Traduzione Handout Giunchilgia

Nicolas Torriglia Chat GPT

Contents

Chapter 1

Il mondo e la mente

1.1 Frames della mente

1.2 Illusioni ottiche

1.3 Fallacie mentali

1.4 Quindi?

Chapter 2

Rappresentazioni

2.1 Il gap semantico

2.2 Rappresentazioni mentali

2.3 Rappresentazioni

2.4 Esercizi

Chapter 3

Modelli e teorie asserzionali

L'osservazione 2.15 potrebbe suggerire che non esista una soluzione al problema della soggettività delle rappresentazioni mentali. Tuttavia, ciò non è vero. L'osservazione chiave è che le rappresentazioni sono costruite dagli esseri umani con lo scopo specifico di far convergere il più possibile le rappresentazioni mentali della stessa rappresentazione, minimizzando in particolare la probabilità di incongruenze. La domanda da affrontare è come costruire tali rappresentazioni.

3.1 Modelli

Il punto di partenza sono le rappresentazioni mentali analogiche, poiché le nostre rappresentazioni hanno inizio da qui. Considera l'esempio seguente.

Esempio 3.1 (Cosa c'è in una rappresentazione analogica) Considera la rappresentazione analogica illustrata nell'immagine della Figura 3.1. Possiamo vedere tre persone, che possiamo assumere abbiano i nomi Paolo, Stefania e Sofia, che sono amici, vari cani, il fatto che siano uno a destra dell'altro e, naturalmente, molto altro.

Figure 3.1: Una rappresentazione analogica di una situazione quotidiana

Osservazione 3.1 (Cosa c'è in una rappresentazione analogica) Considera la rappresentazione analogica illustrata nell'immagine della Figura 3.1. Possiamo vedere tre persone, che possiamo assumere abbiano i nomi Paolo, Stefania e Sofia, che sono amici, vari cani, il fatto che siano uno a destra dell'altro e, naturalmente, molto altro.

Intuizione 3.1 (Fatto) Un fatto `f` è un qualcosa che succede certamente ad una certa coordinata spaziotemporale

Definizione 3.1 (Modello) Un modello `M` è un set di fatti `F = {f}`

$$\texttt{M = \{f\}} \tag{3.1}$$

Osservazione 3.2 (Fatti e modelli) I fatti sono gli elementi atomici, non ulteriormente decomponibili, di un modello. Nota che, a differenza dei modelli, i fatti sono considerati come una nozione primitiva e quindi non possono essere definiti formalmente.

Esempio 3.2 (I fatti di un modello rappresentato nella Figura 3.1) Un modello, uno tra molti altri, della situazione rappresentata nella Figura 3.1 potrebbe contenere, ad esempio, i seguenti fatti:

```
Sofia is a person          Paolo is a man
Rocky is a dog             Sofia is near to Paolo
Sofia has blond hair       Sofia is a friend of Paolo
Rocky is an animal         Rocky is Sofia's dog
```

Osservazione 3.3 (La soggettività dei fatti) I fatti sono ciò che viene osservato e anche descritto, ad esempio, a terze persone. Il problema è che, proprio a causa di quanto discusso nella Sezione 2.2 e, nello specifico, il fatto di cosa siano i fatti è soggettivo e nascosto nelle menti delle persone che li percepiscono. Quanti altri fatti più e/o diversi da quelli elencati nell'Esempio 3.2 potresti immaginare? Potrebbero essere infiniti! Nota che ogni fatto può essere decomposto in qualsiasi insieme di fatti più semplici se questo è l'attuale focus dell'osservatore. Quindi, per esempio, anziché concentrarsi su Sofia, potrei concentrarmi sui suoi capelli, sulle sue gambe o su...

