Università degli Studi di Modena e Reggio Emilia

Facoltà di Ingegneria – Sede di Modena

Corso di Laurea in Ingegneria Informatica – *Nuovo Ordinamento*

Studio e sperimentazione dell'annotazione lessicale delle ontologie definizionali

Relatore:
Prof. Sonia Bergamaschi

Candidato:
Elisa Fusari

Correlatore:
Ing. Serena Sorrentino

Anno Accademico 2008-2009

Parole chiave:

Top-ontology
Annotation
WordNet
DOLCE

Indice

Introduzione

Il Web è un'immensa rete di collegamenti tra elementi che contengono svariati tipi di risorse, e la sua sempre maggiore diffusione ha creato problemi sia nel reperimento delle informazioni sia nello sviluppo di sistemi di integrazioni tra sorgenti eterogenee [**1**].

La ricerca nel Web è ostacolata dal fatto che gli utenti devono specificare i loro concetti per mezzo di parole chiave, infatti la comprensione automatica del linguaggio è limitata dalla sua ambiguità e, per far proseguire il progresso nell'e-commerce e nell'integrazione software, è necessario dare ai computer un linguaggio comune con una ricchezza più simile al linguaggio dell'uomo.

Il problema sta principalmente nel rapporto tra parole e contenuti, infatti questi collegamenti sono spesso deboli, troppo generici e vaghi.
Nel Web, le pagine sono collegate tra loro da legami sintattici, costituiti da indici che localizzano le loro URL in modo univoco, e quindi piuttosto solidamente.
Ma i legami, oltre a portare in un determinato luogo, dovrebbero descrivere anche il luogo in cui conducono: il problema sta principalmente nella debolezza dei legami semantici, che dovrebbero definire i significati delle pagine o dei dati che si stanno utilizzando.
Questa difficoltà è facilmente comprensibile se si pensa alle tante possibilità con cui si può definire, descrivere e rappresentare una stessa porzione di realtà: anche avendo una base comune di conoscenze, non si può essere certi che gli stessi elementi saranno sempre espressi con gli stessi termini, e viceversa che ciascuna parola farà sempre riferimento ad un'entità univoca.

Per fare fronte alla problematica si è introdotto il concetto di annotazione, il quale fa riferimento a entità artificiali che vengono utilizzate per descrivere un documento, o una sua parte o addirittura una sua intera collezione.
L'annotazione può essere sia manuale, attraverso i linguaggi di mark-up come XML, xml schema, ecc; sia automatica, utilizzando metodi di elaborazione automatica del linguaggio naturale, i quali costituiscono ancora una buona parte del lavoro di ricerca nel settore.
Essa è formata da coppie *(l, def)*, dove *l* è una *label* o etichetta e *def* è una definizione più o meno formalmente espressa del suo significato ed uso inteso.
L'annotazione di un termine racchiude in sé anche il concetto di disambiguazione del testo [**2**], il quale consiste nell'assegnare ad ogni parola il suo significato o senso più corretto in base al contesto in cui è utilizzata.
La disambiguazione, presuppone quindi l'utilizzo di una sorgente lessicale, solitamente un thesaurus o un'ontologia, da cui reperire le informazioni relative ai concetti e alle relazioni che vi intercorrono. In letteratura, le principali tecniche di disambiguazione del linguaggio naturale, utilizzano come sorgente lessicale WordNet. WordNet, può essere definito come un database lessicale in lingua inglese, il quale tuttavia, nel corso del suo utilizzo, ha presentato alcune lacune, sia rispetto ai termini in esso contenuti, sia rispetto alle relazioni semantiche che vi intercorrono.

Per questo la ricerca ha unito i campi delle scienze informatiche, linguistiche e filosofiche, avendo tutte queste discipline esperienza nel campo della creazione di descrizioni e terminologie standard per tutti i tipi di entità, in modo da creare un'ontologia formale con lo scopo di trovare un compromesso tra il linguaggio impoverito e specifico delle macchine e quello onnicomprensivo ed ambivalente dell'uomo.

L'obiettivo di questa tesi è dunque quello di analizzare le ontologie fondamentali più importanti e conosciute, osservando le loro strutture e i loro utilizzi. Si eseguirà, inoltre, l'annotazione di tali ontologie rispetto a WordNet, attraverso l'utilizzo di un componente di annotazione lessicale (ALA–Automatic Lexical Annotator) integrato all'interno di MOMIS (Mediator envirOnment for Multiple Information Sources), un sistema di integrazione dei dati strutturati e semi-strutturati che utilizza il database lessicale WordNet come risorsa di riferimento. Infine, saranno confrontati i risultati sperimentali conseguiti per verificare se le ontologie possano avere elementi in comune o essere allineate a WordNet, e se sia quindi

possibile un loro futuro utilizzo come mezzo alternativo o complementare all'annotazione stessa.

Il contenuto della tesi è così strutturato:

- *Capitolo 1: Il Thesaurus WordNet*. Il capitolo ha lo scopo di illustrare brevemente quello che sono le caratteristiche strutturali e il contenuto del database lessicale WordNet.
- *Capitolo 2: Le ontologie*. Il capitolo ha il compito di chiarire cosa si intende per ontologia, le sue tipologie, la terminologia usata, le tecniche per il matching e le problematiche date dall'uso di WordNet come ontologia.
- *Capitolo 3: L'ontologia Cyc.* Nel capitolo si descrivono le principali caratteristiche dell'ontologia definizionale Cyc, la struttura, i contenuti e i suoi utilizzi.
- *Capitolo 4: L'ontologia SUMO.* Nel capitolo si analizzano le principali caratteristiche dell'ontologia definizionale SUMO, la struttura, i contenuti e i suoi utilizzi.
- *Capitolo 5: L'ontologia DOLCE*. Nel capitolo si esaminano le principali caratteristiche dell'ontologia definizionale DOLCE, la struttura, i contenuti e i suoi utilizzi.
- *Capitolo 6: Analisi delle ontologie definizionali*. Il capitolo ha lo scopo di descrivere gli strumenti usati e la procedura seguita per analizzare le ontologie, illustrando più approfonditamente il linguaggio OWL e i software Protégè e MOMIS.
- *Capitolo 7: Annotazione di DOLCE Lite +*. Nel capitolo si presentano il procedimento usato per l'annotazione, manuale ed automatica, della versione di DOLCE utilizzata ed il confronto dei risultati ottenuti.
- *Capitolo 8*: *Conclusioni e Sviluppi futuri*. Il capitolo ha lo scopo di riassumere gli esiti e di presentare i possibili proseguimenti della sperimentazione eseguita.

Capitolo 1

1. Il Thesaurus WordNet

Il database lessicale maggiormente utilizzato e riconosciuto è WordNet [**3**]: esso è stato sviluppato presso l'università di Princeton sotto la direzione del professore George A. Miller ed è disponibile gratuitamente anche per fini commerciali ed al di fuori della ricerca presso il sito www.cogsci.princeton.edu/wn, a condizione che siano citati gli autori ed il sito ufficiale del progetto.

Esso non è un semplice dizionario di termini inglesi, ma un sistema lessicale di riferimento il cui disegno si basa sulla memoria umana e sulle teorie psico-linguistiche: i vocaboli, infatti, non sono disposti in ordine alfabetico, ma per affinità di significato. WordNet comprende quattro categorie sintattiche: nomi, verbi, aggettivi e avverbi, ciascuna delle quali è suddivisa in diversi insiemi di sinonimi; questi ultimi sono associati ad un unico significato condiviso da tutti i termini che li compongono. Un termine ovviamente può possedere più di un significato ed essere presente in molti di questi insiemi, ed anche in più di una categoria sintattica.

1.1 La terminologia di WordNet

WordNet usa una particolare terminologia per indicare i suoi elementi. Le principali definizioni sono:

- *Categoria Sintattica*: sono le grandi categorie in cui sono suddivisi i termini (ed anche i file in cui sono contenuti) di WorNet (nomi, verbi, aggettivi, avverbi);
- *Lemma*: è la parola/termine a cui vengono associati uno o più significati; a volte un lemma può essere composto da due o più parole le quali sono quindi unite dal carattere *underscore* (_);

- *Synset*: rappresenta un significato che viene associato ad un insieme di lemmi appartenenti alla stessa categoria sintattica; in pratica ad un synset corrispondono più lemmi ed è quindi rappresentabile, oltre che dalla sua glossa, anche dall'insieme dei suoi lemmi;
- *Glossa*: rappresenta la descrizione a parole di un significato specifico; ogni synset, oltre a contenere un insieme di sinonimi, possiede anche una glossa;
- *Relazione Semantica*: è una relazione di WordNet che lega due synset appartenenti alla stessa categoria sintattica;
- *Relazione Sintattica*: è una relazione che collega due lemmi appartenenti a due synset distinti (ma sempre appartenenti alla stessa categoria sintattica).

1.2 La matrice lessicale

La base della semantica lessicale, è la consapevolezza dell'esistenza di un'associazione tra la forma di una parola (il modo in cui è scritta e letta) e il significato ad essa legato; una corrispondenza non di tipo univoco ma di tipo molti a molti, la quale dà origine ai concetti di:

- *Sinonimia*: proprietà per cui lo stesso significato è esprimibile tramite l'uso di due o più parole distinte;
- *Polisemia*: proprietà per cui ad una stessa parola sono associati due o più significati distinti; queste parole ambigue dal punto di vista del significato, sono dette *polisemiche*, per distinguerle da quelle con un significato univoco che sono dette *monosemiche*.

WordNet rappresenta le relazioni tra le parole e i diversi significati, costruendo la cosiddetta *matrice lessicale* (Figura 1).

In tale matrice, le righe rappresentano i significati che è possibile attribuire ad una parola, mentre le colonne, rappresentano i diversi termini: volendola leggere usando la terminologia di WordNet, si ha che ad ogni riga è associato un synset, e ad ogni colonna un lemma.

	W_1	W_2	W_3	W_4	W_5
M_1	$E_{1,1}$				
M_2		$E_{2,2}$			
M_3		$E_{3,2}$	$E_{3,3}$		
M_4					
M_5				$E_{5,4}$	
M_6					$E_{6,5}$

Figura 1: Matrice Lessicale

Ogni elemento non nullo interno alla matrice, indica che il particolare lemma o termine, situato in quella riga, può essere usato per rappresentare il significato associato a quella colonna.

Se all'interno di una colonna sono contenuti più elementi, si ha un caso di polisemia; se, al contrario, due o più elementi compaiono sulla stessa riga, si è in presenza di un caso di sinonimia.

In WordNet il concetto di matrice lessicale, si esprime mantenendo la separazione tra lemmi e synset, ovvero tra termini e significati. Un synset è espresso nei file usati in WordNet, tramite l'insieme dei termini che sono ad esso associati; tuttavia, un insieme di parole di questo tipo, non è spesso sufficiente a descrivere un significato, così è associata a ciascun synset anche una descrizione del significato, attraverso la glossa.

1.3 Le relazioni

WordNet si contraddistingue da un semplice dizionario, anche per la presenza di diversi tipi di relazioni sia tra i synset che tra i lemmi, le quali permettono di creare, al suo interno, gerarchie di significato: si hanno relazioni semantiche quando gli operandi sono synset, mentre si hanno relazioni lessicali se gli operandi sono lemmi.

Non possono invece esistere relazioni tra operandi appartenenti a categorie sintattiche differenti. All'interno di WordNet tutte le relazioni sono rappresentate tramite puntatori e caratteri speciali, che indicano il tipo di relazione specificata.

1.3.1 Le relazioni semantiche

Le relazioni semantiche coinvolgono sempre due concetti, due synset e quelle principali sono:

- *Iponimia/Ipernimia*: possono essere considerate l'equivalente delle gerarchie di specializzazione/generalizzazione per database relazionali o per l'ereditarietà dei modelli ad oggetti e sono le relazioni più numerose all'interno del database lessicale di WordNet; una relazione semantica di questo tipo è valida solo per le categorie di nomi e verbi; l'iponimia lega un concetto ad uno più generale (un synset X è un iponimo di un synset Y, se è corretta l'affermazione "X è un tipo di Y") mentre l'ipernimia lega un concetto ad uno più particolare, più specializzato (un synset X è ipernimo di un synset Y, se Y presenta tutte le caratteristiche di X più una sua caratteristica particolare ed aggiuntiva);
- *Meronimia/Olonimia*: un concetto X è meronimo di un concetto Y se è lecito affermare che "X è una parte di Y", mentre l'olonimia indica la relazione inversa; entrambe queste relazioni si applicano alla categoria sintattica dei nomi;
- *Implicazione*: è una relazione posta tra due verbi ed è verificata se è vera la seguente affermazione, "un verbo X implica un verbo Y, se X non può verificarsi a meno che non si sia verificato (o non si stia verificando) Y"; non è solo una relazione semantica ma può anche avere delle implicazioni lessicali tra i verbi (tra singoli termini);
- *Relazione causale*: è una relazione simile all'implicazione ma senza inclusione temporale;
- *Raggruppamento di verbi*: relazione utilizzata per produrre gruppi nella categoria sintattica di verbi, in cui i synset hanno tutti un significato semantico molto simile;
- *Antinomia/Similarità*: la relazione è utilizzata nella categoria sintattica degli aggettivi; la prima lega le coppie di synset che sono in contrapposizione semantica tra loro (contrari), che vengono detti synset primari; ad essi sono collegati per similarità dei synset satelliti, che condividono indirettamente le relazioni di antinomia insieme al significato principale a cui sono legati (antinomia indiretta → similarità);
- *Attributo*: rappresenta il legame che intercorre tra un aggettivo ed un nome di cui esprime il valore; gli aggettivi in grado di esprimere il valore di un attributo sono gli aggettivi descrittivi (es. alto → altezza);

- *Coordinazione*: è un tipo di relazione derivata, infatti due synset si dicono coordinati se possiedono lo stesso ipernimo, ovvero se risultano essere la specializzazione del medesimo concetto.

1.3.2 Le relazioni lessicali

L'altra tipologia di relazione è quella lessicale, la quale coinvolge sempre due lemmi e non due synset; le principali sono:

- *Sinonimia*: la relazione, diversamente dalle altre, non è espressa formalmente attraverso l'uso di puntatori, ma è rappresentata tramite l'appartenenza da parte dei due vocaboli sinonimi, allo stesso synset; infatti, il termine synset deriva dall'espressione set of synonym ed è stato coniato proprio per dare questa idea; per ogni coppia di termini appartenenti allo stesso synset, esiste quindi, una relazione di sinonimia implicita.

  La sinonimia è definita in due modi:

  1. Due termini sono sinonimi se la sostituzione di uno per l'altro non cambia mai il valore della frase in cui è fatta la sostituzione (Leibniz).
  2. Due termini sono sinonimi, all'interno di un contesto linguistico C, se la sostituzione di un termine con l'altro, all'interno di C, non varia il valore della frase (definizione relativa ad un contesto).

  WordNet adotta la seconda definizione, più permissiva e semplice (risulta molto difficile trovare due sinonimi in senso lato, indipendentemente dal tipo di contesto, come descritto da Leibniz nella prima definizione) concludendo che due lemmi sono sinonimi solo all'interno di uno stesso contesto, e di un certo synset (non possono essere sinonimi due termini appartenenti a categorie sintattiche differenti).
- *Antinomia*: è la relazione lessicale tra due lemmi che sono l'uno il contrario dell'altro; infatti, non sempre è corretto considerare l'antinomia come relazione semantica, ad esempio i synset possono essere concettualmente opposti ma la vera relazione di antinomia si stabilisce solo tra i singoli lemmi;
- *Relazione di pertinenza*: concerne gli aggettivi relazionali, i quali svolgono un ruolo che è riassunto con le espressioni "associato con", "pertinente a" oppure "di" in relazione ad un nome; l'aspetto di un aggettivo relazionale, risulta molto simile a quello del nome a cui è legato, leggermente modificato (es. mentale → mente);

- *Vedi anche*: è una relazione lessicale che lega singoli lemmi di synset differenti, con motivi molto diversi tra loro;
- *Relazione partecipiale*: è una relazione che lega tra loro avverbi e aggettivi, detti partecipiali, rispettivamente ai nomi o ai verbi da cui derivano (es. bruciato → bruciare);
- *Derivato da*: alcuni aggettivi derivano da antichi nomi Greci o Latini, (ciò è vero sia per la lingua italiana che per la lingua inglese su cui è costruito WordNet), e questa relazione lega gli aggettivi odierni ai nomi stranieri da cui provengono.

Capitolo 2

2. Le ontologie

Un altro strumento utilizzato nell'annotazione e nella disambiguazione del significato delle parole è l'ontologia [**4**]. Per ontologia si intende un insieme di concetti (detti anche classi), legati da interconnessioni semantiche (dette anche relazioni concettuali o attributi semantici) ed un eventuale livello logico che permetta di inserire nuovi fatti a partire da quelli già codificati all'interno della risorsa (ad esempio un insieme di assiomi o di micro-teorie).
La distinzione che si ha tra un normale database e un'ontologia è che in quest'ultima si ha l'esistenza di un modello semantico teorico: l'ontologia è una teoria logica. L'interpretazione di un'ontologia non è lasciata all'utente che legge il diagramma o al sistema che l'ha implementata, ma è esplicitata: la semantica contiene le regole per l'interpretazione della sintassi, che non implica la definizione diretta dei significati, ma vincola la possibilità di interpretazione di quello che si è dichiarato.

L'esponenziale crescita della comunicazione attraverso la rete ha favorito il fenomeno dell'utilizzo di ontologie nella caratterizzazione delle informazioni, accrescendone il loro valore. Mediante l'organizzazione ontologica è infatti possibile velocizzare, semplificare e migliorare il reperimento, la classificazione e l'integrazione delle informazioni.

Quest'area di ricerca è tuttora in pieno sviluppo partendo dai modelli di comunicazione ai metodi per l'integrazione di basi di dati, all'analisi di consistenza e sicurezza dei sistemi informatici e perfino al supporto dell'apprendimento. Il Semantic Web ne è l'esempio più importante: lo scambio di dati attraverso il Web comporta l'interazione tra persone e sistemi eterogenei in cui è necessario avere un sistema comune e condiviso per la comunicazione e comprensione dell'informazione.

Per ottenere un modello univoco da utilizzare nell'organizzazione dell'informazione si necessita in primo luogo della risoluzione dei problemi riguardanti l'ambiguità semantica, ovvero deve essere affrontata la problematica della disambiguazione del significato delle parole. Tale problematica è uno dei principali temi che i ricercatori, che si occupano di *ontologia applicata* [**5**] cercano di risolvere: l'attribuzione dell'esatto significato a ciascun termine è effettuata tramite l'analisi del termine stesso e del contesto in cui è utilizzato, ovvero dall'esame di più termini legati tra loro da relazioni semantiche, i quali costituiscono quindi "una particolare concettualizzazione del mondo" (N. Guarino).

Le ontologie sono usate sia per dare un accesso semantico alle risorse specifiche, in questo caso il significato di un termine è in parte già conosciuto e l'ontologia è limitata a quelle relazioni strutturali che sono rilevanti per le query; sia per la negoziazione del significato delle parole, ovvero mettersi d'accordo su quale senso attribuirvi (ottenere un *consenso*) a seconda del contesto in cui sono utilizzate, dando inoltre la possibilità di cooperazione sia tra agenti artificiali che tra agenti artificiali e persone (comunicazione uomo-macchina).

2.1 Ontologie di dominio e ontologie definizionali

Le ontologie sono principalmente di due tipi: si parla di ontologie pesanti e ontologie leggere.
Le prime sono anche dette o*ntologie definizionali* (o *fondazionali* o *top-ontology*): queste utilizzano un linguaggio formale molto espressivo e sono caratterizzate dalla presenza di numerose relazioni tra i propri elementi costitutivi, le classi (o categorie); ciò consente una migliore rappresentazione delle scelte strutturali dell'ontologia. Esse sono utilizzate come ponte tra le concettualizzazioni alternative e come traduttori del significato inteso nei diversi ambienti, grazie alla possibilità che danno di definire una struttura terminologica trasversale alle comunità d'uso.

