiretto di questa radice.

Risalire tutta la catena di iperonimi del synset non è quindi una soluzione percorribile, perché troppo generale e univoca; risalire la catena di un solo nodo, viceversa, potrebbe risultare troppo specifico perché troppo ‘vicino’ al synset iniziale.

L'idea è quindi fermarsi a un nodo intermedio della catena di discendenza, che sia sufficientemente informativo, generale e ad un buon livello di astrazione.

La soluzione individuata consiste quindi nel prendere la lista di tutte le parole appartenenti ai synset iperonimi diretti e indiretti di quello di partenza (chiamiamola H) e confrontarli con ognuna delle parole della glossa associata al synset di partenza (chiamiamola P) abbinando a ciascun confronto un valore di similarità, ottenuto tramite la metrica Jaro-Winkler [Jar89,Win99], così da poter confrontare anche eventuali varianti di H contenute in P, non solo parole identiche.

Il ruolo selezionato è l'iperonimo della coppia con il valore maggiore. Il calcolo del ruolo è significativo solo per nomi e verbi, in quanto aggettivi e avverbi non hanno relazioni di iperonimia.

Nel caso in cui il termine non abbia iperonimi, si esamina la possibilità che esso sia istanza di una qualche classe. Se la verifica ha esito positivo viene recuperata tutta la lista degli iperonimi diretti e indiretti della classe di cui il termine è istanza e l'operazione di paragone viene fatta su questa lista. Per esempio il synset ('Asimov', 'Isaac Asimov') non ha iperonimi, ma è legato da una relazione di classe/istanza al synset ('writer', 'author'). In caso negativo l'algoritmo stabilisce che il termine non possiede un ruolo.

Il codice Prolog che implementa questa attività è riportato nel seguito con qualche semplificazione fatta a solo scopo di leggibilità. Il predicato `findRoles/2` analizza

la lista di significati un elemento alla volta e su ciascun elemento di interesse, ovvero sostantivi e verbi, chiama `findRoleByGloss/2`.

```
findRoles([],[]):-!.

findRoles([sense(Syn,Word,POS,_)|Rest],[(Word,Role,POS,Syn)|Roles]):-
    (POS == noun ; POS == verb), !, findRoleByGloss(Syn,Role),
    findRoles(Rest,Roles).

findRoles([sense(Syn,Word,POS,_)|Rest],[(Word,noRole,POS,Syn)|Roles]):-
    findRoles(Rest, Roles).

findRoleByGloss(S,Role):-
    s(S,_,Word,_,_,_), !, /* (predicato offerto dalla versione Prolog di
WordNet) ricerca in WordNet la parola Word appartenente al synset
identificato da S */
    g(S,G),   /* (predicato offerto dalla versione Prolog di WordNet)
unifica G con la glossa associata a S */
    atom_codes(G,Chars),   /* (predicato di sistema) trasforma la
stringa G in lista di char */
    tokenize(Chars,Tokens),   /* (predicato implementato da Fabrizio
Larosa e offerto dalla libreria http://www.disi.unige.it/person/
MascardiV/BooleAndLesk/Fabrizio_Larosa.zip disponibile sotto licenza
GPLv2) opera la "tokenizzazione" ovvero la individuazione delle parole
nella glossa */
    subClassOf(S,SynList),  /* trova la lista dei synset degli
iperonimi/classi di S */
    findall(W,(member(H,SynList), s(H, ,W, , , )),L),   /* dalla lista
di identificatori di synset, ricava la lista di parole a cui tali
identificatori corrispondono */
    computeDistance(Tokens,L,Distances),   /* calcola distanze tra tutti
gli elementi di Tokens e quelli di L */
    bestRole(Distances,(Word,0),Role).   /* Role è il ruolo che dà
l'approssimazione migliore */
```

FASE 8: Creazione delle relazioni

Per creare le relazioni l'idea è quella di basarsi sulla vicinanza tra le parole e sulla parte del discorso con cui sono state etichettate nella fase di disambiguazione.

L'aggettivo è la parte del discorso che qualifica un nome, che ne evidenzia una proprietà, una caratteristica o che semplicemente dà delle informazioni su di esso. Quando compare in una frase si riferisce sempre a un nome.

L'avverbio, similmente, descrive il modo con cui un'azione viene fatta; figura sostanzialmente da commento al verbo a cui si riferisce all'interno della frase.