Osservazione 3.4 (Fatti mutualmente (in)coerenti in un modello) Il modello dell'Esempio 3.2 potrebbe essere esteso per affermare il fatto che Sofia è una donna o che Paolo è una persona. Tuttavia, avremmo problemi nell'estenderlo aggiungendo il fatto che Paolo è una donna o che Sofia è un cane, poiché avremmo due fatti mutualmente inconsistenti, qualcosa che sappiamo non può accadere nel mondo come lo percepiamo. Vedi anche l'Osservazione 2.9. Un modello non può contenere fatti che, almeno intuitivamente, sono mutualmente inconsistenti. Oltre a questo semplice esempio, la questione è come formalizzare questa intuizione e quindi come essere in grado di rilevarla mediante il ragionamento sui modelli.

Osservazione 3.5 (Fatti e affermazioni) Per essere considerato un fatto, deve essere descritto linguisticamente come tale. Non è per caso che nell'Esempio 3.2 abbiamo indicato i fatti attraverso un insieme di descrizioni in linguaggio naturale. Chiamiamo tali descrizioni "affermazioni". Il modo più semplice di pensare a un'affermazione è come a una frase dichiarativa in linguaggio naturale articolata in termini di un soggetto che si trova in una relazione più o meno complessa con un oggetto (come ad esempio, "Stefania sta camminando con i cani verso il centro della città"), o di un soggetto che detiene.

3.2 Teorie asserzionali

Osservazione 3.6 (Affermazioni e teorie assertive) Le affermazioni sono descrizioni indivisibili, diciamo atomiche, di un fatto. Le teorie assertive sono descrizioni di modelli.

Intuizione 3.2 (Affermazione) Un'affermazione a è una rappresentazione linguistica atomica di un certo fatto `f`.

Definizione 3.2 (Teoria assertiva) Una teoria assertiva $\mathcal{T}_A$ è un insieme di affermazioni.

$$\mathcal{T}_A = \{a\} \tag{3.2}$$

Abbiamo bisogno di definire che un'affermazione è la descrizione di un fatto specifico e, più in generale, che una teoria assertiva descrive un modello.

Esempio 3.3 (Una teoria assertiva del modello rappresentato nella Figura 3.1) Una teoria assertiva, una tra molte altre, che descrive i fatti dell'Esempio 3.2 in linguaggio naturale potrebbe essere, ad esempio:

`Sofia è una persona`	`Paolo è un uomo`
`Sofia è vicina a Paolo`	`Rocky è il cane di Sofia`
`Sofia è un'amica di Paolo`	`Sofia ha i capelli biondi`
`Rocky è un cane`	`Rocky è un animale`

Come si può notare, la teoria assertiva è in italiano, poiché essendo in linguaggio naturale può essere espressa anche in questo modo, e potrebbe essere altrettanto espressa in inglese.

3.3 Funzioni d'interpretazione

Definizione 3.3 (Funzione d'interpretazione) Sia $\mathcal{I}_A$ una funzione di interpretazione di una teoria assertiva, definita come:

$$\mathcal{I}_A : \mathcal{T}_A \rightarrow \mathtt{M} \tag{3.3}$$

Diciamo che un fatto $\mathtt{f} \in \mathtt{M}$ è l'interpretazione di $a \in \mathcal{I}_A$, e scriviamo:

$$\mathtt{f} = \mathcal{I}_A(a) = a^{\mathcal{I}_A} \tag{3.4}$$

per intendere che a è una descrizione linguistica di `f`. Diciamo che `f` è l'interpretazione di a, o, equivalentemente, che a denota `f`. (Più asserzioni possono portare ad un fatto)