Le seconde invece sono spesso delle semplici tassonomie, in cui il significato dei termini si suppone noto e condiviso nell'ambiente in cui sono utilizzate.
Le ontologie *di dominio* o *particolari* [**5**], sono ontologie che possono dare diverse visioni degli stessi spaccati di realtà oppure occuparsi di realtà diverse ma contenenti elementi analoghi: è necessario quindi sviluppare strumenti di supporto alla comunicazione tra applicazioni che usufruiscono di diverse ontologie. Tali strumenti sono fondamentali per la condivisione del significato generale dei termini e sono fondati a loro volta su ontologie di

carattere sempre più generale e rigoroso rispetto a quelle ontologie minimali che sono adottate in ambienti ristretti e specifici.

D'altra parte non si può neppure parlare di un approccio monolitico riguardante lo sviluppo di una teoria onnicomprensiva delle ontologie fondazionali, ovvero la creazione di un'unica gigantesca ontologia contenente tutta la conoscenza disponibile, la quale sarebbe difficilmente sostenibile sia dal punto di vista teorico che applicativo date le incompatibilità già presenti nei livelli più basilari. È quindi plausibile avere un approccio di tipo modulare in cui i singoli moduli ontologici siano il più possibile integrabili tra loro, tramite relazioni formali; "è più importante permettere alle persone di capirsi l'una con l'altra, che rafforzare la loro cooperazione con l'utilizzo di un'unica ontologia" (Wonder Web Project [**16**]).

2.2 Terminologia dell'ontologia

Il linguaggio delle ontologie spesso contiene i seguenti tipi di entità:

- *Classe* (o concetto): sono le entità principali di un'ontologia e sono interpretate come un set (insieme) di individui nel dominio;
- *Individuo* (o oggetto o istanza): sono visti come i particolari elementi del dominio;
- *Relazione* (o predicato): è l'ideale nozione di legame, indipendentemente a ciò a cui si applica, esse sono interpretate come un sottoinsieme del dominio dell'ontologia;
- *Tipo di dato*: sono le parti dei domini che specificano i valori assunti dagli individui;
- *Valore del dato*: sono semplici valori e non hanno identità.

Le entità possono essere collegate da diversi tipi di relazioni, che si dividono principalmente in:

- *Specializzazione*: si stabilisce tra due classi o tra due proprietà ed è intesa come l'inclusione dell'interpretazione di una nell'altra;
- *Esclusione*: si stabilisce tra due classi o tra due proprietà ed è intesa come l'esclusione dell'interpretazione di una dall'altra, cioè quando l'intersezione tra le due è vuota;
- *Instanziazione*: si stabilisce tra individui e classi, o tra istanze di proprietà e proprietà, o tra valori e tipi di dati ed è intesa come il legame di appartenenza, "essere membro di".

2.3 Classificazione degli approcci di matching tra ontologie

Per *matching* si intende il processo utilizzato per trovare relazioni o corrispondenze tra entità di differenti ontologie. L'allineamento di una o più ontologie è il risultato di questo processo (*mapping*). Per scoprire le corrispondenze semantiche, fra gli elementi di diverse ontologie esistono differenti metodi [**6**], i quali combinati tra loro assicurano l'analisi completa e approfondita del loro dominio: maggiore è il numero di metodi utilizzato migliore e più dettagliato risulterà il matching.

Le metodologie di matching si differenziano per tipo di input, processo utilizzato e tipo di output; esse sono state classificate secondo i seguenti criteri:

- *Completezz*a: l'unione delle sottocategorie deve coprire tutta l'estensione della categoria superiore;
- *Separazione*: le sottocategorie che dividono una categoria devono essere disgiunte per costruzione in modo da realizzare uno schema ad albero appropriato;
- *Omogeneità*: il criterio usato per le divisioni successive di una categoria deve essere della stessa natura;
- *Saturazione*: l'insieme delle tecniche di matching usate deve essere il più specifico e diversificato possibile in modo da ottenere una fine distinzione tra le possibili categorie alternative.

Sia le tecniche già esistenti che quelle più nuove, se possiedono queste proprietà, garantiscono la stabilità delle ontologie e possono essere implementate in algoritmi che restituiscono sia risultati esatti che approssimati, a seconda dello scopo del sistema di matching.

La classificazione delle tecniche automatiche per il matching fra categorie, inizia con la distinzione tra quelle elementari e quelle composte. La prima tipologia è analizzata attraverso due tassonomie ad albero che condividono le loro estremità-foglie, le quali rappresentano appunto le classi dei matchers elementari e i loro esempi concreti (Figura 2).

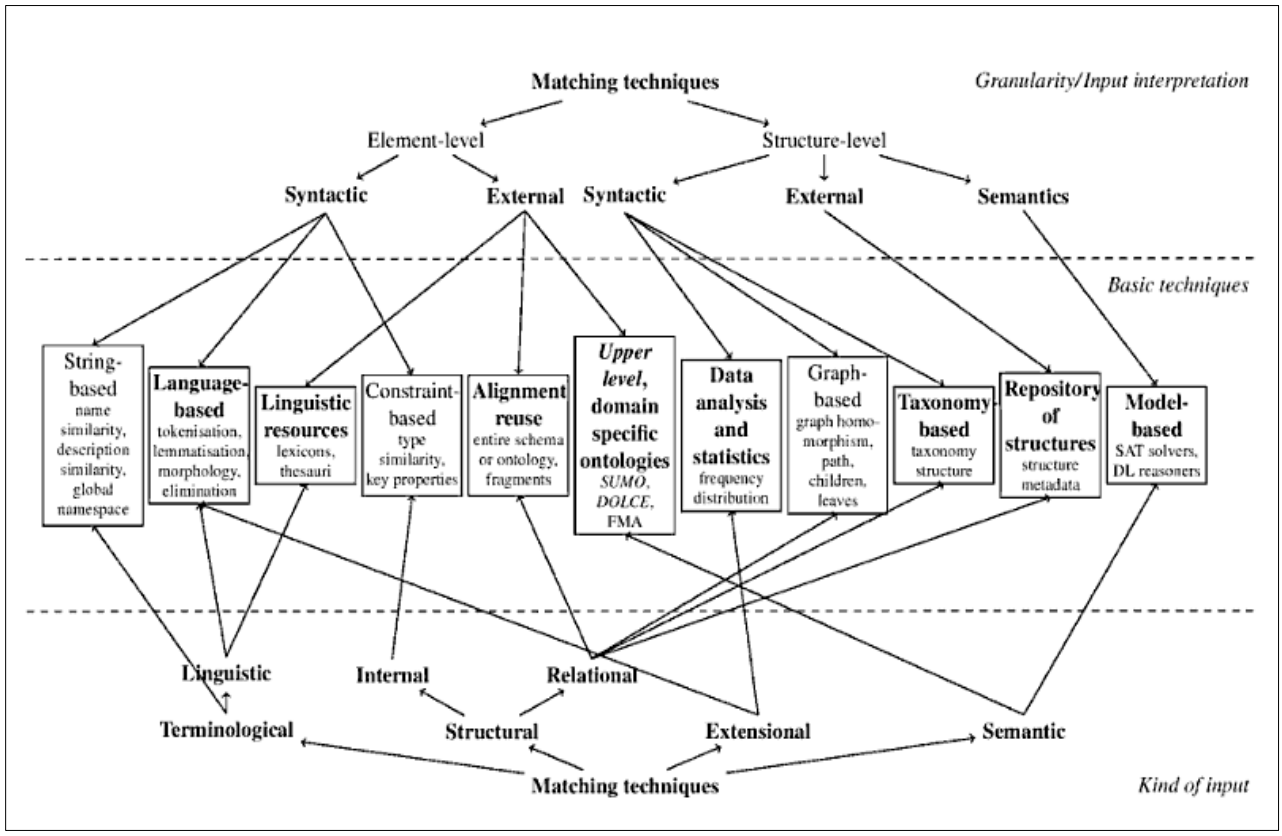

Figura 2: Classificazione degli approcci elementari di matching

La prima tassonomia classifica le tecniche in base alla *Granularità* (ovvero se operano a livello di elementi o di strutture) e all'*Interpretazione degli Input*, mentre la seconda usa come criterio il *Tipo di Input* (ovvero il tipo di oggetto che viene manipolato).

Nella prima tassonomia le tecniche si dividono in *Livello-Elemento* e *Livello-Struttura*, ovvero si analizzano le entità e le istanze isolate o collegate tra loro in disposizioni più complesse; quindi si ripartiscono entrambe in tecniche *Sintattiche*, *Esterne* e *Semantiche*: le prime interpretano l'input considerando la sua sola struttura, le seconde invece fanno anche uso di risorse ausiliarie e di conoscenza comune, come thesaurus; infine le ultime utilizzano dei modelli semantici formali per giustificare i loro risultati.

Le ontologie fondamentali più importanti (WordNet, DOLCE [**16**], SUMO [**14**], Cyc [**10**]…) si classificano proprio come tecniche di Livello-Struttura e di tipo Esterne, anche qui si potrebbero dettagliare ulteriormente in sintattiche o semantiche, ma in questa prima analisi, ciò è tralasciato.

Nella seconda tassonomia, invece, le tecniche si dividono, a seconda del tipo di dato su cui lavorano, in *Terminologiche* (stringhe), *Strutturali* (strutture), *Semantiche* (modelli) ed *Estensionali* (istanze di dati): le prime si specializzano ulteriormente in *Linguistiche* se si basano sull'interpretazione delle stringhe come oggetti linguistici e non solo come sequenze di caratteri; le seconde invece si suddividono ulteriormente in *Interne*, se considerano la struttura interna delle entità, e *Relazionali*, se considerano invece i legami tra di esse.

In conclusione, le principali classi di matchers risultano:

- *Tecniche String-based*, usate per collegare nomi e descrizioni delle entità di un'ontologia, considerando le parole come sequenze di caratteri e basandosi sulla somiglianza delle stringhe che denotano lo stesso concetto;
- *Tecniche Language-based*, considerano i nomi come parole di una certa lingua;
- *Tecniche Constraint-based*, usano algoritmi che si basano sui vincoli applicati alle entità;
- *Risorse Linguistiche*, usano thesauri o lessici specifici per legare i nomi con relazioni linguistiche (sinonimia, iponimia, ecc.);
- *Riutilizzo di Allineamento*, usa la somiglianza tra i collegamenti usati, soprattutto nei casi in cui le diverse ontologie trattino gli stessi argomenti o le ontologie siano di grandi dimensioni e quindi frazionate per poter essere allineate tra loro;
- *Ontologie Formali di Livello Superiore o di Dominio*, le prime sono usate come risorse esterne di tecniche basate sulla logica e la semantica, mentre le seconde servono come supporto alla conoscenza di particolari argomenti.

Di quest'ultimo caso non sono ancora stati realizzati dei sistemi di matching, ma è ragionevole affermare che ciò possa avvenire in un prossimo futuro (ad esempio si sta già pensando di utilizzare DOLCE per formalizzare la struttura di WordNet [**8**]).

Altre tecniche usate sono quelle di Livello-Struttura che considerano insiemi di entità e istanze collegate tra loro, ad esempio le tecniche basate sui Grafici, sulle Tassonomie, sui Modelli o su Insiemi di Strutture (frammenti di ontologie).

In particolare poniamo l'attenzione sulle tecniche basate sulla semantica: esse fanno riferimento a modelli teorici e sono quindi dei metodi deduttivi, ma per collegare diverse

ontologie tra loro è necessario prima allinearle ad un'ontologia generale, in modo da avere un punto di partenza per poi poter applicare i diversi metodi.

Infatti, quando due ontologie devono essere combinate, spesso si ha la mancanza di un terreno comune su cui basare il confronto; per questo si usano ontologie formali intermedie che definiscono un contesto comune o un *background* di conoscenze. Queste ontologie, coprono il dominio di interesse di quelle da combinare e sono essenziali nella disambiguazione dei molteplici possibili significati dei termini.

Le risorse esterne che costituiscono gli intermediari per il matching si diversificano secondo tre criteri:

- *Ampiezza*: ovvero se sono risorse generiche o di uno specifico dominio, le prime permettono di usare allineamenti già esistenti ed espandibili, mentre i secondi garantiscono una maggiore precisione nei collegamenti;
- *Formalità*: ovvero se sono ontologie formali con la propria descrizione semantica o risorse informali come WordNet; le prime permettono di creare relazioni tra i termini, mentre le seconde di ampliare il loro set di sinonimi e significati;
- *Stato*: ovvero che tipo di riferimento sono tra ontologie, thesauri, annotazioni di istanze ecc.

2.4 WordNet come ontologia

In precedenza si è trattato di WordNet come il principale database lessicale utilizzato nell'annotazione per la disambiguazione del testo; ma WordNet è spesso usato anche per la tassonomia-ontologia che contiene al suo interno grazie alle relazioni esistenti tra i vari synset [**7**].

La necessità di poter utilizzare WordNet come un'ontologia è nata dal problema che si riscontra nelle ontologie delle applicazioni: spesso non presentano un lessico sufficientemente ricco per spiegare le loro entità reali e le relazioni che intercorrono tra le loro classi.

L'annotazione ontologica rimane però necessaria per identificare entità reali, attraverso il riconoscimento di proprietà e relazioni che caratterizzano i loro attributi e ruoli in un particolare contesto, facendo sempre riferimento ad un'ontologia di base.

Si utilizza quindi, l'annotazione manuale, che può essere una soluzione se si tratta di domini ristretti, in cui i termini sono quasi tutti noti e l'ambiguità dei termini non è uno dei problemi principali, ma diventa impensabile per quelle applicazioni che richiedono un largo impiego di documentazione (basti pensare al numero di notizie giornaliere che ricevono i notiziari).

L'unica alternativa rimanente è quella dell'annotazione automatica. La scelta ricade sul database lessicale di WordNet, il quale possiede una grande rappresentazione strutturata di concetti lessicali che sono stati usati nello sviluppo degli algoritmi per la disambiguazione del significato delle parole.

Tuttavia si è visto come i tentativi di allineamento di WordNet a delle ontologie pre-esistenti fossero incompleti o inaccurati, visto l'enorme numero di synset presenti: i risultati sono stati spesso parziali o imprecisi, mentre l'idea di un allineamento manuale risulta improponibile in termini di costi e tempi di lavoro. È nata quindi l'idea di creare un'ontologia basata solo su WordNet.

Per prima cosa si è scelto un metodo per selezionare sia fino a che profondità spingersi nei livelli del database, e sia quali termini possano essere i migliori candidati a diventare entità e quali a rimanere solo delle istanze. Quindi si sono scelti come classi, i termini più generali e aventi una certa importanza semantica, in modo da mantenere una copertura di lessico più estesa possibile; il nome delle classi coincide con il termine più frequente nel synset mentre gli altri sinonimi rimangono solo istanze [**7**].

In questo modo si è ristretto notevolmente il numero di classi, riducendo a sua volta l'ambiguità dei termini e facendo ottenere risultati più positivi agli algoritmi di disambiguazione.

Quest'ultimi sono stati rielaborati in modo da considerare sia il contesto, che la sintattica e la semantica dei termini, calcolando inoltre la maggiore frequenza d'uso di ciascun vocabolo, per ottenere un'annotazione più precisa.

Il lavoro è stato svolto solo nella parte di WordNet riguardante i verbi, ma si è visto che può essere applicata in modo analogo anche alle altre categorie sintattiche, anche se con risultati inferiori.

2.4.1 Problemi ontologici di WordNet

WordNet presenta alcune problematiche [**8**] che devono essere risolte prima di poterlo considerare come una vera e propria ontologia lessicale, per esempio molte delle sue relazioni lessicali devono essere reinterpretate come relazioni semantiche.

Il primo problema critico che si è trovato in WordNet è la confusione tra concetti (synset) e singole istanze (lemmi); esplorando la sua tassonomia troviamo, infatti, come iponimi di uno stesso synset sia dei concetti complessi, generali e che possono essere degli ipernimi a loro volta, sia delle istanze semplici e particolari: per esempio possiamo vedere che sotto il termine "Organizzazione" troviamo i concetti "Compagnia", "Alleanza", "Federazione", "Comitato" insieme a delle istanze come "Croce_Rossa" o "Arma_Repubblicana_Irlandese".

Questo è il prodotto di una mancanza di espressività, infatti ci dovrebbero essere due tipi di specializzazione, una concetto-concetto e una concetto-istanza, in modo da distinguere le diverse relazioni.

Un'altra imprecisione presente in WordNet è la confusione che si ha all'interno del synset del termine Astrazione, il quale presenta sia concetti del livello Oggetto, come Spazio e Tempo, sia del Meta-livello, come Attributi e Relazioni. Ciò crea ambiguità nella definizione del termine che può quindi assumere due aspetti non compatibili.

Si è inoltre trovata nell'analisi una violazione del principio di rigidità, il quale è uno dei vincoli fondamentali che devono essere rispettati nella metodologia di OntoClean [**7**] (processo utilizzato per testare l'adeguatezza dei legami tassonomici di un'ontologia): la più comune violazione che si è trovata è legata alla distinzione tra ruolo e tipo. Infatti, spesso si creano delle relazioni di questo genere senza guardare prima i termini che dipendono da quelli considerati, i quali possono essere in contrasto con la relazione inserita o non presentare le caratteristiche essenziali che servono per sostenere la rigidità del legame: per esempio Persona è collegato sia a "Organismo" che ad "Agente_Causale", ma quest'ultimo non ha niente a che vedere col senso di "agente intenzionale", anzi racchiude termini come "Germicida" e "Vasocostrittore" che violano appunto il vincolo di rigidità.

Il problema della distinzione tra tipo e ruolo, si ripresenta anche nei livelli più bassi della tassonomia, in cui si registra un certo livello di eterogeneità nei concetti più generali: alcuni

termini hanno dei synset che mescolano vocaboli intuitivamente troppo specifici rispetto agli altri, ad esempio in “Animale” si mescolano sia le classi “Mammifero”, “Vertebrato” ecc. sia quelle di “Domestico”, “Da_Lavoro”, ecc. aumentando la confusione nei tipi di relazione.

Infine, è risaputo che WordNet riconosca la maggior parte dei significati convenzionali di una parola; nell’analisi effettuata è però risultato che a volte una polisemia anche importante non è rilevata. Ad esempio vi possono essere degli ipernimi multipli appartenenti a categorie disgiunte nell’ontologia, che quindi generano una logica incoerenza che non permette la rilevazione della polisemia senza la presenza di un esplicito assioma nei livelli più alti (es. “Legge” è vista sia come “Documento_Legale” che come “Regola”, che sono elementi appartenenti a categorie ontologiche opposte e quindi incompatibili).

2.5 Alternative a WordNet

Nei capitoli successivi ci concentriamo soprattutto sull’uso di ontologie esterne formali, come alternativa al database lessicale di WordNet, e come mezzo per sopperire alle problematiche da esso introdotte.

Come detto in precedenza le ontologie definizionali più importanti si classificano tra le tecniche di matching come tecniche di Livello Struttura e di tipo Esterne: contestualizzare un’ontologia diventa quindi allinearla ad una comune di livello superiore usata come fonte di conoscenza esterna.

Per ottenere questo tipo di allineamento, in generale è necessario prima trasformare manualmente le entità del dominio considerato in ontologie leggere, espresse riferendosi alla struttura di quella superiore considerata, e da qui cercare di ricavare i possibili legami tra tutte le entità delle ontologie esaminate. Il procedimento si divide quindi in due fasi: per prima cosa si ha l’*ancoraggio*, ovvero si ha il matching tra l’ontologia fondazionale e le ontologie particolari considerate, quindi si passa alla *derivazione delle relazioni*, ovvero si confrontano tra loro le ontologie particolari e se ne ricavano tutti i possibili legami.