Verbi e nomi sono la sostanza della frase. Legati tra loro danno senso alla frase, descrivendo una situazione, un avvenimento o quant'altro.

Le associazioni tra le varie parti del discorso avvengono in questo ordine:

1. gli aggettivi si legano ai nomi;
2. gli avverbi si legano ai verbi;
3. si associano i verbi e i nomi.

Le prime due associazioni avvengono con le stesse modalità. Si prende ogni aggettivo e lo si lega al nome più vicino a lui nella finestra. Se esistono due nomi alla stessa distanza si creano entrambe le associazioni. La stessa cosa avviene prendendo gli avverbi e legandoli ai verbi più vicini a loro.

Per la terza associazione si seguono degli schemi predefiniti che corrispondono alle sequenze di verbi e nomi che maggiormente ricorrono nelle comuni frasi in lingua inglese.

Gli schemi con cui si confrontano le parole sono i seguenti:

- nome/verbo/nome
- nome/verbo
- nome/nome

Il fatto che nella finestra compaiano aggettivi e avverbi tra i verbi e i nomi influenza solamente quest'ultimo schema di associazioni (nome/nome). Si è deciso quindi di mettere in relazione due nomi solamente se risultino essere molto vicini all'interno della finestra, cioè se non distano l'uno dall'altro più della metà della lunghezza della finestra. Nei primi due schemi risulta invece irrilevante la presenza di aggettivi o avverbi, in quanto non sono presi in esame nel calcolo.

Le associazioni così create vengono scritte su un file di testo.

Il predicato che implementa questa attività è `findRelationsByPOS/2` che, presa una lista di sensi come quella passata a `findRoles`, unifica `Relations` con le relazioni significative tra le parole appartenenti alla lista. Mostriamo per esteso anche il predicato `createAssociations/2` richiamato da `findRelationsByPOS`.

```
findRelationsByPOS(SenseList,Relations):-
      addNumToList(SenseList,ListWithNumbers),  /* assegna un numero
progressivo a ogni elemento e scarta informazioni non necessarie */
    bindAdjectives(ListWithNumbers,CouplesAN),  /* CouplesAN è la lista
delle coppie (Aggettivo,Nome di riferimento) */
    bindAdverbs(ListWithNumbers,CouplesAV),    /* CouplesAV è la lista
delle coppie (Avverbio,Verbo di riferimento) */
   createAssociations(ListWithNumbers,Associations),   /* Associations è
la lista dei gruppi di elementi in relazione tra loro */
   append(CouplesAN,CouplesAV,L1), append(L1,Associations,Relations).
/* concatena i risultati */


createAssociations(ListWithNumbers,Assoc):-
findall((Noun1,Verb,Noun2),(
          member((Noun1,noun,N1),ListWithNumbers),
          member((Verb,verb,N2),ListWithNumbers),
          member((Noun2,noun,N3),ListWithNumbers),
          N1<N2, N2<N3
      ),L1), /* trova le terne del tipo (Nome,Verbo,Nome) */
findall((Noun3,Verb2),(
          member((Noun3,noun,I1),ListWithNumbers),
          member((Verb2,verb,I2),ListWithNumbers),
          I1<I2,
          not(member((_,Verb2,_),L1))
       ),L2), /* trova le coppie del tipo (Nome,Verbo), con Verbo che
ancora non sia stato considerato */
findall((Noun4,Noun5),(
          member((Noun4,noun,M1),ListWithNumbers),
          member((Noun5,noun,M2),ListWithNumbers),
          M1<M2,
          Dist is M2-M1,
          windowsLength(WL),
          Lung is WL/2,
          Dist < Lung
```

```
      ),L3), /* trova le coppie del tipo (Nome,Nome), con nomi vicini
rispetto alle dimensioni della finestra */
      append(L1,L2,L12),append(L12,L3,Assoc). /* concat. i risultati */
```

4 Esperimenti e Risultati

Le sperimentazioni sono avvenute selezionando dal web testi di due-tre frasi appartenenti a tre differenti ambiti: testi letterari, testi scientifici, articoli di news.