Osservazione 3.7 (Funzione di interpretazione, polisemia) Si assume che $\mathcal{I}_A$ sia una funzione, ovvero per ogni fatto esiste solo un'affermazione che lo descrive. In effetti, dobbiamo garantire che se due fatti f_1 e f_2 sono diversi, allora non possono entrambi essere il risultato dell'interpretazione della stessa affermazione a, ovvero non può essere che se $\mathcal{T}_A = f_1$ allora anche $\mathcal{T}_A = f_2$. Questo fenomeno, chiamato polisemia, è diffuso nelle lingue naturali ed è una delle principali fonti di malintesi e, di conseguenza, di costruzione di rappresentazioni mentali divergenti della stessa rappresentazione. La polisemia delle affermazioni deriva direttamente dalla polisemia delle parole. Ad esempio: il nome proprio Java ha tre significati, è un linguaggio di programmazione, un tipo di chicchi di caffè e un'isola. La parola "auto" ha vari significati, ad esempio può significare automobile o una parte di un treno. Parole generali, come "fare", hanno più di dieci significati. La polisemia è comune alla maggior parte delle parole, in particolare con quelle parole che vengono più comunemente utilizzate (le persone tendono a dare alle parole il proprio significato specifico) ed è una delle principali complicazioni (se non l'unica) che sorgono durante la costruzione di sistemi di comprensione del linguaggio naturale.

Osservazione 3.8 (La non ambiguità delle funzioni di interpretazione) Come dalla Sezione 1.3, le descrizioni linguistiche sono ambigue. Come dall'Osservazione 3.7, una delle principali ragioni è la polisemia delle parole. Tuttavia, questa ambiguità è nella mente dell'ascoltatore/lettore. Si può assumere che chi parla/scrive abbia sempre in mente l'analoga rappresentazione unica che sta descrivendo. La nozione di funzione di interpretazione impone questa supposizione, obbligando chi parla/scrive a essere esplicito sul significato inteso.

Osservazione 3.9 (Funzione di interpretazione, sinonimia) Due affermazioni sono sinonimi quando hanno lo stesso significato, ovvero l'interpretazione di due affermazioni diverse a_1 e a : può denotare lo stesso fatto f, ovvero $\mathcal{I}_A(a_1) = I_A(a_2) = f$. Le parole sinonime sono nuovamente diffuse nelle lingue naturali, in particolare con le entità più comuni. Le persone e le entità in generale hanno più nomi, ad esempio nome, cognome, nome più cognome, soprannomi, che sono sinonimi. Diverse lingue generano diversi nomi per la stessa entità (ad esempio, Gran Bretagna, Great Britain). Ci sono anche sostantivi sinonimi, ad esempio auto e automobile. Nota come la parola "auto" sia sia polisemica che sinonima. Anche questo è abbastanza comune. In generale, la sinonimia non è un problema. Tuttavia, nei database relazionali, la sinonimia non è consentita, essenzialmente per motivi di efficienza. I database sono sviluppati sulla base dell'assunzione del nome unico, ovvero nei database, stringhe e affermazioni diverse significano sempre cose diverse.

Esempio 3.4 (Una funzione di interpretazione che fornisce un'interpretazione della teoria assertiva che descrive la Figura 3.1) La funzione di interpretazione naturale che interpreta le frasi nell'Esempio 3.3 (sinistra) nei fatti dell'Esempio 3.2 (destra) è la seguente:

$\mathcal{I}_A$(Sofia è una persona) = Sofia is a person
$\mathcal{I}_A$(Paolo è un uomo) = Paolo è un uomo
$\mathcal{I}_A$(Rocky è un cane) = Rocky è un cane
...

Osservazione 3.10 (Affermazioni e fatti, soggettività) Il problema della soggettività delle rappresentazioni rimane, essendo un fatto inevitabile della vita. Tuttavia, le nozioni di fatto, affermazione e funzione di interpretazione forniscono uno strumento. In primo luogo, si assume che i fatti siano descritti in modo inequivocabile, tramite funzioni di interpretazione, dalle affermazioni che, a loro volta, sono rappresentazioni linguistiche e, come tali, possono essere condivise. In secondo luogo, le affermazioni, anche se selezionate soggettivamente dagli esseri umani, si presume siano atomiche, cioè forniscono il livello minimo possibile di dettagli con cui un modello può essere descritto.