Tra le più importanti ontologie definizionali vi sono Cyc Upper Ontology, Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) e Suggested Upper Merged Ontology (SUMO).

Capitolo 3

3. L'ontologia CYC

Fin dall'inizio lo scopo primario di Cyc [**9**] è stato quello di costruire una larga base di conoscenza, formata da un insieme di nozioni formalizzate utilizzabili per una varietà di ragionamenti e risoluzioni di problemi in diversi contesti. Questo perché si è visto come sistemi contenenti solo conoscenze su domini specifici hanno si portato a grandi risultati, ma sono apparsi difficili da estendere a questioni o tematiche sconosciute, in particolare per le aree riguardanti il linguaggio naturale in cui l'ampiezza del problema è spesso difficile o impossibile da definire in anticipo.

Il progetto di Cyc, iniziato nel 1984 da Douglas Lenat e sviluppato dalla compagnia Cycorp, ha speso gli ultimi vent'anni nella costruzione di un'ontologia comprensiva e di una base per la conoscenza formata dalle informazioni di senso comune, con lo scopo di permettere alle applicazioni di eseguire le loro operazioni come se si utilizzasse il ragionamento umano.

Questa base di conoscenza o Knowledge Base (KB), è una tecnologia già matura ma allo stesso tempo in continuo sviluppo: la sua versione completa contiene più di 2.2 milioni di asserzioni (fatti e regole) e descrive più di 250.000 termini, inclusi circa 15.000 predicati.

Essa oltre a contenere le nozioni legate alla vita di tutti i giorni, comprende anche domini altamente specializzati legati alla chimica, alla biologia, alle malattie e all'organizzazione militare, grazie alla mappatura di diverse ontologie specifiche (SENSUS, FIPS 10-4, WordNet, MeSH, ecc.).

3.1 OpenCyc

Il nome Cyc è un marchio commerciale registrato di proprietà della Cycorp ed è quindi privato, ma una versione ridotta della Knowledge Base è stata resa disponibile al pubblico per poter stabilire un vocabolario comune per il ragionamento automatico: OpenCyc.

La versione corrente di OpenCyc contiene un'ontologia di 47.000 termini, che sono stati definiti ed elaborati usando 306.000 asserzioni, per la maggior parte di tipo tassonomico (non le complesse regole utilizzate in Cyc).

In OpenCyc è incluso anche il *Knowledge Server* eseguibile, contente il motore deduttivo e altri strumenti per accedere, utilizzare ed estendere il contenuto della Knowledge Base (documenti e altro materiale a riguardo sono disponibili al sito http://www.opencyc.org).

Il motore deduttivo [**10**] è un programma che cerca di trovare delle risposte ai problemi posti, ricavandole dalla base di conoscenze, e costruendo quindi l'abilità di ragionamento di OpenCyc: esso utilizza centinaia di moduli, specifici e generali, per la ricerca basata sulla risoluzione dei quesiti, in modo da derivare nuove conclusioni o introdurre nuove ipotesi, nelle asserzioni costituenti la Knowledge Base.

È un programma multi-thread, ovvero è capace di lavorare su più compiti contemporaneamente, e di immagazzinare i risultati parziali in strutture dati dette *problem stores* in modo da permettere al motore di poter interrompere e riprendere i problemi in qualsiasi momento senza dover ricominciare le operazioni dall'inizio.

Il motore deduttivo è capace di derivare nuove conclusioni ragionando sia deduttivamente che induttivamente, e dà una completa spiegazione delle sue risposte, includendo perfino il nome delle risorse da cui sono state ottenute le informazioni. Inoltre, avvisa l'utente se vi sono pro o contro per una particolare conclusione, in modo da modificarla secondo le circostanze o i cambiamenti del contesto. L'utente infine può gestire dozzine di parametri in modo da perfezionare il più possibile la ricerca da eseguire.

3.2 Il linguaggio: CycL

Anche se è una piccola parte del contenuto di Cyc, OpenCyc fornisce due potenziali risorse per gli interessati allo sviluppo delle ontologie e delle basi di conoscenze: la prima è composta

dai concetti fondamentali, i quali possono essere immediatamente utilizzati ed ampliati; la seconda è CycL, un linguaggio espressivo che supporta l'ontologia di OpenCyc.

La sintassi di CycL, esprime elementi di logica di primo ordine [**9**], come il meccanismo di quotazione che permette la differenziazione tra la conoscenza legata a un concetto e la conoscenza legata ai termini che lo definiscono, ma anche elementi di logica di livelli più alti, come la quantificazione dei predicati, delle funzioni o degli assiomi che legano i termini tra loro.

CycL è straordinariamente efficace poiché permette la reificazione e la riflessione, ovvero fornisce la possibilità di fare dichiarazioni di altre dichiarazioni o delle asserzioni sul processo per la creazione dell'asserzione stessa; si basa inoltre, su modelli, cioè è possibile discutere anche di desideri, aspettative e non di sole cose concrete, e consente un ragionamento di default, ovvero distingue i pro e contro di un'affermazione e li confronta; comprende inoltre operatori per la negazione, la congiunzione, la disgiunzione e tante altre operazioni. Ciò ha permesso di rappresentare la semantica della terminologia usata, con lo stesso linguaggio utilizzato per esprimere tutto il resto della conoscenza: in questo modo tutta la KB ha una comune base di rappresentazione ed è mutualmente accessibile per la ricerca deduttiva.

OpenCyc, essendo una proiezione di Cyc, descrive sia la sua tassonomia che la sua ontologia di definizioni. La conoscenza tassonomica è quella delle classi di termini che la compongono e delle relazioni tra di esse, mentre la conoscenza delle definizioni è principalmente quella riguardante il significato inteso di predicati e funzioni, anche questi dipendenti tra loro.

I legami tra i predicati servono soprattutto per mantenere la correttezza semantica: essa si basa sul già citato senso comune, in modo da implementare vincoli che convalidino la conoscenza ed eliminino le relazioni prive di significato.

L'ontologia contiene un'ampia varietà di categorie e domini, che sono identificati attraverso nomi: i nomi dei concetti sono delle *Costanti* (*#$nome*), le quali devono essere definite sia per gli *individui* che per le *collezioni*, ovvero le istanze semplici e le classi, ma anche per le *funzioni*, che producono nuovi termini da quelli dati come input, e le *funzioni-verità*, che possono essere applicate a uno o più concetti e ritornano il valore vero o falso.

I principali predicati [**11**] che legano i vari concetti sono *#$isa* e *#$genIs*, il primo indica che un concetto è un'istanza di una Collezione mentre il secondo che una Collezione è una sotto-collezione di un'altra. Altri esempi possono essere *#$disjointWith* che indica che due Collezioni non hanno nessun membro in comune, o *#$comment*, che è il predicato più frequente nella documentazione.

Grazie al linguaggio CycL, tutta la KB risulta definita e divisa in *micro-teorie*, ovvero collezioni di concetti e legami, tipicamente inerenti allo stesso argomento, che devono essere privi di contraddizioni; ciascuna micro-teoria è individuata dalla propria costante-nome (*#$nomeMt*).

3.3 La struttura

La struttura della Knowledge Base di Cyc è tradizionalmente suddivisa in tre ontologie, in base al livello di generalità delle informazioni che contengono: superiore, di mezzo e inferiore (Figura 3).

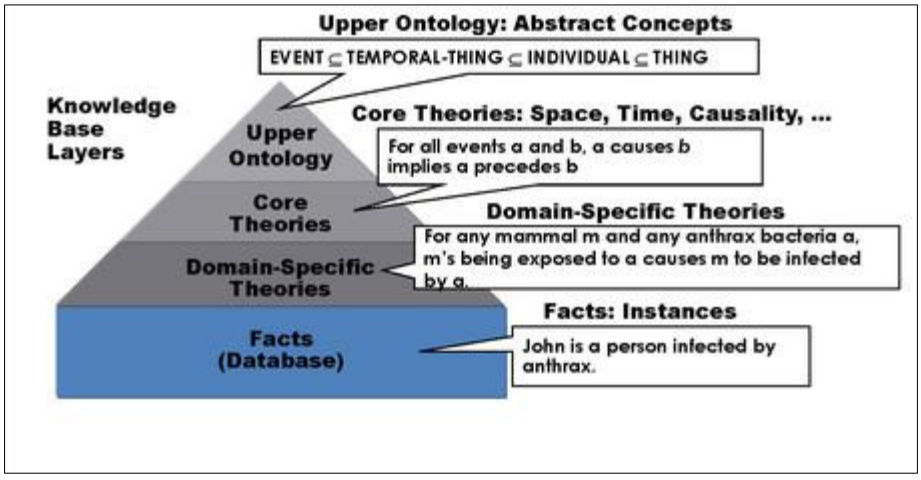

Figura 3: Struttura dell'ontologia di Cyc

La prima si limita ai concetti più ampi, astratti e altamente strutturali, come la temporalità, la matematica o i tipi di relazioni; essa è la parte più piccola delle ontologie ma è allo stesso tempo quella maggiormente referenziata. L'ontologia di mezzo invece contiene un livello di astrazione che è ampiamente usato ma che non è universale in tutti campi della conoscenza, ad esempio contiene relazioni geospaziali, interazioni umane o eventi ed elementi di tutti i giorni. Infine, l'ultima comprende tutti i domini specifici della conoscenza (il cosiddetto

livello-foglia), in cui si approfondiscono particolari campi di studio, come la chimica o una particolare nazione o persona; questa è la sezione più ampia ma allo stesso tempo quella con il minor numero di relazioni.

3.4 ResearchCyc

Nel 2006 la Cycorp ha rilasciato un'altra versione gratuita di Cyc, indirizzata alla comunità di ricerca, chiamata ResearchCyc. In aggiunta alle informazioni tassonomiche presenti in OpenCyc, ResearchCyc contiene una conoscenza semantica molto significativa riguardante i concetti già presenti in quella di base, un ampio lessico e un'interfaccia Java per l'inserimento e l'analisi delle nozioni.

3.5 Problematiche

Il progetto Cyc è stato descritto come "uno dei lavori più controversi nella storia delle intelligenze artificiali" [**12**], infatti è stato criticato sotto diversi aspetti:

- la complessità del sistema, data dalle sue ambizioni enciclopediche, e la conseguente difficoltà nel fare delle aggiunte;
- il trattamento insoddisfacente del concetto di sostanza e della relativa distinzione tra proprietà intrinseche ed estrinseche;
- la mancanza di possibilità di confronto dell'efficienza del motore deduttivo di Cyc;
- la corrente incompletezza del sistema sia in ampiezza che in profondità, e la relativa difficoltà nel misurarne il grado;
- la documentazione limitata;
- la mancanza di insegnamenti e spiegazioni on-line che rende difficile, per utenti inesperti, utilizzare il sistema;
- un grande numero di lacune, non solo nell'ontologia di oggetti ordinari ma soprattutto nelle asserzioni che li descrivono.

3.6 Applicazione di Cyc nella disambiguazione del testo

Il problema della disambiguazione del senso delle parole, ovvero la questione di come determinare quale significato attribuire all'occorrenza di una parola in un particolare contesto,

è stato a lungo considerato uno degli aspetti più difficili ma promettenti della ricerca in campo linguistico.

Tutti i risultati ottenuti in questo campo sono diventati dei grandi passi avanti per le applicazioni basate sul linguaggio naturale, Natural Language Processing (NLP) Applications, dal filtraggio delle e-mail al recupero e classificazione dei documenti, o dall'annotazione semantica dei testi fino alla risoluzione dei problemi.

Il lavoro nell'area della disambiguazione testuale porta l'attenzione sulla definizione ed implementazione di algoritmi, i quali determinano l'affinità semantica relativa ai sensi delle parole che si trovano nello stesso contesto. Questi algoritmi comprendono o combinano generalmente due tipi di approcci: il primo è l'approccio lessicale, esso identifica il senso delle parole per mezzo di stringhe, disambiguando parti della parola stessa o estraendo stringhe presenti nelle definizioni dei suoi significati; il secondo è l'approccio tassonomico, che identifica il senso delle parole grazie ai nodi presenti in una gerarchia di sensi e usa le relazioni tra di essi per misurare l'affinità semantica dei termini.

Con la sempre maggiore disponibilità di dizionari on-line e thesaurus, la ricerca nell'area della disambiguazione delle parole si è mossa verso la rifinitura ed estensione di approcci sia lessicali che tassonomici basati sull'affinità semantica: ad esempio WordNet ne è diventato uno strumento standard, essendo una combinazione dei due approcci, data dall'unione di un'ontologia di nomi con una serie di synset formati da sinonimi e glosse per le definizioni.

Relativamente poco, invece, si è fatto nel campo dell'applicazione di un'ontologia completa al problema della disambiguazione: il beneficio che si otterrebbe utilizzandola, include la disponibilità di un più ricco insieme di relazioni tra i significati dei termini, la possibilità di sviluppare un più robusto meccanismo per determinare l'affinità semantica e l'opportunità di rappresentare tutto o una parte del contenuto del testo nel formalismo usato per esprimere l'ontologia.

Ad esempio con l'applicazione dell'ontologia di Cyc al problema della disambiguazione del testo [**13**], si trova sia un metodo per la determinazione dell'affinità semantica, costruito utilizzando anche le complesse relazioni che reggono l'ontologia, sia un metodo per verificare

se i significati delle parole danno anche un contributo semantico all'argomento trattato nel testo.

La rappresentazione del senso delle parole in Cyc, si basa sulla sua Knowledge Base ed è principalmente dichiarativa, facendo uso di un vocabolario per la rappresentazione delle parole, delle loro proprietà e per la mappatura dei predicati. Il predicato *#$denotation* è una delle relazioni più usate nella rappresentazione dei concetti legati al linguaggio naturale.

L'ambiguità semantica è riconosciuta dal sistema quando un termine che deve essere inserito nel lessico di Cyc, ha una mappatura condivisa da più voci.

Diversamente dalle tassonomie-dizionario come WordNet, Cyc non è un'ontologia di significati, infatti, non tutti i nodi che la compongono possono essere definiti come il senso di un termine: i concetti delineati in Cyc sono quelli necessari alla rappresentazione del ragionamento di senso comune, e quindi non si può costruire una mappa comprendente tutta l'ontologia perché difficilmente "traducibile" con le parole del linguaggio di tutti i giorni.

Allo stesso tempo però ci sono anche molti significati che non sono presenti nell'ontologia, conseguenza del fatto che si vuole rappresentare solo la realtà comune.

Cyc è stata utilizzata per facilitare la disambiguazione del testo in diversi sistemi, sia per la ricerca sul Web che per la realizzazione di strutture interattive. La sua applicazione si deve principalmente alla sua tassonomia di conoscenze, usata in modo da generare stringhe disambiguate che servono per chiarire le domande inserite dagli utenti.

3.6.1 TextLearner

TextLearner [**13**] è un'applicazione basata su Cyc, costruita per acquisire la conoscenza dalla lettura di un testo; al contrario delle precedenti applicazioni essa massimizza l'importanza del contesto, costruendo un'esplicita e formale rappresentazione della struttura del documento letto.

I documenti esaminati sono espressi nella Knowledge Base come una serie innestata di strutture linguistiche chiamate CIS (Contextualized Information Structures), le quali analizzano il testo scomponendolo sia in paragrafi che in singole frasi, e danno la possibilità

al sistema di ragionare sulle ricorrenze di una particolare parola in relazione alle occorrenze di un altro termine, frase o paragrafo.

L'approccio di TextLearner al problema della disambiguazione è di generare tutte le possibili interpretazioni di un termine e relazionarle direttamente alle CIS come ipotesi alternative. Queste ipotesi sono poi considerate come delle micro-teorie, al cui contenuto non possono accedere le une con le altre. Ciò permette a Cyc di "ragionare" sulle ipotesi, esplorando le loro conseguenze semantiche e come si relazionano con le altre ipotesi, senza generare delle contraddizioni.

Il trattamento delle interpretazioni semantiche come ipotesi consente al sistema di segnalare la risoluzione dell'ambiguità solo quando tutto è definito; inoltre il motore deduttivo di Cyc può interrogare la sua Knowledge Base per vedere quale valutazione generalmente il sistema preferisce, aggiornando le preferenze man mano che le scelte vengono fatte.

Abbiamo visto come i metodi per determinare l'affinità semantica che si basano su sistemi formalizzati, generalmente delle tassonomie, non siano niente di nuovo: TextLearner si differenzia perché sopporta tutto il peso di un'intera ontologia; ovvero si ha l'utilizzo delle proprietà dei livelli più alti, inesistenti in una semplice tassonomia, ma che sono facilmente esprimibili nel linguaggio di Cyc, CycL.

Infatti CycL, contiene il proprio vocabolario per l'affinità semantica: esso utilizza le relazioni *#$nearestIsa* o *#$nearestIsaOfType* per determinare la vicinanza dei significati dei vocaboli, distinguendo se questi sono istanze, collezioni o tipi di altri termini, in modo da individuare la relazione che li lega e con quale intensità. In aggiunta a queste relazioni, vi sono i legami tra le coppie con affinità semantica diretta, ovvero legate dai predicati *#$isa* e *#$genIs*.

L'applicazione di questa metrica per l'affinità semantica avviene su tre livelli, documento, paragrafo e frase, che si diversificano per un peso addizionale che è aggiunto ai valori rappresentanti le distanze-relazioni tra le parole.

Più complesso è il meccanismo per la determinazione del contributo semantico, il quale è stato realizzato a parte in TextLearner.

Durante la creazione del modello della struttura del documento, l'applicazione tiene traccia delle singole frasi del testo e memorizza le relazioni che hanno con i termini che le compongono. Queste frasi sono poi inviate a dei moduli che trasformano il linguaggio naturale in linguaggio CycL (NL-to-CycL), i quali generano tutte le loro possibili semantiche in modo formale. Se il senso di una parola contribuisce ad una delle interpretazioni in CycL della frase, allora ciò va a favore di quel significato nella successiva disambiguazione, altrimenti si ha il contrario.

In alcuni casi quindi TextLearner riesce a misurare il contributo semantico che hanno i significati delle parole nelle frasi, anche se ciò comporta una notevole difficoltà nel dover prima mappare la lingua naturale (inglese) con un linguaggio formale.

Ciascuna frase in CycL, prodotta dai moduli NL-to-CycL, è considerata come un'ipotesi di ciò che la frase in input vuole dire; quindi una successione di regole interroga queste ipotesi, confrontandole con delle proposizioni ricavate con il motore deduttivo di Cyc. Se una di queste proposizioni viene trovata e se, essa, è anche uno dei sensi candidati per una parola o una frase all'interno della stessa parte di testo, ciascun termine di CycL che la compone è un argomento considerato come un contributo semantico alla parziale comprensione da parte di Cyc della frase iniziale.

Capitolo 4

4. L'ontologia SUMO

Il SUO Working Group [**14**], un gruppo di lavoro sanzionato dall'IEEE composto da collaboratori appartenenti ai campi dell'ingegneria, della filosofia e delle scienze dell'informazione, ha riconosciuto la necessità di creare un'ontologia ampia e completa e allo stesso tempo gratuita, pubblica e standard: è nata quindi come documento di partenza, SUMO (Suggested Upper Merged Ontology), un'ontologia di alto livello che consente di definire termini generali e può essere usata come base per ontologie con domini più specifici.

SUMO può essere utilizzata come schema di base per le nuove conoscenze, in modo da facilitarne l'inserimento nei diversi sistemi, o come mezzo per l'integrazione di diversi database, i quali possono essere mappati ad un'ontologia comune, o infine come completamento di ontologie di dominio specifico, che in questo modo possono condividere un maggior numero di termini e definizioni.

4.1 La realizzazione

SUMO è stata creata dalla combinazione dei contenuti di diverse ontologie accessibili pubblicamente, all'interno di un'unica struttura, comprensiva e coesa: essa contiene l'Ontolingua server, John Sowa's upper level ontology, le ontologie sviluppate dall'ITBM-CNR e tante altre, ed il suo contenuto può essere visualizzato e scaricato dal sito http://ontology.teknowledge.com.