Disambiguazione

Nei testi letterari le maggiori difficoltà nella disambiguazione sono dovute all'utilizzo dei nomi propri di persona come protagonisti delle vicende; il disambiguatore non riesce a trovare per loro nessuna associazione all'interno di WordNet, la parola va persa e con essa anche tutte le relazioni che la coinvolgono. Come riportato nella Sezione 5, un lavoro futuro sarà quello di rafforzare l'uso di WordNet con tecniche di riconoscimento automatico di nomi di entità.

Inoltre in questi testi è ricorrente la presenza di un soggetto che compie numerose azioni, ma vista la limitatezza della finestra solamente la prima o le prime due gli vengono attribuite. Per i testi letterari la percentuale di parole che trovano un corretto abbinamento in WordNet si aggira intorno al 65-70%.

Le stesse problematiche si riscontrano negli articoli di news, dove molto spesso il soggetto principale è un'azienda, un'impresa o un personaggio pubblico. Il contesto risulta essere però più circoscritto rispetto ai testi letterari; questi articoli raccontano infatti in modo non molto descrittivo vicende di cronaca o avvenimenti speciali in cui l'attenzione è focalizzata sul soggetto principale. Gli articoli di news solitamente non presentano le digressioni che caratterizzano i testi letterari e le percentuali di parole disambiguate correttamente migliorano, portandosi al 75-80%.

I testi scientifici hanno la caratteristica di essere specifici di un argomento; in questo modo tutte le parole "cooperano" per pervenire a una corretta disambiguazione. Il contesto in cui è inserito il testo risulta chiaro per la presenza di numerosi termini caratteristici. Si riduce drasticamente il numero di parole polimorfe e la percentuale di parole disambiguate correttamente sale al 90-95%.

Creazione delle relazioni

La percentuale di relazioni esistenti trovate risulta nella grande maggioranza dei casi inferiore alla percentuale delle parole correttamente disambiguate. Questo è dovuto principalmente a due fattori:

1. la lunghezza limitata della finestra;
2. la scorretta disambiguazione di un concetto.

Il primo fattore porta termini, tra i quali esiste una relazione, fuori dai limiti della finestra, impedendo una possibile associazione tra questi concetti, nonostante a essi sia stato attribuito il giusto significato. Un aumento della lunghezza della finestra porta a un numero maggiore di relazioni individuate ma fa aumentare la complessità e il tempo del calcolo.

Effetti deleteri derivano dall'attribuzione di un significato errato a una parola, nel caso in cui gli sia abbinato un synset appartenente a una categoria sintattica differente da quella che in realtà si intendeva nel testo. Ad esempio, l'attribuzione di un significato della categoria degli aggettivi a una parola che intendeva essere un verbo al passato, porta alla mancata creazione delle relazioni che coinvolgevano quel verbo.

Il numero di relazioni trovate nei testi scientifici continua a rimanere alto proprio grazie alla corretta disambiguazione. La percentuale di relazioni trovate è mediamente dell'80%.

Negli articoli di news la presenza di errori nell'attribuzione del significato si fa sentire maggiormente. In questi articoli le relazioni trovate sono il 66-67%.

Nei testi letterari si aggiunge la difficoltà nel collegare soggetti distanti dall'azione che compiono all'interno del testo. La presenza di lunghe digressioni spinge i concetti a una distanza tale da rendere in certi casi impossibile ricostruire la relazione. In questo caso la percentuale arriva al 53-54%.

A titolo di esempio, riportiamo nel seguito alcune relazioni significative tratte dal secondo libro delle Storie di Erodoto (esempi 1 e 2) e dalla fiaba "Il pescatore e sua moglie" dei fratelli Grimm (esempi 3 e 4):

1. Frase: *"There are other lilies too, in flower resembling roses, which also grow in the river"*; ruoli assegnati a concetti: *role(lilies, plants), role(roses, shrub)*; relazioni tra concetti: *resemble(lilies, roses)*

2. Frase: *"for the rest of Egypt becomes a sea and the cities alone rise above water"*; ruoli assegnati ai concetti: *role(Egypt, African nation), role(sea, water)*; relazioni tra concetti: *become(Egypt, sea)*

3. Frase: *"So the man went home, and saw his wife standing at the door of a nice trim little cottage"*; ruoli assegnati ai concetti: *role(man, male), role(home, housing), role(saw, consider), role(wife, partner), role(door, barrier), role(cottage, house)*; relazioni tra concetti: *go(man, home), see(man, wife), stand(wife, door), little(cottage).*