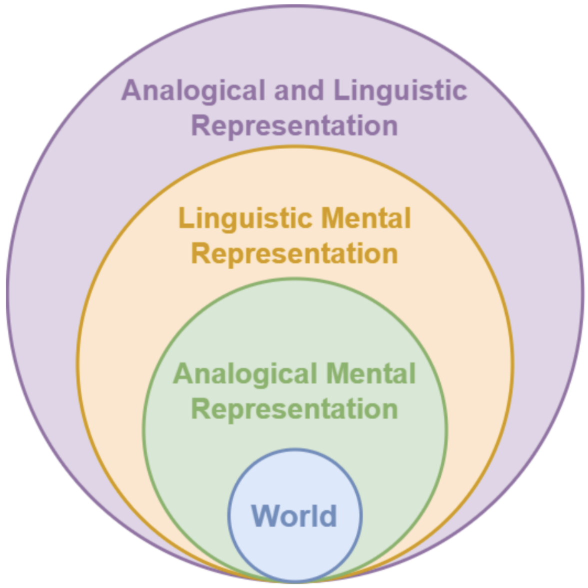


Figure 3.2: Diagramma di rappresentazione

Chapter 4

Modelli formali e teorie asserzionali

Per evitare un ragionamento fallace, è necessario rappresentare modelli e teorie assertive in modo inequivocabile, ovvero formale. Quattro sono le caratteristiche che ci interessano:

- **Formalità**: dovrebbe essere un linguaggio logico, ovvero con una sintassi e una semantica ben definite;
- **Universalità**: dovrebbe essere in grado di rappresentare tutti i tipi di fatti;
- **Intuitività**: dovrebbe consentire affermazioni i cui elementi di base (nomi di entità, concetti e proprietà) così come la loro struttura (ovvero come gli elementi di base sono collegati per costruire affermazioni) dovrebbero essere, da un lato, intuitivi per le persone mentre, dall'altro lato, avere una mappatura diretta alla struttura e all'organizzazione del dominio di riferimento;
- **Efficienza** computazionale: $\mathcal{L}_A$ dovrebbe consentire un motore di inferenza veloce ed efficiente, sfruttando l'efficienza intrinseca delle strutture dati utilizzate per memorizzare il modello del mondo.

Osservazione 4.1 (Tipi di linguaggi assertivi) I modelli ER e UML sono intuitivi ma non universali, rappresentano solo fatti di conoscenza. I database sono computazionalmente efficienti ma rappresentano solo fatti di dati. Il linguaggio naturale è universale ma la sua semantica non è formalmente definita e non è computazionalmente efficiente. Quest'ultima debolezza si estende anche a sottoinsiemi ben definiti delle lingue naturali. Come esempio di questo caso, i linguaggi logici sono universali con sintassi e semantica ben definite, ma non sono intuitivi da comprendere e anche computazionalmente inefficienti (vedi anche la Sezione 6.5).

Nella Sezione 4.1, introduciamo alcune definizioni di base della teoria degli insiemi, utili per definire D, mentre, nella Sezione 4.2, introduciamo alcune definizioni di base della teoria dei grafi utili per definire $\mathcal{L}_A$.

4.1 Teoria degli insiemi

4.1.1 Definizioni base

Possiamo definire gli insiemi in due modi:

- **Per elencazione**: L'insieme è descritto elencando tutti i suoi elementi (ad esempio, $A = \{a, e, i, o, u\}$).
- **Per astrazione**: L'insieme è descritto attraverso una proprietà dei suoi elementi (ad esempio, $A = \{x|x$è una vocale dell'alfabeto latino$\}$).

Abbiamo le seguenti definizioni di base.

Definizione 4.1 **(Insieme vuoto)** $\emptyset$ è l'insieme che non contiene alcun elemento.

Definizione 4.2 **(Appartenenza)** $a \in A$, l'elemento a appartiene all'insieme A.

Definizione 4.3 **(Non appartenenza)** $a \notin A$, l'elemento a non appartiene all'insieme A.