Il linguaggio utilizzato per la sua realizzazione è una versione di KIF (Knowledge Interchange Format) chiamata SUO-KIF.

La procedura seguita per costruire SUMO vede come primo passo l'identificazione dei contenuti per i livelli più alti dell'ontologia, il loro collegamento al sito del gruppo di lavoro e la loro traduzione nel linguaggio utilizzato.

Quindi si ha il passaggio alla fase più complessa della combinazione semantica, ovvero l'integrazione dei contenuti delle varie ontologie all'interno dello stesso lavoro, comprensivo e consistente.

Innanzitutto si sono divise le ontologie in due classi, quelle contenti le definizioni dei concetti di alto livello e quelle contenenti le nozioni di livello più basso, quindi si sono create due ontologie che poi sono state fuse tra loro. Nella prima classe non si sono incontrate molte difficoltà essendo i concetti spesso ripetuti e sovrapposti, mentre nell'allineamento dei contenuti di basso livello si sono incontrate quattro tipologie di allineamento. Nel primo caso di allineamento si ha che nessuna delle estremità dell'ontologia contiene una corrispondenza con il concetto/assioma che deve essere inserito, il quale è però utile e non in contraddizione con alcun principio filosofico; il problema quindi si riduce alla ricerca del posto in cui inserirlo, creando eventualmente livelli intermedi tra i concetti.

Nel secondo il concetto/assioma da inserire è giudicato fuori luogo in una struttura che si prefigge l'essere di ampia applicazione e accettazione, ed è quindi rimosso; anche se il giudizio è in qualche modo soggettivo ciò non diminuisce l'importanza della costruzione oggettiva dell'ontologia.

Nel terzo caso di allineamento si ha una perfetta sovrapposizione tra l'elemento dell'ontologia e il concetto/assioma da mappare, ovvero si ha equivalenza tra il nuovo assioma e quello esistente; quindi anche se i termini differiscono, il concetto rimane invariato e tutte le definizioni sono interscambiabili.

Infine, l'ultimo caso è quello più complesso, ovvero quando si ha solo una parziale sovrapposizione tra il nuovo contenuto e quello già esistente nell'ontologia: spesso esso non può essere tradotto con un concetto più generale o più specifico e quindi è necessario riformulare le definizioni dei concetti in modo da eliminare le incompatibilità o fare una revisione generale della teoria applicata.

4.2 La struttura

Il miglior modo di mostrare la struttura e il contenuto di SUMO è di mostrare sistematicamente i concetti dei livelli più alti e le relazioni tra di essi: la radice dell'ontologia è "Entità" e questa si divide nei concetti "Fisico" ed "Astratto", il primo contiene tutto ciò che ha posizione nello spazio e nel tempo, mentre l'altro tutto il resto. La Figura 4 mostra i vari concetti e l'indentazione indica la relazione di sottocategoria.

```
Physical
    Object
        SelfConnectedObject
            ContinuousObject
            CorpuscularObject
        Collection
    Process
Abstract
    SetClass
            Relation
    Proposition
    Quantity
        Number
        PhysicalQuantity
    Attribute
```

Figura 4: I livelli più alti di SUMO

Il concetto "Fisico" si suddivide in "Oggetto" e "Processo"; infatti per l'ontologia di SUMO, dopo uno dei principali dibattiti, si è scelto di sviluppare l'orientamento 3D, che considera gli oggetti come qualcosa di completamente presente in ogni momento della loro esistenza (al contrario dell'orientamento 4D che invece sostiene come tutto sia un continuo evolversi nel tempo).

In questa analisi non ci dilunghiamo nella descrizione della seconda categoria, la quale è stata uno dei punti più problematici nella creazione di SUMO: all'interno dell'ontologia si è semplicemente integrato il centro Process Specification Language (PSL), la cui maggior parte dei verbi sono stati fatti rientrare in questa categoria, eliminando quelli statici o che non si riferivano in qualche modo ad un processo e sistemandoli in una gerarchia, sviluppando gli assiomi formali tra i vari concetti.

Immediatamente sotto al concetto di “Oggetto”, vi sono quelli di “Oggetto Connesso” e “Collezione”: il primo include tutti gli oggetti le cui parti sono immediatamente legate le une alle altre, mentre il secondo è composto da parti disconnesse la cui relazione è quella di *membro*, che indica la posizione spazio-temporale della collezione e l’invarianza della sua identità rispetto ad un’eventuale aggiunta o eliminazione di uno dei suoi membri.

A sua volta “Oggetto Connesso” si divide in “Oggetto Continuo” e “Oggetto Corpuscolare”, il primo si ha quando le sue parti hanno le stesse caratteristiche e anche le stesse dell’intero oggetto, mentre il secondo se avviene il contrario.

Ritornando alla prima suddivisione dell’ontologia, consideriamo il ramo “Astratto”: esso si fraziona in quattro categorie “Insieme”, “Proposizione”, “Attributo” e “Quantità”.

La prima si sviluppa nelle categorie “Classe”, ovvero un set di elementi aventi le proprietà necessarie per farne parte, e successivamente in “Relazione”, che è una classe di tuple ordinate. La seconda invece corrisponde alla nozione di contenuto semantico o informativo e comprende tutto ciò che indica una proposizione, sia essa espressa da una frase o da interi libri. La terza include tutte le qualità, proprietà, ecc. che non sono classificate come oggetti. Infine, la quarta si divide in “Numero” e “Quantità Fisica”, la prima rappresenta il valore di un qualsiasi sistema di misurazione mentre l’altra indica una cifra accompagnata da un’unità di misura.

Durante la costruzione di questa ontologia sono stati incontrati alcuni problemi, come la scelta dell'orientamento 3D o 4D, o come l’inconsistenza teorica di alcuni elementi rilevanti nel campo dell’ingegneria; quest’ultimi sono stati raccolti in packages indipendenti e, dove possibile, collegati all’ontologia generale.

4.3 Vantaggi

SUMO è stata costruita anche se erano già esistenti altre ontologie di alto livello, ad esempio Cyc Ontology, la quale è però di difficile utilizzo come standard ed è solo in parte resa disponibile al pubblico. Inoltre SUMO ha il vantaggio di essere parte di un progetto sponsorizzato dall’IEEE (ovvero gli utenti possono confidare sul fatto che essa sarà utilizzata da un gran numero di persone), e di essere basato solo su principi pratici: tutti i legami

strettamente filosofici sono stati eliminati e ciò ha reso l'ontologia di facile utilizzo e comprensione.

4.4 Mapping fra SUMO e WordNet

Visto il continuo sviluppo delle ontologie e la necessità dell'utilizzo di esse nelle diverse applicazioni, un importante utilizzo di SUMO è il mapping rispetto al database lessicale, WordNet [**15**]. In questo modo le applicazioni utilizzanti WordNet, ottengono maggiore consistenza grazie all'ontologia, i vari files di interesse possono essere estratti separatamente gli uni dagli altri in modo da semplificare le varie operazioni, e chi vuole creare un'ontologia specifica collegata a SUMO ha già anche i riferimenti al database lessicale.

Per prima cosa si sono scelte le relazioni da considerare nel mapping dei synset di WordNet con i concetti di SUMO: la sinonimia, quando si ha la coincidenza del significato del termine nei due ambienti (=), l'ipernimia, quando il concetto ha più ampio significato del termine del synset che vi deve essere mappato (+), l'istanziazione, quando il termine del synset è un particolare membro del concetto dell'ontologia (@).

Nei primi livelli del mapping non sono stati riscontrati dei grossi problemi, essendo le nozioni di WordNet e SUMO equivalenti. Alcune difficoltà sono state affrontate in seguito, ad esempio nell'inserimento del synset di "Spazio", il quale concetto non è presente nell'ontologia: si è scelto quindi di operare come con la nozione parallela di "Tempo", il quale è stato inserito nella categoria di "Misura del Tempo", comprendendolo in "Misura di Lunghezza", che cattura l'aspetto quantitativo del concetto di spazio. Un altro problema si è riscontrato nell'inserimento di synset che hanno un irriducibile aspetto soggettivo, come i termini "Migliore", "Difficile", ecc.: per questi si è deciso di creare un'apposita categoria chiamata "Attributo di Valutazione Soggettiva". Infine, un altro caso particolare si è verificato quando un synset poteva essere mappato in più concetti o viceversa: spesso si sono quindi inseriti dei legami di sinonimia tra più termini oppure di specificazione a più concetti combinati tra loro.

Ciò ha permesso la creazione di un ponte tra i concetti strutturati e il testo "libero", utilizzato da un crescente numero di applicazioni: si è realizzato anche un tool il quale permette di inserire termini inglesi ed ottenere come risultato la definizione del concetto presente

nell'ontologia, grazie al legame con i synset di WordNet. Le applicazioni NLP (utilizzanti il testo "libero") hanno inoltre la possibilità di assegnare ad esso un significato strutturato, e di operare perfino nel campo della disambiguazione del significato, utilizzando algoritmi che ricavano il concetto contenuto in SUMO dal contesto in cui è utilizzato il termine, o in altri casi di essere facilitate nella creazione automatica di sommari e nella ricerca semantica.

Il mapping rispetto a WordNet è stato anche un ulteriore controllo della completezza di SUMO, ad esempio sono state aggiunte altre categorie laddove i concetti erano rimasti troppo generici, dando la possibilità all'ontologia di essere utilizzata per esprimere qualsiasi cosa si voglia, in un contesto formale.

Capitolo 5

5. L'ontologia DOLCE

DOLCE (Descriptive Ontology for Linguistic and Cognitive Engineering), è il primo modulo della Library of Foundational Ontologies, una libreria di ontologie definizionali, che è stato sviluppato all'interno del progetto Wonder Web [**16**].

Essa non è da intendere come un candidato per un'ontologia standard universale, ma come punto di partenza per poterne creare di nuove, per poterle confrontare e integrare tra loro, per chiarire le varie relazioni tra i futuri moduli da aggiungere e per sottolineare i legami tra le ontologie esistenti e le risorse linguistiche come WordNet.

DOLCE è un insieme di termini e di relazioni con un chiaro orientamento *cognitivo*, che cerca di catturare le categorie legate al linguaggio naturale e al senso comune costruendo delle 'scatole cognitive' in grado di cogliere diverse rappresentazioni della realtà, legate alla percezione umana, allo sfondo culturale e alle convenzioni sociali. Questo approccio fa si che le categorie siano inserite nel cosiddetto livello *mesoscopico*, ovvero sono considerate come delle nozioni descrittive che esplicitano legami ed entità già noti, riflettendo più o meno la superficie della conoscenza e del linguaggio.

È un'ontologia di *particolari*, nel senso che il dominio del discorso è ristretto ad essi; infatti sono considerate come universali le relazioni e le proprietà, le quali compaiono nell'ontologia ma solo per caratterizzare e organizzare i particolari. La differenza formale tra essi sta nella possibilità di avere istanze, gli universali le possono avere mentre i particolari no.

Un'altra particolarità di DOLCE è il suo *approccio moltiplicativo*, ovvero entità distinte possono essere co-localizzate nella stessa regione spazio-temporale: esse sono entità costituite dalla stessa materia ma che in momenti diversi possiedono delle proprietà differenti ed incompatibili (es. vaso-creta).

5.1 La struttura

DOLCE si basa sulla fondamentale distinzione tra entità *enduring* (continuante) e *perduring* (occorrente), che differiscono per il loro comportamento nel tempo: gli *endurant* sono interamente presenti (tutte le loro parti costituenti sono presenti) in ogni istante della loro presenza, mentre i *perdurant* sono presenti nel tempo grazie ad un successivo accumulo di diverse loro parti costituenti, ovvero in ogni istante della loro presenza sono solo parzialmente presenti (es. foglio-testo che si sta leggendo).

Un'altra distinzione tra essi è la possibilità di cambiamento; gli endurant sono nel tempo, e possono subire cambiamenti, mentre i perdurant occorrono nel tempo e quindi non possono mutare, pena la perdita della loro identità.

La relazione principale che si stabilisce tra essi è quella di *partecipazione*: un endurant vive nel tempo partecipando in alcuni perdurant (o occorrenze).

In DOLCE è presente, inoltre, la distinzione tra la qualità e il valore (*quale*) che essa assume, esso descrive la particolare posizione all'interno del suo spazio concettuale (*quality region*), il quale rappresenta la nostra percezione della struttura di ciascun tipo di qualità.

Le qualità in DOLCE sono considerate come le principali entità che misuriamo e percepiamo (forma, colore, dimensione, suono, odore, ecc.) e sono ben distinte dal concetto di proprietà, che è considerato come universale.

Le qualità sono legate alle entità, esse infatti sono costantemente dipendenti dall'entità a cui fanno riferimento: se viene a mancare l'entità anche la qualità cessa di esistere. Due entità non possono avere la stessa qualità ma possono avere lo stesso valore di qualità.
Anche spazio e tempo sono considerate come qualità, con i rispettivi spazi concettuali. Un'altra proprietà delle qualità è l'essere dirette o indirette, infatti quest'ultime derivano dalle

qualità spazio-temporali dei partecipanti all'entità stessa. Infine possiamo osservare che nessuna relazione di appartenenza è definibile per le qualità nell'ontologia di DOLCE.

L'ultima categoria in cui l'ontologia si suddivide sono le entità *astratte*: esse non hanno qualità spazio-temporali e non sono considerate loro stesse delle qualità; la classe astratta che consideriamo in questa versione è quella degli spazi concettuali (quality region).

Le relazioni presenti all'interno di DOLCE devono essere innanzitutto più generali possibili in modo da essere applicate in diversi contesti, intuitive e ben studiate. Le principali sono:

- *Appartenenza*, ovvero se "X è parte di Y" ed entrambe sono dei perdurant;
- *Appartenenza Temporanea*, ovvero se "X è parte di Y durante t" ed entrambe sono endurant per i quali è necessario conoscere in quale momento la relazione avviene;
- *Dipendenza*, ovvero "X dipende da Y se, necessariamente, Y è presente ogni qualvolta lo è X";
- *Dipendenza Spaziale*, è una specializzazione della dipendenza, in cui X e Y devono essere oltre che compresenti anche co-localizzati;
- *Costituzione*, ovvero se "X costituisce Y durante t" e quindi alcune proprietà che sono accidentali per X, sono essenziali per Y;
- *Partecipazione*, ovvero se "X partecipa in Y durante t" e quindi si hanno endurant coinvolti in un'occorrenza;
- *Attribuzione di qualità*, legame tra la qualità e l'entità a cui si riferisce;
- *Attribuzione di valore*, legame tra la qualità e il suo valore assunto, il quale può variare nel tempo o rimanere costante.

Le categorie base si dividono ulteriormente in altre sottocategorie. Gli endurant si dividono in *Physical* e *Non-Physical*, i primi hanno qualità spaziali dirette e si scompongono in *Amounts of matter*, *Objects* e *Features* a seconda del loro tipo di unità; gli altri invece si dividono prima in *Mental* e *Social Objects*, e successivamente in *Agentive* e *Non-Agentive* a seconda della loro intenzionalità.

I perdurant invece comprendono tutto ciò che è chiamato evento, processo, attività o stato, e si dividono secondo le nozioni di *homeomericity* (un'occorrenza lo è se e solo se tutte le sue parti temporali possono essere descritte nello stesso modo per tutta la durata dell'occorrenza)

e di *cumulativity* (se un'occorrenza si basa sulla somma di più istanze). Infine, le qualità si dividono a seconda del tipo di entità a cui fanno direttamente riferimento.

Nella Figura 5.1 è mostrata la tassonomia base, le cui categorie sono considerate come rigide proprietà, mentre in Figura 5.2 si possono vedere esempi delle categorie basilari poste all'estremità inferiori della gerarchia.

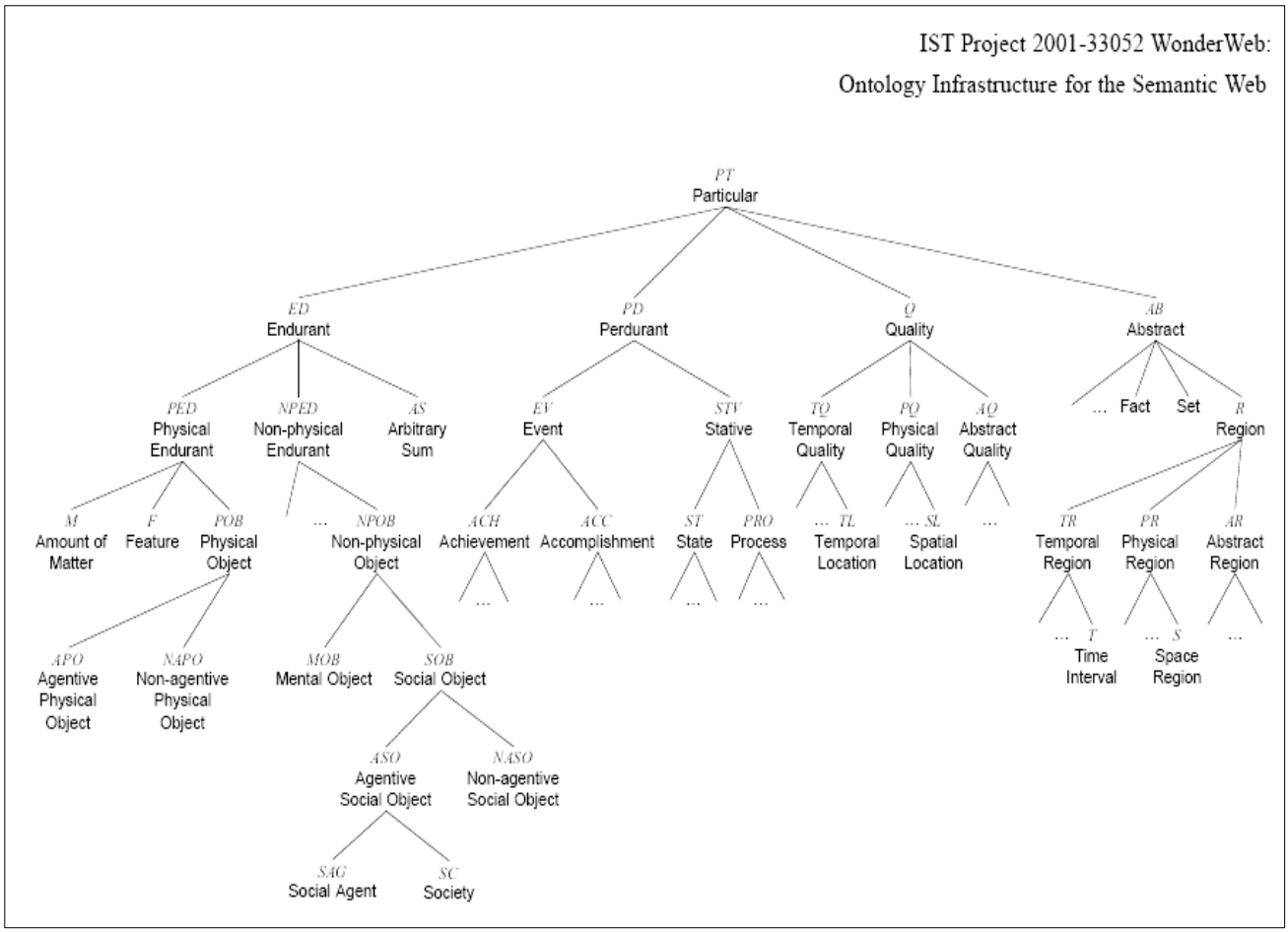

Figura 5.1: La tassonomia di DOLCE

"Leaf" Basic Category	Examples
Abstract Quality	*the value of an asset*
Abstract Region	*the conventional value of 1 Euro*
Accomplishment	*a conference, an ascent, a performance*
Achievement	*reaching the summit of K2, a departure, a death*
Agentive Physical Object	*a human person (as opposed to legal person)*
Amount of Matter	*some air, some gold, some cement*
Arbitrary Sum	*my left foot and my car*
Feature	*a hole, a gulf, an opening, a boundary*
Mental Object	*a percept, a sense datum*
Non-agentive Physical Object	*a hammer, a house, a computer, a human body*
Non-agentive Social Object	*a law, an economic system, a currency, an asset*
Physical Quality	*the weight of a pen, the color of an apple*
Physical Region	*the physical space, an area in the color spectrum, 80Kg*
Process	*running, writing*
Social Agent	*a (legal) person, a contractant*
Society	*Fiat, Apple, the Bank of Italy*
State	*being sitting, being open, being happy, being red*
Temporal Quality	*the duration of World War I, the starting time of the 2000 Olympics*
Temporal Region	*the time axis, 22 june 2002, one second*

Figura 5.2: Categorie all'estremità della tassonomia di DOLCE

5.2 DOLCE Lite + e DOLCE Ultralite

Le versioni "Lite" o "leggere" sono traduzioni semplificate della versione completa di DOLCE, che vengono utilizzate in numerosi progetti riguardanti le ontologie e che non considerano alcuni aspetti come l'indicizzazione temporale o le relazioni composte [**17**].