4. Frase: *"This time the sea looked a dark grey colour, and was overspread with curling waves and the ridges of foam as he cried out"*; ruoli assegnati ai concetti: *role(sea, water), role(grey, color), role(overspread, cover), role(waves, movement), role(ridges, elevation), role(foam, bubble)*; relazioni tra concetti: *look(sea, dark), look(sea, grey), overspread(sea), curling(waves), ridges(foam).*

5 Lavori collegati e sviluppi futuri

Alcuni dei principali contributi allo studio dei ruoli semantici nell'ambito del trattamento del linguaggio naturale e della linguistica computazionale riguardano l'estrazione di pattern lessico sintattici [Hea92] e di dipendenze sintattiche tra parole o tra i verbi e i loro argomenti [AD01, GJ02, CJ00], allo scopo di individuare relazioni semantiche basate sulla co-occorrenza di parole o classi di parole o sulla struttura logico-argomentativa di un verbo.

Un uso massiccio di tali tecniche è presente nell'ambito dell'apprendimento automatico di ontologie, sia per estrarre gerarchie di concetti, istanze e relazioni [Cim06, CHS05, PS05], che per il riconoscimento automatico di entità nominali quali persone, luoghi, eventi [ECD+05, YE09] da corpora annotati, thesauri e contenuti sempre più vasti e disponibili, non strutturati o semi-strutturati, quali quelli disponibili sul Web. Nella maggior parte dei lavori citati vi è un impiego prevalente di metodi di apprendimento statistico e modelli probabilistici.

L'approccio descritto in [NV04] utilizza invece un algoritmo di disambiguazione del senso, usato per interpretare i concetti e le loro relazioni semantiche, basato sul riconoscimento di pattern semantici estratti da WordNet (synset, relazioni, glosse) e da corpora annotati, codificati in una grammatica context-free che ne descrive le strutture, chiamate interconnessioni semantiche. La disambiguazione avviene con un algoritmo iterativo che, dato un termine, il suo contesto e tutte le possibili interconnessioni semantiche, utilizza pesi assegnati a ciascuna produzione della grammatica, fino ad ottenere quella che viene chiamata interpretazione composizionale, rappresentata dalla regola di produzione con peso maggiore.

Nessuno degli approcci citati utilizza l'algoritmo di Lesk adattato per estrarre ruoli e relazioni tra ruoli all'interno di testi; l'utilizzo di tale algoritmo in questo ambito rappresenta quindi un nostro contributo originale.

Per quanto riguarda gli sviluppi futuri, oltre all'individuare le tecniche più adatte ad affrontare problemi classici nel campo della WSD quali l'uso di pronomi anaforici e cataforici, di deittici e di anafore, ci concentreremo sul miglioramento del nostro algoritmo per quel che riguarda:

- l'individuazione di segmenti di testo tramite la distinzione della punteggiatura che termina il periodo rispetto a quella che delimita le abbreviazioni o le iniziali dei nomi seguiti da un punto. L'algoritmo, allo stato attuale, riconosce tutti i segni di punteggiatura come terminatori di un periodo.
- l'individuazione di nomi propri di persona, azienda o luogo, non presenti in WordNet e pertanto eliminati durante l'analisi del testo. L'eliminazione comporta il mancato riconoscimento di relazioni all'interno delle frasi, riducendo di molto l'informazione estratta. Una possibile soluzione potrebbe essere quella di adottare approcci per il riconoscimento automatico di nomi di entità.

Una valutazione qualitativa delle metriche usate e un'analisi sull'ottimizzazione dei parametri che tenga conto delle caratteristiche dei contentuti testuali nei diversi contesti è un lavoro futuro che ci proponiamo di condurre. Un'analisi comparativa del nostro approccio con strumenti proposti in letteratura e considerati stato dell'arte sia nell'ambito dell'IE che della WSD è altresì parte delle nostre prospettive future.

Riferimenti bibliografici

[AD01] E. Agirre and D. Martinez. Learning class-to-class selectional preferences. In Proc. Of ConLL'01, pp 1-8. Association for Computational Linguistics, 2001.

[AE06] E. Agirre and P. Edmonds . Word Sense Disambiguation: Algorithms and Applications. Text, speech, and language technology series, Vol 33, Springer, 2006.