Esse includono, invece, molti moduli basati sul DnS (Descriptions & Situations), un componente utilizzato nella contestualizzazione degli elementi, soprattutto legati al trattamento delle entità sociali come organizzazioni, collettività, piani, norme e oggetti informativi.

DOLCE Ultralite [**18**], è un'altra versione, sempre "alleggerita", dell'assiomatizzazione in OWL di DOLCE e del modulo DnS; essa presenta una semplificazione dei nomi di molte

classi e proprietà, un'aggiunta di linee di commento e un allineamento con l'insieme di schemi sui contenuti. A dispetto della sua semplificazione, che aumenta notevolmente la velocità con cui controlla la consistenza e la classificazione dei domini delle ontologie in OWL che vi vengono aggiunte, la sua espressività non è significativamente diversa dalla precedente versione DOLCE Lite +.

Tutte queste versioni sono sviluppate e mantenute da Aldo Gangemi e dal ramo della LOA (Laboratory for Applied Ontology) con sede a Roma.

5.3 Mapping fra DOLCE e WordNet

L'ontologia DOLCE può essere utilizzata per analizzare la struttura ontologica di WordNet: essendo in crescita il numero delle applicazioni che utilizzano WordNet più come un'ontologia che come una risorsa lessicale, risulta molto importante analizzarlo ponendosi come riferimento un'ontologia fondazionale.

L'esaminare WordNet rispetto a DOLCE è stato un primo passo per risolvere le inadeguatezze che WordNet aveva manifestato (vedi paragrafo 2.4.1), si è quindi cercato di identificare le corrispondenze all'interno dell'ontologia stessa, considerando inizialmente solo la tassonomia principale ed escludendo i synset con pochi figli.

Si è provato a portare maggiore chiarezza nella struttura dei termini e dei synset, eliminando il più possibile l'eterogeneità (ad es. in Entità e in Astrazione) e spostando a livelli più bassi quei termini che suonavano troppo specifici, precisando così le tipologie di relazioni.

5.3.1 OntoWordNet

Abbiamo visto che WordNet può essere utilizzato come un'ontologia se alcuni dei suoi legami lessicali sono reinterpretati in accordo con una semantica formale che possa spiegare l'utilizzo di un elemento lessicale in un certo contesto e per un certo scopo, ovvero se si ha "una specificazione formale delle concettualizzazioni che sono espresse dai significati dei synset di WordNet" [**19**].

Questo è l'obiettivo principale del progetto di ricerca chiamato OntoWordNet, iniziato due anni fa nel ISTC-CNR, e adesso esteso con altri partners, e che vede la collaborazione delle università di Priceton, Berlino e Roma.

Il progetto si è prefisso di riorganizzare e arricchire il database lessicale seguendo alcuni precisi obiettivi:

- *Logico*: i synset di WordNet sono trasformati in tipi logici, con una semantica formale per le relazioni lessicali;
- *Ontologico*: WordNet è trasformato in una libreria ontologica generica, con un chiaro criterio che separi le diverse categorie ontologiche (concetti, istanze, relazioni, ecc.);
- *Contestuale*: WordNet è modularizzato in accordo con le conoscenze possedute nei diversi domini di interesse;
- *Semantico*: il lessico di WordNet deve essere ordinato secondo le preferenze, la frequenza e le combinazioni più rilevanti.

Si sono cercati di realizzare questi obiettivi migliorando la relazione di iponimia/ipernimia dividendola in specializzazione/generalizzazione, per i legami tra concetti, e istanziazione, per i legami tra concetti e singole istanze (come era risultato anche dall'analisi di OntoClean).

Inoltre, analizzando la struttura di WordNet, sono stati fissati alcuni principi per convertire la forma di database lessicale in una vera e propria ontologia, come "i synset sono classi di significati equivalenti", "le glosse sono assiomatizzazioni" o "le glosse costituiscono i synset", ecc. da cui si è poi cercato di estrarre la vera e propria stesura formale.

Presi questi come punti di partenza, sono state ricavate un numero di relazioni implicite interne a WordNet, come i legami tra synset e i synset presenti nella sua glossa, che sono poi state semi-automaticamente formalizzate: prima si sono utilizzati algoritmi di disambiguazione del significato per organizzare i termini in modo coerente all'ontologia fondazionale di DOLCE, quindi si sono usati metodi sia top-down che bottom-up per creare le associazioni tra di essi.

Durante il progetto si è visto come WordNet, dopo la sua formalizzazione, potesse essere considerato un'estensione di DOLCE: mentre quest'ultima costituisce la cima della struttura piramidale del dominio dell'ontologia, la libreria risultante dal progetto di OntoWordNet va ad occupare i gradini più bassi della "piramide", in cui i domini di interesse diventano man mano più specifici, lasciando quindi spazio ad un possibile successivo arricchimento (Figura 5.3).

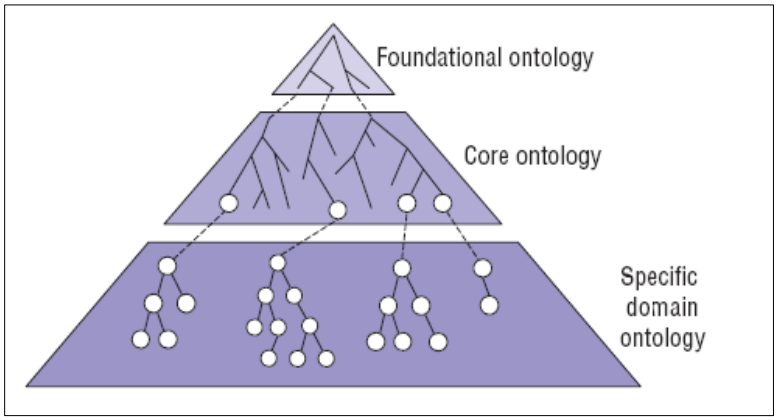

Figura 5.3: I tre livelli di generalità nel dominio dell'ontologia

Essendo però DOLCE molto astratta si sono incontrate delle difficoltà nel trovare da quale categoria fondazionale dell'ontologia far partire i legami con i synset di WordNet: si è preferito quindi utilizzare una versione più leggera dell'ontologia a cui sono stati aggiunti dei plugin per dei domini più specifici, DOLCE-Lite+. Questo ha permesso di "dolcizzare" ben 809 synset, dando risultati più che positivi.

5.4 Applicazione di DOLCE e SUMO: SWIntO

Il progetto Smart Web INTegrated Ontology (SWIntO) [**20**] è un'altra applicazione in cui DOLCE è stata utilizzata per le sue principali caratteristiche di ontologia fondazionale:

- *La chiarezza concettuale*: è un punto di riferimento per un rigoroso confronto tra diversi approcci ontologici possibili ed è una traccia per analizzare, armonizzare ed integrare ontologie esistenti e altri tipi di metadati standard;
- *Fornisce un modello base*: è un punto di partenza per la costruzione di nuove ontologie, contenendo già una serie di entità ontologiche che possono essere riutilizzate nei domini specifici del progetto;

- *Fornisce schemi per il disegno di un'ontologia*: propone degli schemi che spesso ricorrono in tutte le ontologie, come l'ubicazione dello spazio e del tempo, e che devono essere inseriti per dare consistenza all'ontologia stessa.

Il progetto si sviluppa su diversi temi principali come gli eventi sportivi, la navigazione, il discorso, i dati multimediali e l'informazione linguistica, i quali devono essere integrabili tra loro e quindi allineati ad una stessa ontologia fondazionale che permetta loro una chiara organizzazione.

Inoltre, l'utilizzo di un'ontologia fondazionale a base del progetto permette di non perdere i significati dei termini e di risolvere le loro ambiguità, che sono problemi molto frequenti quando si devono integrare domini di interesse così diversi gli uni con gli altri.

Tra le tante ontologie DOLCE è stata scelta perché rispondeva alle caratteristiche richieste, ovvero essere:

- *Descrittiva*: mira a descrivere i legami ontologici che stanno dietro al linguaggio e alla conoscenza, rifacendosi alla struttura superficiale del linguaggio naturale e del senso comune;
- *Moltiplicativa*: mira a dare uno spaccato della realtà permettendo a diverse entità di essere co-localizzate nella stessa regione spazio-tempo;
- *Paradigma 4D (Perdurantism)*: risolve la problematica della variazione delle entità nel tempo e della loro identità assumendo che esse abbiano parti sia spaziali che temporali (e quindi 4 dimensioni).

Un'altra caratteristica è quella del possibilismo (negazione della tesi "tutto ciò che esiste è attuale") che però rende più complessa l'ontologia e quindi più complicato il suo utilizzo nel progetto: proprio per compensare ciò è introdotta anche un'altra ontologia, SUMO (Suggested Upper Merged Ontology) nata dall'aggregazione di diverse ontologie di alto livello e presentante una tassonomia molto ricca, di cui si è precedentemente parlato.

Per prima cosa i ricercatori hanno modificato entrambe le ontologie di base ottenendo due moduli chiamati SmartSUMO e SmartDOLCE, che, integrati tra loro, hanno poi costituito l'ontologia fondamentale per il successivo inserimento delle informazioni riguardanti i domini di interesse più specifici. Quest'ultimi sono stati inseriti creando nuovi tipi di relazione in

modo da permettere il loro allineamento con l'ontologia di base, completando così il progetto SWIntO.

In particolare da DOLCE sono stati presi due importanti moduli, l'OIO (Ontology of Information Objects) e il DnS (Descriptions & Situations): il primo contiene un modello per stabilire le relazioni tra entità in un sistema di informazione ed entità fisiche, il secondo invece è molto importante per quelle ontologie che necessitano della contestualizzazione.

Infine per quanto riguarda l'informazione linguistica, necessaria per la rappresentazione delle entità ontologiche, è stato introdotto un modello lessicale multilingue chiamato LingInfo. Esso permette di assegnare ad ogni semplice parola o termine più complesso, il suo significato con una più o meno estesa rappresentazione della sua forma linguistica (nome, verbo, parola composta, ecc.), grazie alla scomposizione morfosintattica del vocabolo.

Tutto ciò consente di utilizzare SOBA (SmartWeb Ontology-Based Annotation) [**20**], un'applicazione di SWIntO, come strumento di annotazione: LingInfo, infatti, risulta essere sia un'interfaccia ottimale tra i sistemi, che permette di mappare i termini nell'ontologia di base, sia una ricca risorsa di informazioni per l'annotazione linguistica.

SOBA è costituito da un collegamento Web, che scarica automaticamente i documenti importanti riguardanti i domini di interesse dai loro siti principali, da un componente di annotazione linguistica che procede con la disambiguazione del testo grazie alla parte svolta da LingInfo, e infine da un componente che trasforma l'annotazione linguistica in una rappresentazione basata sull'ontologia di riferimento: ciò spiega il ruolo fondamentale dell'ontologia, senza la quale non si sarebbe avuta la chiarezza concettuale necessaria allo sviluppo di questa applicazione.

Capitolo 6

6. Analisi delle ontologie definizionali

Le ontologie definizionali sono state analizzate più approfonditamente per verificare la possibilità di eseguire l'annotazione rispetto ad esse: sono state prese come esempi le ontologie precedentemente esaminate, Cyc, SUMO e DOLCE, essendo le più conosciute e utilizzate.

Per prima cosa sono state cercate e scaricate le versioni disponibili al pubblico di ciascuna di esse dai rispettivi siti di appartenenza: OpenCyc da http://www.opencyc.org/downloads, SUMO da http://www.ontologyportal.it e DOLCE Lite + (versione 3.9) da http://www.loa-cnr.it/DOLCE.html.

Delle varie versioni delle ontologie presenti si sono scelte quelle scritte in OWL (Web Ontology Language) [**22**], il principale linguaggio delle ontologie ed anche il linguaggio necessario alla corretta esecuzione dei software successivamente utilizzati per l'annotazione.

Il primo di essi è Protégè [**24**], un'open platform per la modellazione delle ontologie e l'acquisizione delle conoscenze, la quale è stata utilizzata per verificare dimensioni e contenuti delle versioni scaricate, grazie alla sua grafica chiara e immediata. Il secondo invece è MOMIS (Mediator envirOnment for Multiple Information Sources) [**25**], un framework per l'estrazione e l'integrazione di informazioni per sorgenti dati strutturate e semistrutturate, con il quale si è eseguita tutta la parte riguardante l'annotazione.

6.1 OWL: Web Ontology Language

Il Web Ontology Language (OWL), è una famiglia di linguaggi per la rappresentazione della conoscenza e per la creazione delle ontologie, sostenuto dal World Wide Web Consortium (W3C). OWL è considerato una delle fondamentali tecnologie che sostiene il Semantic Web ed attrae interessi sia accademici che commerciali.

L'OWL è frutto della ricerca e della revisione del linguaggio per le ontologie chiamato DAML+OIL, ed è tuttora in sviluppo grazie al gruppo di lavoro W3C che continua a creare versioni sempre più aggiornate (OWL 1.1, OWL 2, ecc.[**21**]). Esso è stato creato per fornire un modo comune all'elaborazione dei contenuti delle informazioni presenti sul Web, invece che visualizzarli solamente, e per essere direttamente letto dalle applicazioni dei computer, invece che dagli uomini, grazie ad un linguaggio macchina facilmente interpretabile.

OWL fornisce tre sottolinguaggi, aventi crescente espressività, costruiti per diverse tipologie di utilizzo e di utenti:

- *OWL Lite*: serve a quegli utenti che necessitano principalmente di una classificazione gerarchica e di vincoli semplici;
- *OWL DL*: serve a quegli utenti che vogliono la massima espressività, includendo *completeness* computazionale (ovvero che tutte le conclusioni siano computabili) e la *decidability* (ovvero che tutte le computazioni terminino in un tempo finito); esso contiene tutti i costrutti linguistici dell'OWL ma possono essere usati solo con alcune restrizioni; OWL DL è così chiamato in riferimento alla Description Logic (DL), il campo di ricerca da cui deriva la logica formale su cui si fonda il linguaggio;
- *OWL Full*: serve agli utenti che oltre a volere la massima espressività necessitano anche della libertà sintattica del RDF (uno standard web scritto in XML [**23**]).

I dati descritti da un'ontologia in OWL sono interpretati come un insieme di *individui* e di *proprietà-relazioni* che collegano i primi: l'ontologia diventa quindi un insieme di assiomi che pongono vincoli su gruppi di individui (chiamati *classi*) e sui tipi di relazioni permessi tra di essi. Grazie al linguaggio OWL si ha la possibilità di definire tutti questi elementi e di organizzarli in una rete semantica, la quale permette al sistema di dedurre altre informazioni da quelle già esplicitamente presenti.

6.2 Protégè

Protégé è uno strumento open-source sviluppato alla Stanford Medical Informatics; esso si avvale di una numerosa comunità di utenti. Anche se l'evoluzione di Protégè è stata storicamente guidata da applicazione biomediche, il sistema ha un dominio indipendente ed è stato usato con successo in molte altre aree applicative.

Protégè è una piattaforma aperta per lo sviluppo e la modellazione delle ontologie e per l'acquisizione delle conoscenze; con esso si è in grado di elaborare le ontologie in OWL, accedere alle DL (description logics) e di acquisire istanze per il markup semantico (Figura 6.1).

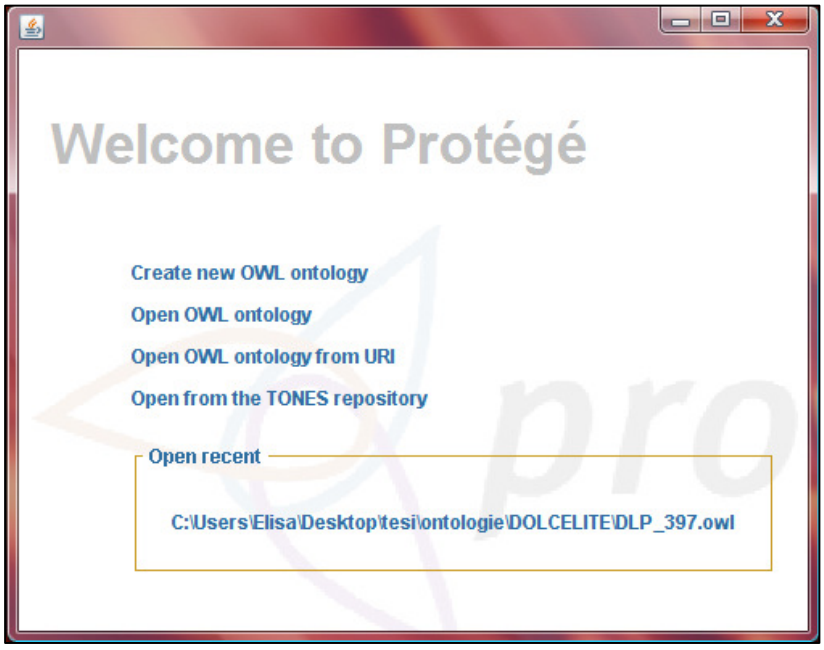

Figura 6.1: Schermata iniziale di Protégè

Essendo le ontologie complicate da costruire e raffigurare, un tool per la revisione grafica avente una semplice sintassi e un meccanismo di costruzione valido, può migliorare significativamente questa situazione. La difficoltà nella realizzazione delle ontologie è insita nel fatto che sono dei modelli formali di domini della conoscenza umana: già quest'ultima è complicata da comprendere, ma si aggiunge anche il fatto che non esiste una sola corretta rappresentazione.
Quindi un'interfaccia utente può semplificare e accelerare i compiti più frequenti e allo stesso tempo promuovere in modo pratico il miglioramento degli schemi di disegno. Inoltre Protégè

propone un'assistenza intelligente per la costruzione delle ontologie, indicando gli errori di modellazione più ovvi.

Come la maggior parte degli strumenti per la modellazione, l'architettura di Protégè è chiaramente suddivisa in una parte di "modellazione" e in una di "visualizzazione": la prima consiste nel meccanismo interno di rappresentazione per le ontologie e le basi di conoscenza; la seconda invece provvede all'interfaccia utente che raffigura la manipolazione dei modelli di base.

Esso principalmente può rappresentare ontologie costituite da classi, proprietà, caratteristiche delle proprietà (vincoli, ecc.) e istanze; infatti comprende una Java API per creare delle query e per manipolare i modelli, i quali rientrano nelle categorie di sistemi object-oriented e frame-based.
Protégè può infine essere utilizzato per caricare, costruire e salvare ontologie in diversi formati, come RDF, UML, XML, OWL e come database relazionali.

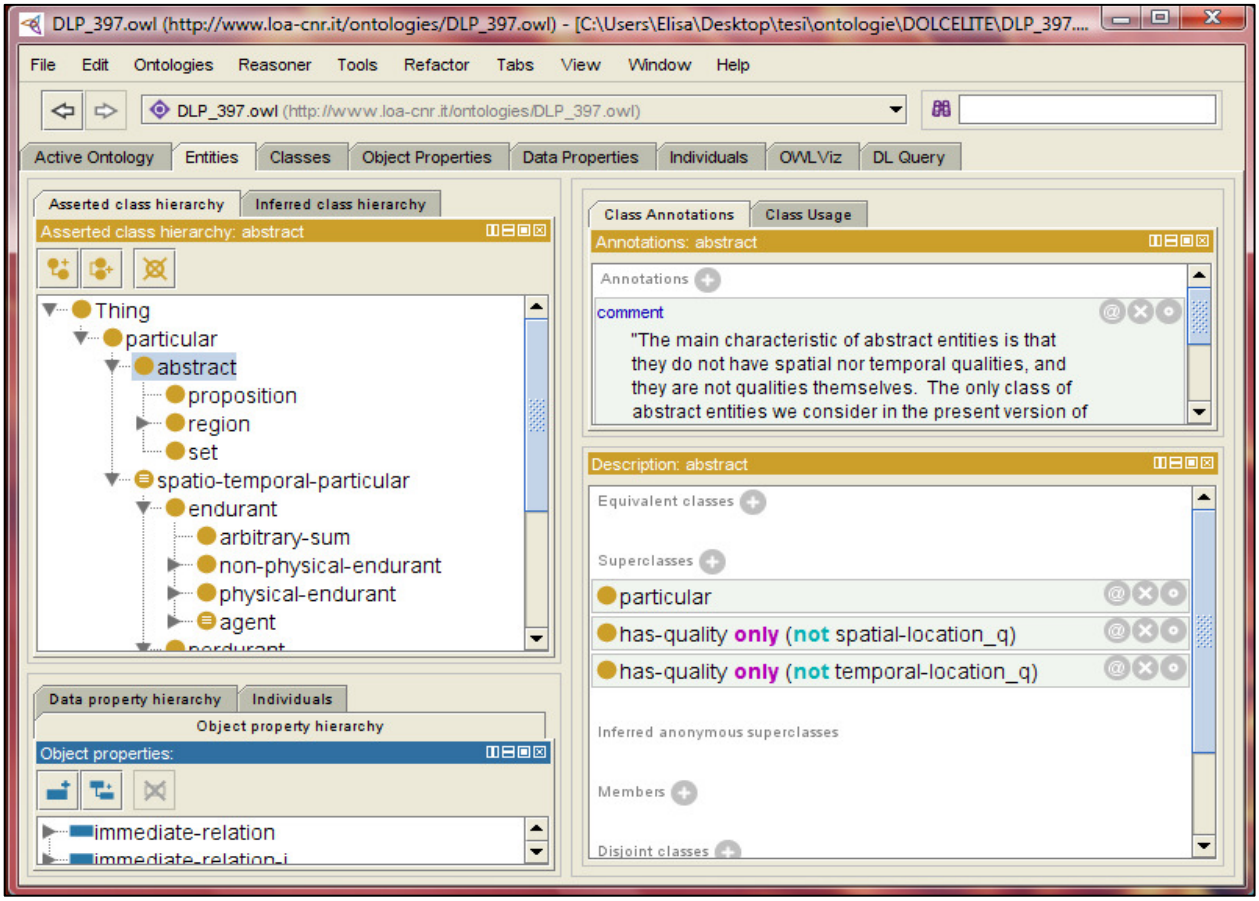

Figura 6.2: DOLCE Lite + visualizzato con Protégè

In questa analisi Protégè è stato utilizzato per verificare i formati delle ontologie scaricate e per visualizzarne i contenuti: sono state caricate senza problemi sia la versione in OWL di SUMO, che la versione Lite di DOLCE (Figura 6.2), mentre per OpenCyc si è avuto un problema di spazio, essendo l'ontologia molto grande (circa 166 MB) il quale ha dato come output un errore (java.lang.outofMemoryError: java heap space).

6.3 MOMIS

MOMIS (Mediator envirOnment for Multiple Information Sources) [**3**], è uno strumento intelligente creato per l'integrazione delle informazioni estratte da una molteplicità di fonti eterogenee; generalmente, nei sistemi informatici basati su Internet, le informazioni e le fonti da cui hanno origine, vengono sviluppate indipendentemente le une dalle altre, risultando di conseguenza, differenti in terminologia, struttura e contesto: è quindi necessario poterle riconciliare.

Lo scopo principale dell'estrazione dell'informazione e delle tecniche di integrazione sviluppate in MOMIS è di costruire viste integrate e sintetizzate delle informazioni provenienti da sorgenti eterogenee, in modo da fornire all'utente un'interfaccia per le query, uniforme, unica ed indipendente da dove le informazioni risiedono e dal loro livello di eterogeneità.

Inoltre, vista la grande quantità di informazioni e di risorse presenti nel Web, è stato importante sviluppare anche tool che permettano l'automatizzazione del maggior numero di tecniche ed attività, per l'estrazione e l'integrazione dei dati: a tale scopo è stato realizzato il tool SI-Designer [**3**].

Le informazioni e gli schemi delle sorgenti locali sono estratti attraverso dei *wrapper*, i quali utilizzano il linguaggio ODL_I^3 sia per ricavare le descrizioni delle sorgenti che per salvare le strutture dati di quest'ultime: l'ODL_I^3 è un'estensione del linguaggio ODL (Object Definition Language) [**28**], il quale è utilizzato per l'integrazione dei dati strutturati (schemi ad oggetti), ed è stato modificato per consentire l'Integrazione Intelligente delle Informazioni (I^3) e l'utilizzo dei dati semistrutturati.

MOMIS usa il sistema lessicale di WordNet come ontologia di riferimento, per costruire il *Common Thesaurus*, il quale risulta composto da relazioni tra gli elementi degli schemi

estratti dalle descrizioni delle rispettive fonti; le relazioni sono ottenute automaticamente da quelle semantiche e lessicali esistenti tra i concetti di WordNet o imposte esplicitamente dall'autore dell'integrazione.

La scelta di WordNet come database lessicale di riferimento, si è basata sia sulla sua ampiezza di contenuti, sia sulla sua grande diffusione, risultando quindi uno strumento professionale completo; inoltre è, come già detto, freeware.

Nell'analisi delle ontologie si è utilizzato MOMIS, proprio per il suo riferimento nell'annotazione a WordNet: se è possibile l'annotazione in generale rispetto a WordNet e se è possibile annotare anche le ontologie fondazionali rispetto a WordNet, allora è possibile in generale l'annotazione rispetto alle ontologie.

Inoltre MOMIS presenta sia la possibilità di annotare manualmente i termini rispetto a WordNet, sia di utilizzare diverse tipologie di algoritmi per l'annotazione automatica, i quali hanno permesso un confronto tra i vari risultati ottenuti e una misurazione della qualità delle operazioni (di ciò si parlerà più approfonditamente nel capitolo successivo).

Per prima cosa si è cercato di estrarre le ontologie attraverso MOMIS, utilizzando il *wrapper OWL*: purtroppo sia OpenCyc che SUMO, essendo molto pesanti, hanno dato problemi a causa dello spazio disponibile non sufficiente (vedi paragrafo 6.2), mentre DOLCE Lite +, dopo alcune correzioni nelle linee di comando del wrapper, è stata facilmente caricata.
Dalla Figura 6.3 è possibile vedere le classi di cui si compone l'ontologia.

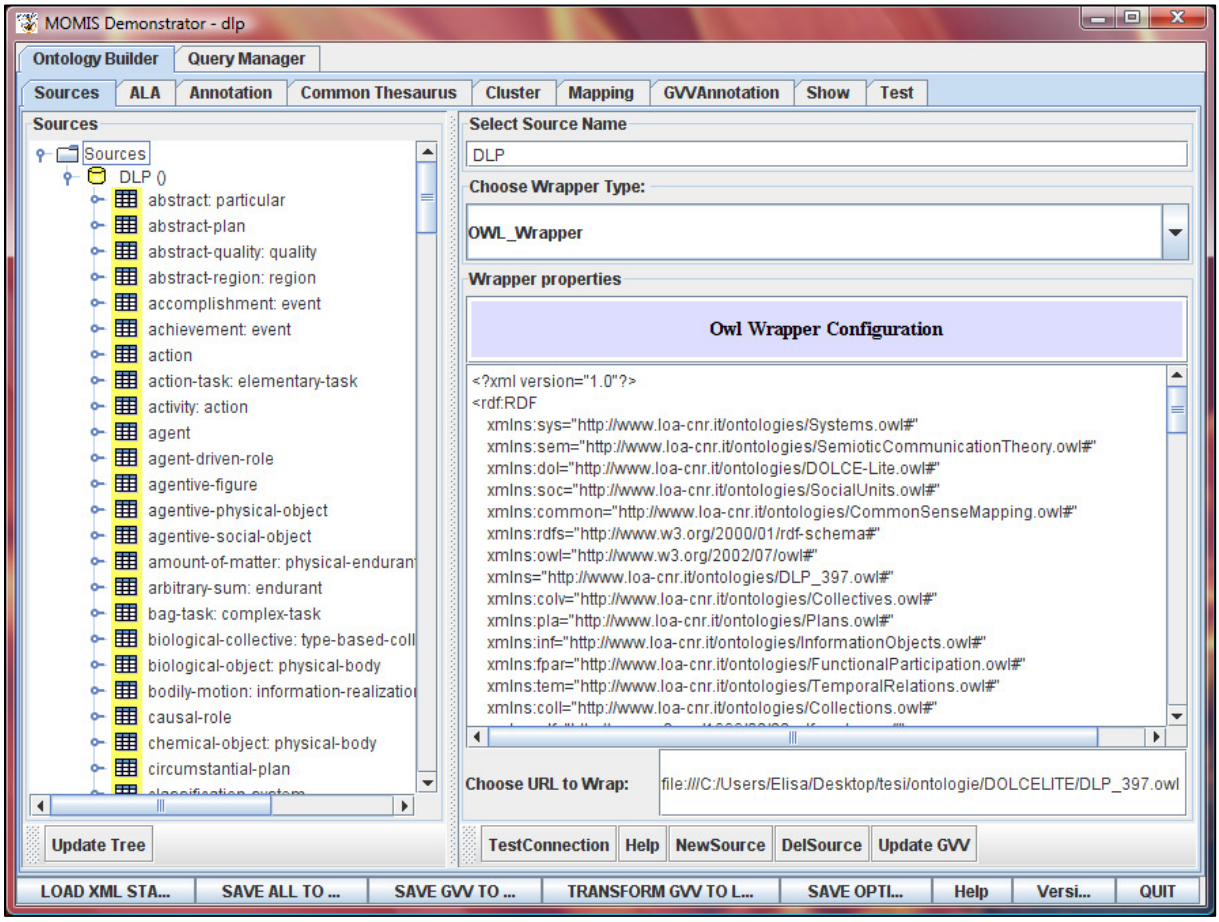

Figura 6.3: DOLCE Lite + in MOMIS

DOLCE Lite + [**27**], presenta le stesse classi principali di DOLCE (vedi paragrafo 5.2), ovvero si basa sulla fondamentale distinzione tra *perdurant* (o eventi), che hanno qualità temporali, *endurant* (o oggetti), che hanno qualità spaziali, ed *elementi astratti*, privi di caratteristiche spazio-temporali. Sono presenti inoltre, il concetto di *qualità* dipendente dall'esistenza dell'entità a cui fa riferimento e distinta dal valore che assume (*quale*), e il concetto di *regione*, intesa come dimensione spaziale, temporale o concettuale, a seconda della qualità considerata.

Ad esse si aggiungono le informazioni contenute nei moduli addizionali come DnS (Description and Situations), Plans, Common_Sense_Mapping, Social_Units, ecc.[**26**], che completano l'ontologia pur mantenendola più "leggera" della versione completa di DOLCE.

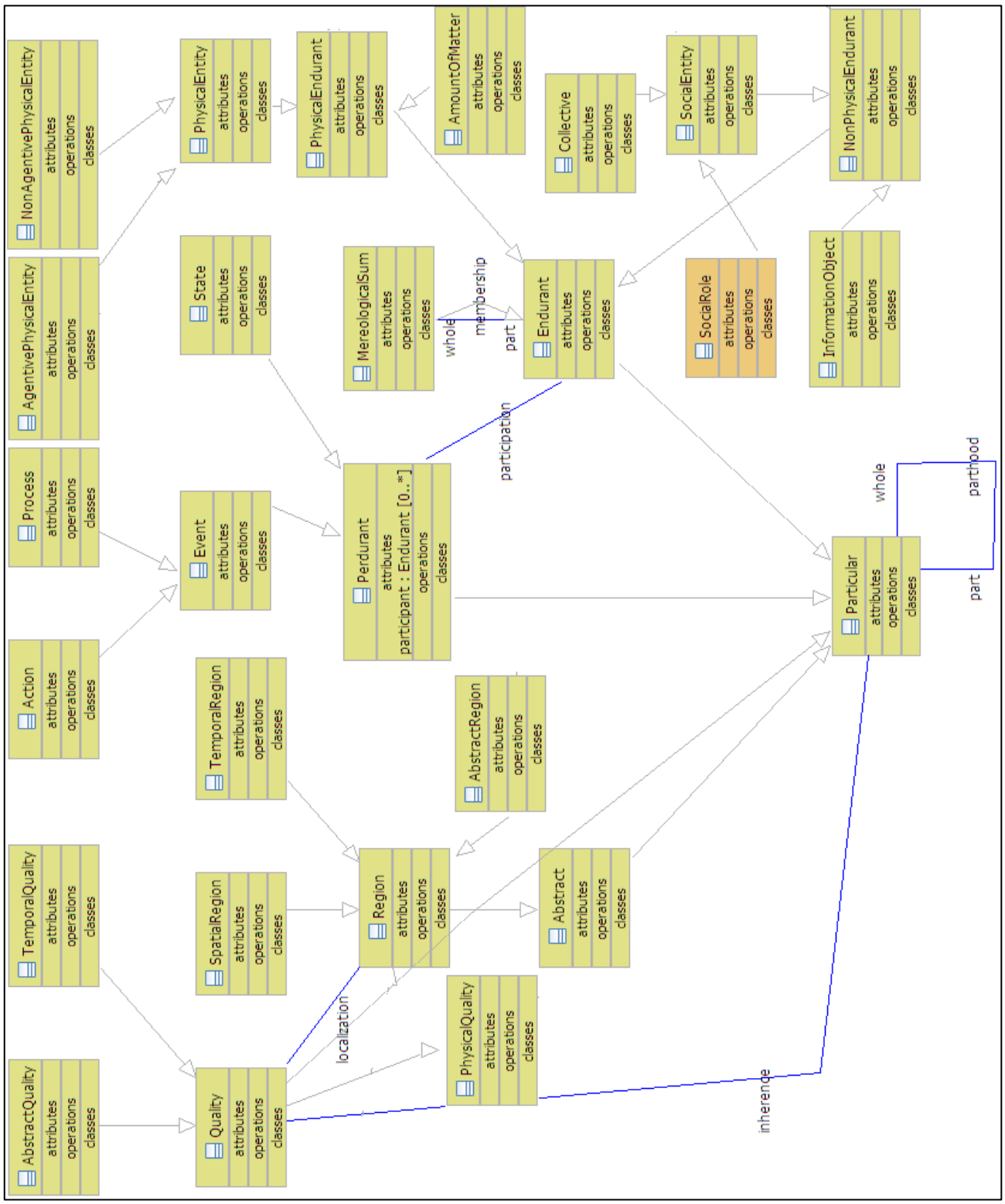

Figura 6.4: DOLCE Lite +

Capitolo 7

7. Annotazione di DOLCE Lite +

La varietà e la quantità di fonti che possono essere integrate è enorme, e come affermato in precedenza, risulta difficile per il progettista-utente avere una conoscenza sufficiente dei contenuti di ciascuna di esse. Per questo motivo e per risparmiare tempo e lavoro, il processo di annotazione deve essere il più automatizzato possibile.

Il problema maggiore risiede nel come i dati sono etichettati, ovvero è spesso difficile capire il significato che si ha dietro i nomi dei concetti appartenenti agli schemi di diverse risorse; l'annotazione lessicale, risulta quindi essere fondamentale per la comprensione del loro significato e per le successive operazioni di integrazione, per le quali è necessario conoscere le corrette relazioni tra i concetti.

L'annotazione, in generale, consiste nell'inserimento di informazioni extra sui dati della risorsa: essa può essere effettuata in relazione ad un'ontologia o ad un vocabolario, i quali avendo informazioni condivisibili, costituiscono una base completa per l'integrazione.

Per annotare DOLCE Lite + si è utilizzato MOMIS (Figura 7.1), il quale possiede diversi algoritmi per l'annotazione lessicale automatica (ALA-Automatic Lexical Annotator): durante questo processo i termini (o dati della risorsa) sono automaticamente annotati rispetto al database lessicale di WordNet (pur essendo il metodo indipendente dalla scelta del riferimento).

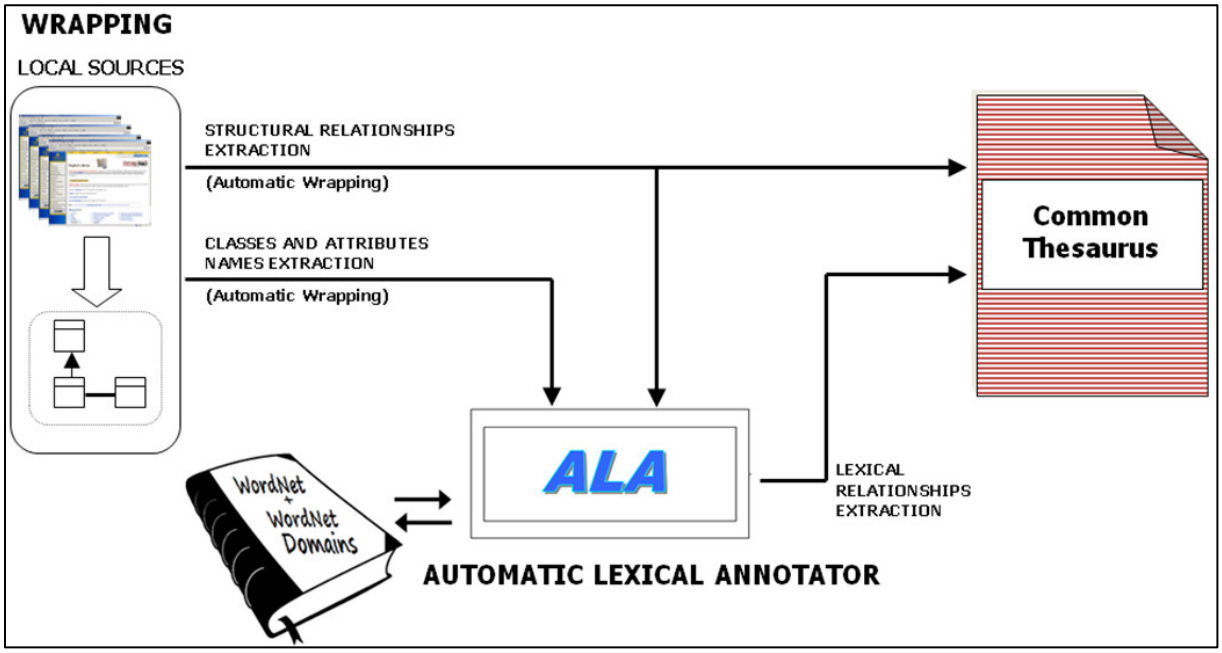

Figura 7.1: Annotazione in MOMIS

Gli algoritmi per l'annotazione automatica fanno riferimento alle tecniche per la disambiguazione del significato dei termini (WSD, Word Sense Disambiguation) [**29**] sviluppate nella'area di ricerca del Semantic Web, e possono essere utilizzati isolatamente o combinati tra loro per ottenere maggiori risultati.