[Bog07] A. Bogdanovych: Virtual Institutions. PhD Thesis, University of Technology Sydney (UTS), Australia. 2007.

[BP02] S. Banerjee and T. Pedersen. An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet, LNCS, Vol. 2276, pp. 136 - 145, 2002.
[BPA+09] A. Bogdanovych, L. Papaleo, M. Ancona, V. Mascardi, G. Quercini, S. Simoff, A. Cohen, and A. Traverso. Integrating Agents and Virtual Institutions for Sharing Cultural Heritage on the Web. In Proc. of the Workshop On Intelligent Cultural Heritage, 2009.
[CHS05] P. Cimiano, A. Hotho, and S. Staab. Learning concept hierarchies from text corpora using formal concept analysis. Journal of Artificial Intelligence Research, 24:305-339, 2005.
[Cim06] P Cimiano. Ontology Learning and Population from Text: Algorithms, Evaluation and Applications. Springer, 2006.
[CJ00] M. Ciaramita and M. Johnson. Explaining away ambiguity: Learning verb selectional preference with bayesian networks. In Proc. Of COLING, Morgan Kaufmann, 2000.
[CL96] J. Cowie, & W. Lehnert. Information Extraction, in (Y. Wilks, ed.) Special NLP Issue of the Comm. ACM, 1996.
[Cov93] M. A. Covington, Natural Language Processing for PROLOG Programmers. 1st. Prentice Hall PTR, 1993.
[ECD+05] O. Etzioni, M.J. Cafarella, D. Downey, A.M. Popescu, T. Shaked, S. Soderland, D.S. Weld, and A. Yates. Unsupervised named-entity extraction from the web: An experimental study. Artificial Intelligence, 165 (1): 91-134, 2005.
[ERS+01] M. Esteva, J. A. Rodríguez-Aguilar, C. Sierra, P. Garcia, and J. L. Arcos. On the Formal Specifications of Electronic Institutions. In Proc. Of AMEC'01, pp. 126-147, 2001.
[Etz64] A. Etzioni. Modern Organizations. Englewood Cliffs, N.J.: Prentice-Hall, 1964.
[GJ02] D. Gildea and D. Jurafsky. Automatic labeling of semantic roles. Computational Linguistics, 28:245-288, 2002.
[Hea92] M. Hearst. Automatic acquisition of hyponyms from large text corpora. In Proc. Of COLING 92, pp. 539-545, 1992.
[Jar89] M. A. Jaro, Advances in record linking methodology as applied to the 1985 census of Tampa Florida. Journal of the American Statistical Society 84 (406): 414-20, 1989.
[Lag00] T. Lager. A Logic Programming Approach to Word Expert Engineering. In Proc. of ACIDCA 2000.
[Les86] M. Lesk, Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone. In Proc. of ICSD, pp. 24 - 26, 1986.
[Mil95] G. A. Miller. WordNet: A Lexical Database for English. Communications of the ACM Vol. 38, No. 11. pp. 39-41, 1995.
[Nor90] D. North . Institutions, Institutional Change and Economic Performance. Cambridge University Press, 1990.
[NV04] R. Navigli and P. Velardi. Learning domain ontologies from document warehouses and dedicated websites. Computational Linguistics, 30(2):151-179, 2004.
[PS05] P.Cimiano and S.Staab. Learning concept hierarchies from text with a guided hierarchical clustering algorithm. In Proc. of Learning and Extending Lexical Ontologies with Machine Learning Methods Workshop, 2005.
[Sar08] S. Sarawagi. Information extraction. Foundations and Trends in Databases, 1(3):261-377, 2008.
[Sch03] J. G. Schlachter, ProNTo_Morph: Morphological Analysis Tool, 2003. Available from http://www.ai.uga.edu/mc/pronto/Schlachter.pdf
[Sea69] J. R. Searle. Speech Acts. Cambridge University Press, 1969.
[Wil09] Y. Wilks, Machine Translation. Its Scope and Limits, Departement of Computer Science, The University of Sheffield, 2009.
[Win99] W. E. Winkler, The state of record linkage and current research problems. Statistics of Income Division, Internal Revenue Service Publication R99/04. 1999.
[YE09] A. Yates and O. Etzioni. Unsupervised methods for determining object and relation synonyms on the web. Journal of Artificial Intelligent Research, 34(1):255-296, 2009.