Oltre a quella automatica, in MOMIS, è possibile eseguire l'annotazione manuale degli elementi delle risorse: questa operazione è stata la prima ad essere applicata all'ontologia presa in esame, per poterne utilizzare i risultati come confronto e misura di quelli dati dall'annotazione automatica.

7.1 Annotazione manuale

Per prima cosa l'ontologia di DOLCE Lite + è stata caricata in MOMIS, quindi si è selezionato il *tab* dell'annotazione manuale e ad ogni termine si è associato il suo significato. Grazie all'uso di WordNet come database lessicale di riferimento, per ciascun termine si ha la scelta di diversi possibili significati da attribuirvi: essi sono classificati secondo le quattro categorie sintattiche di WordNet (nomi, aggettivi, verbi e avverbi) e sono elencati secondo la frequenza con cui un certo significato è attribuito ad un particolare termine.

In questo modo si lascia la possibilità all'utente (o designer) di collegare le parole con il loro significato più appropriato, e quindi di ottenere un'annotazione complessivamente corretta ed

esauriente, essendo chi compie l'operazione a conoscenza del contesto in cui si trovano i termini e delle loro accezioni particolari.

I termini dell'ontologia caricati in MOMIS hanno diversi colori: sono neri, se le parole sono riconosciute dal software perché presenti in WordNet; sono rossi, se invece MOMIS non trova nessun termine uguale nel database lessicale; infine vengono contrassegnati col verde quando l'annotazione è stata eseguita (Figura 7.2).

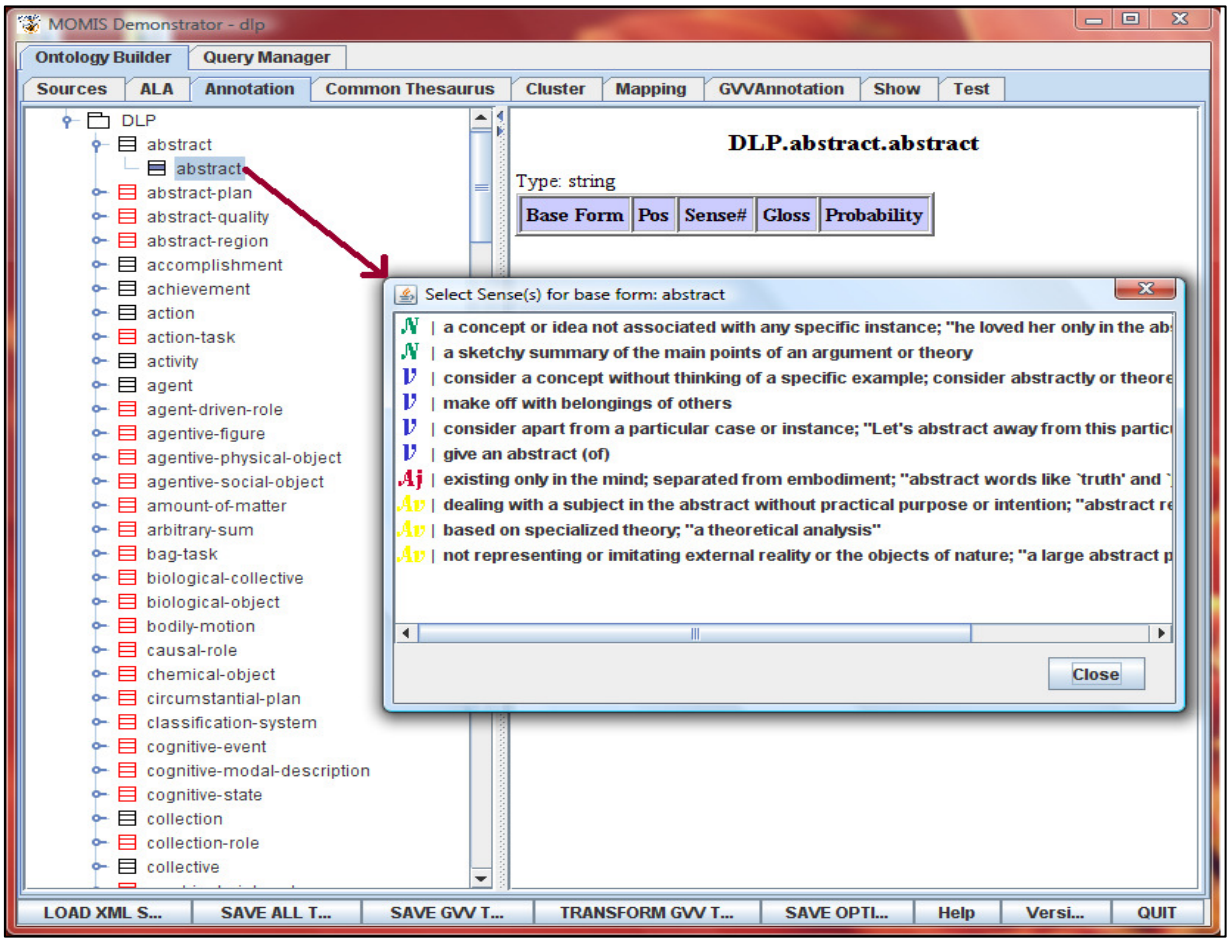

Figura 7.2: Annotazione di un termine di DOLCE Lite + con MOMIS

Nel primo caso è sufficiente fare clic con il tasto destro del mouse sul termine e selezionare dal menù a tendina "*Select sense*" per aprire l'elenco di significati (glosse) associati ad esso (lemma); nel secondo caso invece, si seleziona il comando "*Add base form*" e si digita il termine in una forma conosciuta da WordNet: in entrambe i casi infine si seleziona il senso desiderato per completarne l'annotazione.

Quando un termine è annotato, nella finestra a fianco è visualizzata una tabella contenente nella prima colonna la forma base del termine, nella seconda la categoria sintattica di appartenenza (1=nome, 2=verbo, 3=aggettivo, 4=avverbio), nella terza il numero del senso associato in ordine di frequenza d'uso (1=senso più comune), nella quarta la glossa, e nell'ultima la probabilità dell'annotazione.

7.1.1 Problemi incontrati

Durante l'annotazione manuale si sono incontrati alcune difficoltà soprattutto legate alla forma dei termini dell'ontologia o alla comprensibilità del loro esatto significato.

Per cercare di scegliere la glossa più appropriata per ciascun lemma si è utilizzata la documentazione online di DOLCE Lite +, (presa dal sito della W3C, http://www.w3.org/2001/sw/BestPractices/WNET/DLP3941_daml.html), la quale oltre a fornire tutte le classi, le proprietà e i metodi, ne dà una definizione completa.

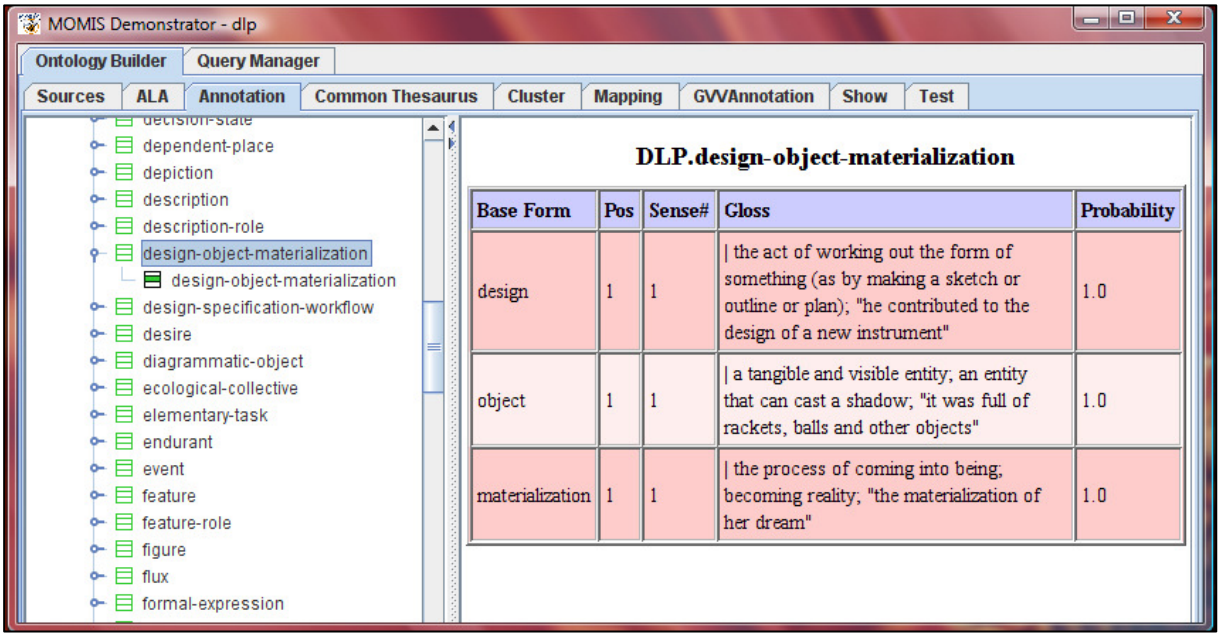

Figura 7.3: Annotazione manuale di un termine composto da più parole in MOMIS

Il problema più frequente è risultato quello dato dalle *lineette* (-); infatti quasi tutti i termini dell'ontologia sono composti da più parole collegate tra loro dai trattini, i quali non permettono a MOMIS di trovare il corrispondente lemma all'interno di WordNet: per risolvere ciò si è pensato di annotare ciascuna parola in modo indipendente, ovvero si sono annotati i termini parola per parola, facendo corrispondere a ciascun termine un numero di

glosse pari al numero di parole che lo compongono (Figura 7.3).

In alcuni casi si sono trovate le parole composte collegate dal carattere *underscore* (_), e quindi si è annotato con un'unica glossa: è il caso dei termini come *physical-body* o *physical-phenomenon* a cui sono state fatte corrispondere le definizioni dei lemmi *physical_body* e *physical_phenomenon*.

In altri casi invece si è trovato che alcuni termini, pur essendo composti da più parole, erano definiti in modo sufficientemente completo da una sola glossa: è il caso del termine *measurement-unit*, in cui la glossa corrispondente al lemma *unit* affermando "any division of quantity accepted as a standard of measurement or Exchange", risulta una definizione abbastanza completa di tutto il termine; si è però preferito annotare anche il lemma *measurement* per rimanere coerenti con il metodo scelto inizialmente.

Un altro problema si è avuto con i termini del tutto inesistenti nel database di WordNet: ad esempio con il termine *quale*, che indica il valore delle qualità nell'ontologia, ma il cui lemma è stato inventato appositamente per lo scopo ed è quindi mancante nel vocabolario di riferimento; si è perciò annotato il termine come unione delle glosse *quality* e *value*, cercando di avvicinare il più possibile la definizione data dall'ontologia.

Lo stesso problema si è avuto per i termini derivati direttamente dalla filosofia, come *endurant* e *perdurant*: nel primo caso, si è annotato il termine come verbo *endure*, la cui glossa dice "continue to exist", essendo sia il lemma che la definizione il più simile possibile al vocabolo originale; nel secondo caso invece, non si è trovato nessun lemma che potesse assomigliare al termine considerato; si è quindi scelto di annotarlo utilizzando la traduzione proposta dalla documentazione dell'ontologia, la quale pone come sinonimo di *perdurant* il termine *occurrence*, il cui lemma è riconosciuto da MOMIS.
In entrambe i casi però non si riesce ad annotare il vero significato dei termini, infatti quest'ultimo è altamente filosofico e complesso [**27**], inoltre nell'ontologia assume un'accezione particolare che non si trova tra le glosse disponibili in WordNet, il quale contiene i sensi delle parole legati all'uso comune.

Questo fatto si riscontra in diversi termini, ad esempio *commitment*, nell'ontologia è definito come "una descrizione modale e cognitiva caratterizzata da obblighi e diritti fissati da almeno

uno dei suoi ruoli", mentre la glossa con cui è stato annotato è una spiegazione della parola "impegno" (traduzione italiana del termine): si vede quindi come con l'annotazione, alcune delle particolari accezioni proprie dell'ontologia siano perse.

Infine alcuni termini sono stati annotati con glosse più simili al significato presente nell'ontologia, ma che non corrispondono alla categoria sintattica corretta: è il caso del termine *collective*, il quale è stato annotato come aggettivo, pur essendo un nome, essendo la glossa "forming a whole or aggregate", più appropriata al senso usato in DOLCE Lite +.

7.2 Annotazione automatica

Come precedentemente accennato, dopo l'annotazione manuale, si è verificato se il processo potesse essere ripetuto anche con quella automatica, in modo da confrontarne i risultati e vedere se sia davvero possibile annotare automaticamente anche un'ontologia definizionale.

Si è già detto che MOMIS fa uso di diversi tipi di algoritmi per l'annotazione automatica, i quali sono raggruppati nella sessione ALA (Automatic Lexical Annotator). Essi possono essere usati indipendentemente gli uni dagli altri o combinati tra loro: per prima cosa sono stati osservati gli output di ciascun algoritmo preso singolarmente, quindi si sono analizzati e confrontati i risultati delle loro combinazioni in serie (*pipe*) e in parallelo.

In generale, gli algoritmi cercano di estrarre le relazioni presenti tra i concetti dell'ontologia in modo da contestualizzare ciascun termine, quindi lo confrontano con i legami presenti in WordNet, per potervi poi associare il significato più appropriato.

Gli algoritmi sono inoltre in grado di eliminare i trattini dalle parole composte, ed associano quindi una o più glosse per ogni parola che compone il termine; ciò risulta coerente con la scelta effettuata per l'annotazione manuale e rende quindi possibile il confronto.

I risultati ottenuti automaticamente sono stati confrontati con l'annotazione manuale, verificandone l'effettiva accuratezza attraverso i due parametri utilizzati nel campo della disambiguazione del significato delle parole: *recall* e *precision* [**29**].
La *recall* è una misura di completezza ed è definita come il numero di annotazioni corrette restituite dall'algoritmo diviso il numero di annotazioni da eseguire, mentre la *precision* è

vista come una misura di esattezza o fedeltà ed è definita come il numero di annotazioni corrette restituite dall'algoritmo diviso il numero di annotazioni eseguite [**31**].
Per gli algoritmi cha associano più glosse ad ogni termine, si sono considerate corrette le annotazioni aventi tra di esse, anche quella considerata appropriata e quindi coincidente all'annotazione manuale.

7.2.1 Structural Disambiguation (SD) algorithm

L'algoritmo di disambiguazione strutturale (SD) [**29**] ricava dallo schema in ODL_I^3, in cui la sorgente è stata trasformata in questo caso dal *wrapper OWL*, le relazioni strutturali tra gli elementi dello schema, per dedurne le relazioni lessicali in riferimento a WordNet.

Esso trasforma le relazioni di specificazione e i vincoli tra le classi in relazioni di sinonimia, iponimia e ipernimia, quindi cerca nel database lessicale di WordNet se vi sono le stesse relazioni tra i lemmi e infine, annota i termini con i significati trovati, che possono essere anche più di uno per ciascuno di essi.

Utilizzando SD per annotare DOLCE Lite +, il risultato è di soli 4 termini restituiti dall'annotazione su 374 (una percentuale di poco superiore all'1%). Le glosse associate non corrispondono all'esatto significato dei termini e riportano dei valori di recall e precision dello 0.53% e del 50%.

7.2.2 WordNet Domains (WND) e WordNet First Sense (WNFS) algorithms

Il primo algoritmo fa riferimento a WordNet Domains [**29**]: esso può essere considerato un'estensione di WordNet, o una sua risorsa lessicale in cui i synset sono annotati con uno o più nomi di domini. Queste informazioni sono complementari a quelle già presenti in WordNet, infatti i domini raggruppano i synset appartenenti agli stessi argomenti riducendo il livello di ambiguità delle parole polisemiche.

L'algoritmo WND per prima cosa, esamina tutti i possibili synset connessi al termine preso in esame, ne estrae i domini associati e con queste informazioni calcola una lista di domini prevalenti; l'operazione è eseguita per tutti i termini da annotare e successivamente si confrontano le liste per determinare i domini predominanti nel contesto; infine i termini sono annotati con i synset che appartengono a quel dominio.

Se un termine non appartiene ad un particolare dominio o non ha nessun synset collegato ad esso, allora si usa l'algoritmo WNFS, il quale ritorna per ciascun termine il primo senso del thesaurus di WordNet, che è il senso più comune e usato per definirlo.

Con il primo algoritmo si associano più glosse a ciascun lemma e sono annotati solo quei termini rientranti nei domini riconosciuti, mentre con il secondo tutti i termini sono annotati con una sola glossa per termine, la quale fa riferimento al primo senso di WordNet: in entrambi i casi però non sempre il significato annotato è quello appropriato per interpretare correttamente l'ontologia.

Utilizzando l'algoritmo WND per annotare l'ontologia si ha come risultato l'annotazione di 84 termini su 374 (22.5% circa), con una recall del 2% circa e una precision del 9.5%.
Mentre con l'algoritmo WNFS si riescono ad annotare 360 termini (96%), con una recall del 72% circa e una precision del 74%.

7.2.3 Gloss Similarity (GS) e Iterator Gloss Similarity (IGS) algorithms

L'algoritmo GS, utilizza un metodo di disambiguazione del significato basato sull'estrazione delle glosse collegate ai termini all'interno di WordNet. Ciascuno di esso è infatti legato ad una o più glosse che descrivono tutti i possibili sensi della parola, e ad una serie di frasi-esempio che descrivono il possibile uso di ciascun significato della parola in un certo contesto.

L'algoritmo per ricavare il synset corretto da associare al termine si basa sulla *vicinanza topologica* [**30**], con la quale si intende quell'insieme di attributi che sono strettamente legati al termine in ambito topologico.
La logica del metodo si basa sul fatto che esaminando le glosse di un termine ambiguo, si può ricavare quella corretta esaminando i termini che a sua volta la compongono e analizzando la vicinanza topologica di essi rispetto al vocabolo da disambiguare; la glossa che risulta avere più termini 'vicini topologicamente' al termine da disambiguare, è quella avente il senso corretto.

È lo stesso concetto riproposto in ambito topologico del metodo di Lesk [**30**], il quale afferma che il senso esatto di un termine in un contesto linguistico, è quello avente una glossa, presente in un dizionario, che condivide più vocaboli possibili con il contesto stesso.

L'algoritmo IGS, propone lo stesso metodo dell'algoritmo precedente, ma invece che fare riferimento alle glosse di un solo termine, considera tutte le glosse presenti nello stesso contesto.

Esso associa inizialmente una glossa ad un termine, quindi partendo da esso, disambigua tutti gli altri, l'operazione è poi reiterata in modo da verificare che anche la prima associazione fosse corretta e così via per tutti termini, in una concatenazione di glosse.
Il processo termina quando lo stesso synset è attribuito a tutti i termini per due volte; questo significa infatti che un'ulteriore iterazione non porterebbe cambiamenti nella disambiguazione dei significati.

Nell'analizzare DOLCE Lite +, entrambe gli algoritmi hanno annotato 360 termini su 374, (96%), associando una sola glossa per ciascuna parola dell'ontologia. Nel primo algoritmo si hanno una recall del 33% e una precision del 34%, mentre nel secondo i valori sono del 34% e del 35.6%: pur essendo i risultati molto simili, si hanno alcune annotazioni differenti, poiché il secondo algoritmo annota i termini concatenando le glosse.

7.2.4 Parallel execution

L'esecuzione in parallelo, prevede la contemporanea attuazione di tutti gli algoritmi precedentemente descritti e l'unione di tutti i risultati finali: questo fa si che ad ogni termine siano associati più synset. Il risultato è quindi il più ricco possibile ma perde in parte il significato di disambiguazione, essendo ad ogni termine associati un gran numero di significati anche non totalmente coerenti tra loro.

L'annotazione in parallelo sull'ontologia dà come risultato 360 termini annotati su 374 (96%), ma un'accuratezza poco soddisfacente.
Questo è comunque uno dei risultati più completi possibile, infatti si sono trovate una recall del 76.5% e una precision del 79%, avendo 74 annotazioni errate, spesso riguardanti gli stessi lemmi che sono ripetuti in più termini dell'ontologia, e 14 annotazioni mancanti, perché termini non riconosciuti dal database lessicale.

7.2.5 Pipe execution

L'esecuzione in pipe, o in serie, prevede l'esecuzione di tutti gli algoritmi in successione; ovvero ciascun algoritmo lavora sugli output dati dal precedente.

Ogni termine è disambiguato da al massimo un algoritmo, infatti se un termine non è annotato allora si passa al successivo, e così via.

In MOMIS è possibile scegliere manualmente l'ordine in cui eseguire gli algoritmi, si è scelto però, di utilizzare il *reliability values order*, che dispone automaticamente gli algoritmi a seconda della loro affidabilità, in modo da ottenere l'annotazione più accurata e precisa.

L'annotazione dà come risultato 360 termini su 374 (96%), e l'associazione di una sola glossa per parola, tranne che per quei termini che sono annotati solo dal primo o dal secondo algoritmo (SD e WND).
Il risultato ottenuto è il più simile a quello ottenuto con l'annotazione manuale e dalla comparazione si sono trovate una recall del 72% e una precision del 75%, infatti si hanno 91 annotazioni errate e 14 annotazioni mancanti.

7.3 Risultati sperimentali

I risultati ottenuti sono stati riuniti in schemi in modo da darne un quadro chiaro e complessivo e da rendere più immediato il confronto tra i dati. La tabella mostra gli esiti degli algoritmi presi singolarmente e l'istogramma ne riassume visivamente i contenuti (Figure 7.4 e 7.5).

	SD	**WND**	**WNFS**	**GS**	**IGS**
N annotazioni Restituite	4	84	360	360	360
N annotazioni Mancanti	370	290	14	14	14
N annotazioni Errate	2	76	92	236	232
N annotazioni restituite in Percentuale	1,1%	22,5%	96,3%	96,3%	96,3%
RECALL	**0,53%**	**2,14%**	**71,66%**	**33,16%**	**34,22%**
PRECISION	**50,0%**	**9,5%**	**74,4%**	**34,4%**	**35,6%**

Figura 7.4: Tabella di sintesi dei risultati dell'annotazione degli algoritmi

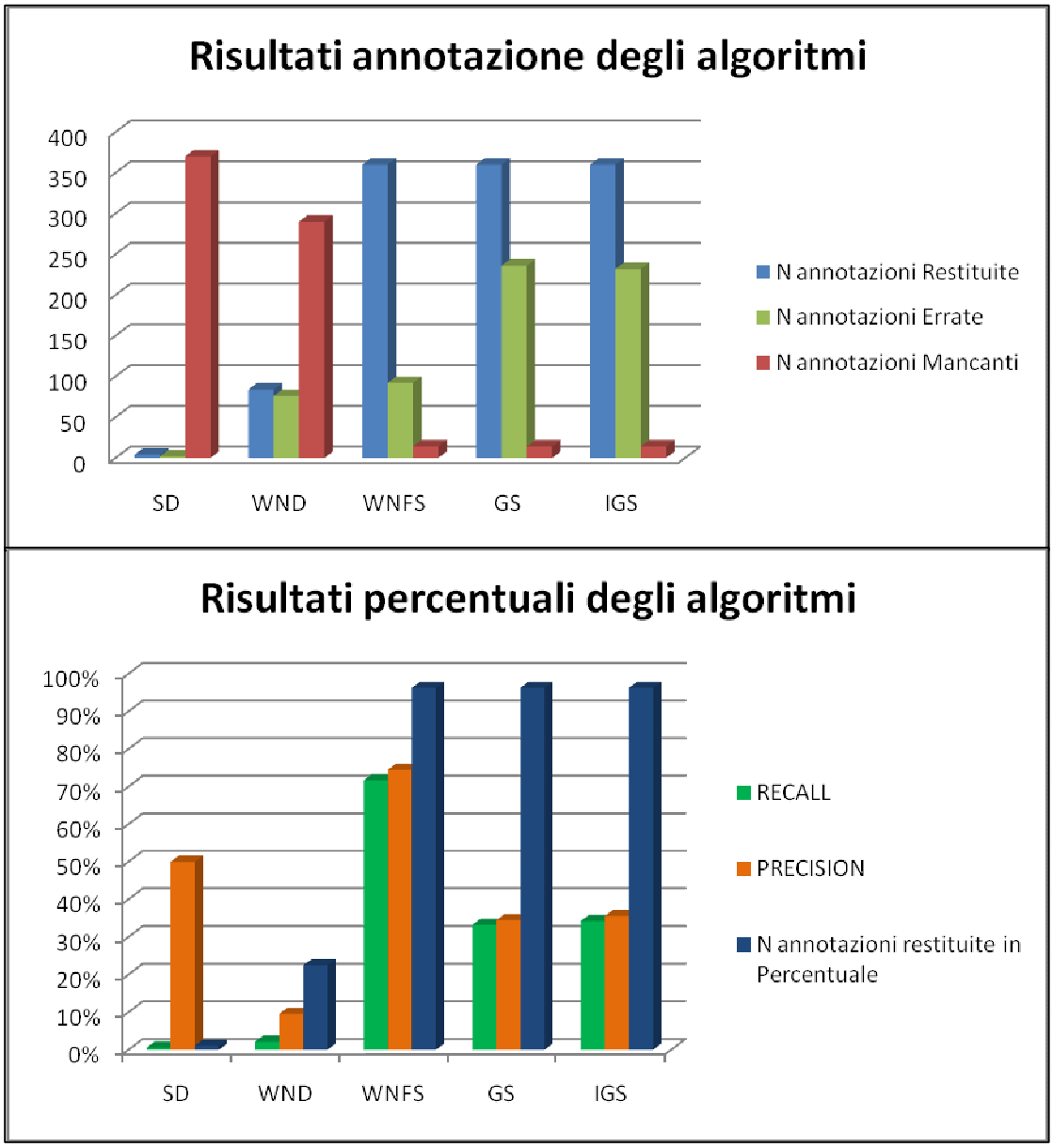

Figura 7.5: Istogrammi di sintesi dei risultati dell'annotazione degli algoritmi

Le successive invece mostrano il confronto tra annotazione in parallelo e in pipe, i cui risultati sono stati valutati rispetto l'annotazione manuale (Figure 7.6 e 7.7)

	Parallel exec.	Pipe exec.
N annotazioni Restituite	360	360
N annotazioni Mancanti	14	14
N annotazioni Errate	74	91
N annotazioni restituite in Percentuale	96,3%	96,3%
RECALL	**76,5%**	**71,9%**
PRECISION	**79,4%**	**74,7%**

Figura 7.6: Tabella di confronto tra annotazione in parallelo e in pipe

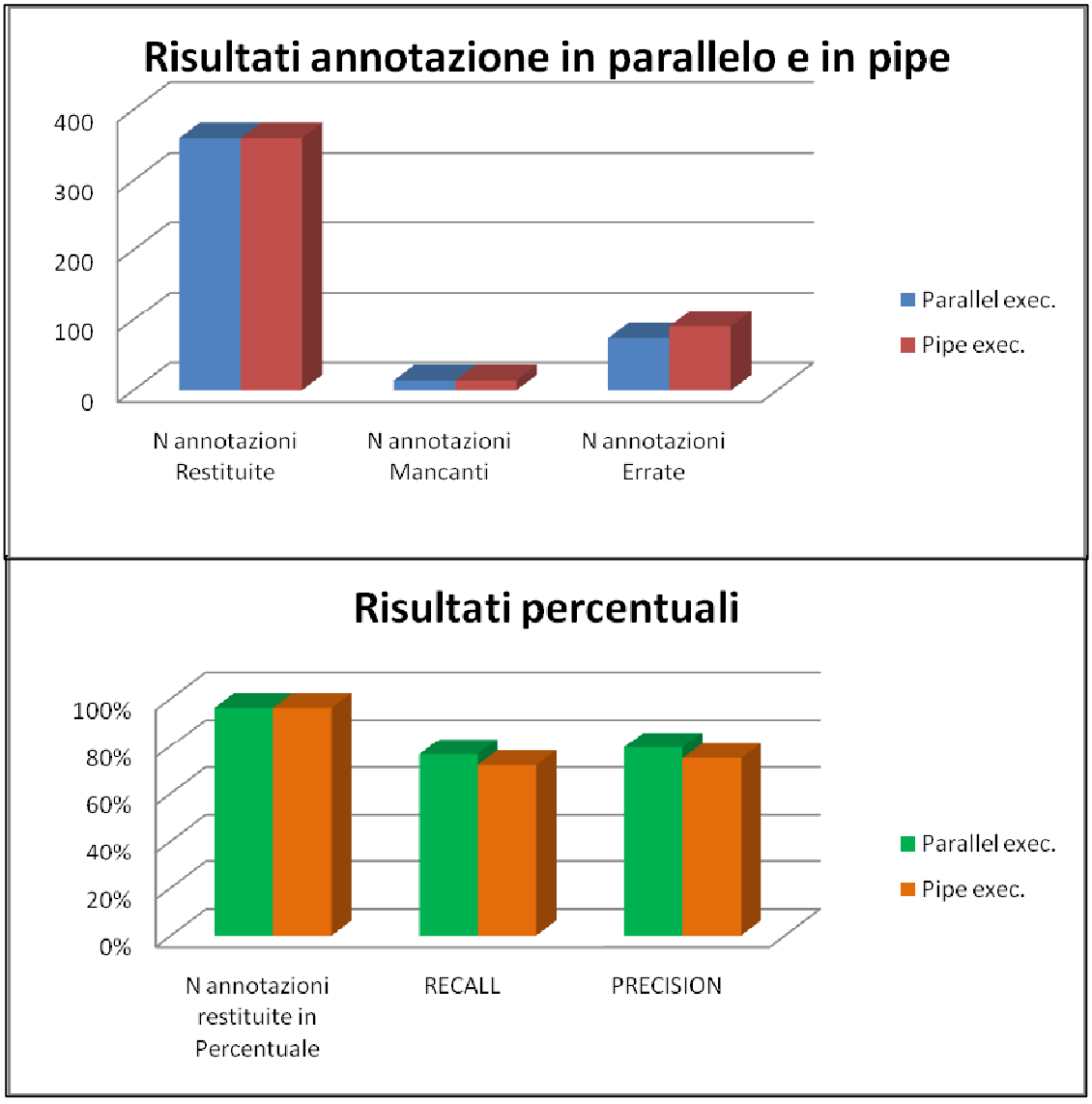

Figura 7.7: Istogrammi di confronto tra annotazione in parallelo e in pipe

Dal grafico è facile dedurre che l'esecuzione in parallelo degli algoritmi ottiene risultati migliori rispetto all'esecuzione in pipe, ma come precedentemente affermato, una minore accuratezza, essendo associati ad ogni termine più di un significato.

Capitolo 8

8. Conclusioni e Sviluppi futuri

Lo scopo di questa tesi è stato di analizzare le ontologie definizionali nel campo dell'annotazione e verificare la possibilità di allinearle a WordNet.

Durante l'operazione di analisi, le principali difficoltà affrontate sono state: (1) l'adattamento iniziale dell'ontologia al linguaggio OWL, che però è stata velocemente risolta grazie alle diverse versioni disponibili di ciascuna ontologia; (2) la presenza di numerosi termini composti, caratteristica comune a molte ontologie, ma che è stata superata grazie ai particolari algoritmi di MOMIS che riescono a scomporre i vocaboli; (3) infine, la presenza di termini filosofici o aventi accezioni diverse da quelle utilizzate nel linguaggio comune, che però sono stati facilmente definiti con l'annotazione manuale o utilizzando sinonimi di senso comune.

In conclusione, visti i risultati positivi ottenuti in fase sperimentale, si può affermare che buona parte dei termini di un'ontologia fondazionale possono essere annotati in maniera automatica, e attraverso l'annotazione manuale è possibile rifinire tale annotazione associando ai termini rimanenti il significato corretto rispetto a WordNet. Tale risultato, dimostra come sia possibile, attraverso un processo di annotazione semi-automatica, effettuare l'allineamento fra un'ontologia definizionale come DOLCE Lite + e il database lessicale WordNet.
Inoltre è lecito sostenere che l'ontologia definizionale potrà essere usata come strumento di annotazione, visto il concreto parallelismo con il database lessicale e la possibilità di dare una maggiore consistenza ai dati, definendoli non solo tramite la glossa ma anche attraverso la semantica della sua struttura.

Come progetti futuri sarà interessante esaminare, oltre alle basi di conoscenze ampie e generiche, le ontologie di dominio. In questo modo si potrà comprendere il loro possibile ruolo nella disambiguazione del significato, si potranno vedere i vantaggi dati da un insieme di nozioni più particolareggiato e approfondito sull'argomento da trattare e i possibili sviluppi che permetterebbero nell'annotazione di ambiti più specifici.

Ringraziamenti

Desidero ringraziare la Prof.ssa Sonia Bergamaschi e, in particolare, l'Ing. Serena Sorrentino per tutto l'aiuto fornito durante la realizzazione di questa tesi.

Ringrazio la mia famiglia, per la pazienza, l'aiuto e l'incrollabile sostegno durante tutti gli anni di studio, e soprattutto per l'affetto che da sempre e in ogni istante mi dimostrano.

Ringrazio tutti gli amici, che mi sostengono, sopportano e accompagnano in tutti i momenti della mia vita con la speranza che continuino sempre a farlo.

Un sincero grazie a tutti quanti.

Bibliografia

[**1**] DBGroup@unimo, Materiale didattico su Semantic Web

[**2**] Tesi Laurea Specialistica NOD: Serena Sorrentino: *Metodi di Disambiguazione del Testo ed Estensioni di WordNet nel sistema MOMIS*, Introduzione e Capitolo 1.

[**3**] Tesi Laurea VOD: Veronica Guidetti: *Intelligent Information Integration systems: extending lexicon ontology,* Capitoli 2,3,4.

[**4**] DBGroup@unimo, Materiale didattico su Ontologie

[**5**] Oltramari A., Borgo S., Catenacci C., Ferrario R., Gangemi A., Guarino N., Masolo C., Pisanelli D., *Il ruolo delle ontologie nella disambiguazione del significato*, 2003, Networks 2: 14-24, http://www.swif.uniba.it/lei/ai/networks/.

[**6**] Euzenat J., Shvaiko P., Ontology matching, Capitoli 2: *The matching problem*, 3: C*lassification of ontology matching techniques* e 4: *Basic techniques*, 2007.

[**7**] Sanfilippo A., Tratz S., Gregory M., Chappel A., Whitney P., Posse C., Paulson P., Baddeley B., Hohimer R., White A., *Ontological Annotation with WordNet*, In SemAnnot 2005, 5th International Workshop on Knowledge Markup and Semantic Annotation, 7th November 2005, Galway, Ireland, vol. 185, ed. Siegfried Handschuh, Thierry Declerck, & Marja-Riitta Koivun, pp. 27-36. Sun SITE Central Europe Workshop Proceedings (CEUR-WS.org), Aachen, Germany.

[**8**] Gangemi A., Guarino N., Masolo C., Oltramari A., Schneider L., *Sweetening ontologies with DOLCE*, in Benjamins et al. (eds.), Proceedings of the 13th European Conference on Knowledge Engineering and Knowledge Management (EKAW2002), Sigüenza, Spain, 2002.

[**9**] Matuszek C., Cabral J., Witbrock M., DeOliveira J., *An Introduction to the Syntax and Content of Cyc*, In *Proceedings of the 2006 AAAI Spring Symposium on Formalizing and Compiling Background Knowledge and Its Applications to Knowledge Representation and Question Answering*, Stanford, CA, March 2006.

[**10**] Siegel N., Goolsbey K., Kahlert R., Matthews G., The *Cyc System: Notes on Architecture*, http://www.cyc.com, Cycorp, Inc, 2004.

[**11**] Wikipedia, *Cyc*: http://en.wikipedia.org/wiki/Cyc

[**12**] Lenat D., Guha R.V., *Ideas for Applying Cyc*, 12/91, Microelectronics and computer technology corporation, MCC Technical Report Number ACT-CYC-407-91, December 1991.

[**13**] Curtis J., Cabral J., Baxter D., *On the Application of Cyc Ontology to Word Sense Disambiguation*, In Proceedings of the Nineteenth International FLAIRS Conference, pp. 652-657, Melbourne Beach, FL, May 2006.

[**14**] Niles I., Pease A., *Towards a Standard Upper Ontology*, In Proceedings of the 2nd International Conference on Formal Ontology in Information Systems (FOIS-2001), http://citeseer.ist.psu.edu/niles01towards.html.

[**15**] Niles I., Pease A., *Linking Lexicon and Ontologies: Mapping WordNet to the Suggested Upper Merged Ontology*, In Proceedings of the 2003 International Conference on Information and Knowledge Engineering (IKE 03), Las Vegas.

[**16**] Masolo C., Borgo S., Gangemi A., Guarino N., Oltramari A., Schneider L., *The WonderWeb Library of Fundational Ontologies: WonderWeb Deliverable D17*, Preliminary Report (ver. 2.0), 15-08-2002.

[**17**] Wikipedia, *DOLCE,* site: http://en.wikipedia.org/wiki/DOLCE

[**18**] *DOLCE Lite+*, *DOLCE Ultralite*, site: http://www.loa-cnr.it/DOLCE.html

[**19**] Gangemi A., Navigli R., Velardi P., *The OntoWordNet Project: extension and axiomatization of conceptual relations in WordNet*, Meersman R, et al. (eds.), Proceedings of ODBASE03 Conference, Springer, 2003.

[**20**] Oberle D., Ankolekar A., HitzlervP., Cimiano P., Sintek M., Kiesel M., Mougouie B., Vembu S., Baumann S., Romanelli M., Biutelaar P., Enegel R., Sonntag D., Reithiger N., Loos B., Porzel R., Zorn H.P., Micelli V., Schmidt C., Weiten M., Burkhardt F., Zhou J., *DOLCE ergo SUMO: On Foundational and Domain Models in SWIntO (Smart Web INTegrated Ontology),* In Journal of Web Semantics: Science, Services and Agents on the World Wide Web, iss. 5, p. 156, 2007.

[**21**] Wikipedia, *OWL,* site: http://en.wikipedia.org/wiki/Web_Ontolgy_Language

[**22**] DBGroup@unimo, Materiale didattico su OWL

[**23**] DBGroup@unimo, Materiale didattico su RDF

[**24**] Knublauch H., Fergerson R. W., Noy N. F., Musen M. A., *The Protégè OWL Plugin: An Open Development Environment for Semantic Web Applications*, In Third International Conference on the Semantic Web (ISWC-2004), Hiroshima, Japan.

[**25]** DBGroup@unimo, *MOMIS*, http://www.dbgroup.unimo.it/Momis/

[**26**] W3C, *DOLCE Lite +*, 5. *Towards Common Ontologies*, http://www.w3.org/2005/Incubator/eiif/XGR-framework-20090220/

[**27**] W3C, *DOLCE Lite +*, http://www.w3.org/2001/sw/BestPractices/WNET/DLP3941_daml.html

[**28**] Tesi Laurea VOD: Daniele Miselli: *Riscrittura di interrogazioni XML:un approccio basato sull'analisi semantic degli schemi*, Capitoli 4,6,7.

[**29**] Bergamaschi S., Po L., Sorrentino S., *Automatic annotation for mappings discovery in data integration systems*, In Proceedings of the Sixteenth Italian Symposium on Advanced Database Systems, SEBD 2008, 22-25 June 2008, Mondello, PA, Italy.

[**30**] MOMIS Demonstrator, Ontology Builder, ALA, Configure Algorithms, *Algorithm Description*.

[**31**] Wikipedia, *Precision e Recall,* site: http://it.wikipedia.org/wiki/Precisione_e_recall