Definizione 4.4 **(Uguaglianza)** $A = B$, se e solo se A e B contengono gli stessi elementi.

Definizione 4.5 **(Disuguaglianza)** $A \neq B$, se e solo se non è vero che $A = B$.

Definizione 4.6 **(Sottoinsieme)** $A \subseteq B$, se e solo se tutti gli elementi in A appartengono anche a B.

Definizione 4.7 **(Sottoinsieme proprio)** $A \subset B$, se e solo $\subseteq B$ e $A \neq B$.

Definizione 4.8 **(Insieme universale)** L'insieme universale è l'insieme di tutti gli elementi o membri di tutti gli insiemi correlati e è denotato dalla lettera $\mathcal{U}$

Utilizziamo i **diagrammi di Venn** per rappresentare gli insiemi. I diagrammi di Venn consistono in cerchi sovrapposti o intersecati che rappresentano insiemi e le loro relazioni. Ogni cerchio rappresenta un insieme specifico, e l'area in cui i cerchi si sovrappongono rappresenta gli elementi condivisi tra gli insiemi corrispondenti. Un elemento che non appartiene a un insieme è rappresentato come un punto al di fuori del cerchio che rappresenta l'insieme.

Definizione 4.9 **(Unione)** Dati due insiemi A e B, l'unione di A e B è definita come l'insieme che contiene gli elementi appartenenti a A o a B o ad entrambi, e viene denotata con $\mathbf{A} \cup \mathbf{B}$

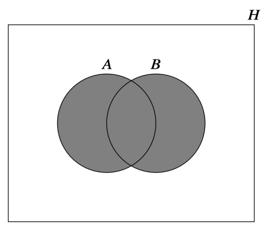


Figure 4.1: Operazione di unione

Definizione 4.10 **(Intersezione)** Dati due insiemi A e B, l'intersezione di A e B è definita come l'insieme che contiene gli elementi appartenenti sia a A sia a B, e viene denotata con $\mathbf{A} \cap \mathbf{B}$

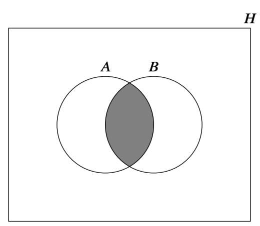


Figure 4.2: Operazione di intersezione

Definizione 4.11 **(Differenza)** La differenza tra due insiemi A e B, è definita come l'insieme che contiene gli elementi che sono membri di A, ma non membri di B. Viene denotata con $\mathbf{A} \setminus \mathbf{B}$

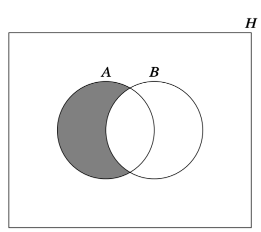


Figure 4.3: Operazione di differenza

Definizione 4.12 (Differenza complementare) Dato un insieme universale U e un insieme A, dove $A \subseteq U$, il complemento di A in U è definito come l'insieme che contiene tutti gli elementi in U che non appartengono ad A e viene denotato con $\mathbf{A^c}$ o $\mathbf{U \setminus A}$

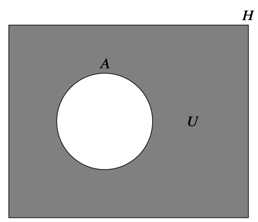


Figure 4.4: Differenza complementare

Teorema 4.1 (Proprietà delle operazioni)

- **Con lo stesso set**
  - $A \cap A = A$
  - $A \cup A = A$
- **Commutativa**
  - $A \cap B = B \cap A$
  - $A \cup B = B \cup A$
- **Insieme vuoto**
  - $A \cap \emptyset = \emptyset$
  - $A \cup \emptyset = A$
- **Associative**
  - $(A \cap B) \cap C = A \cap (B \cap C)$
  - $A \cap (B \cup C) = (A \cup B) \cap (A \cup C)$
- **Leggi di De Morgan**
  - $\overline{A \cup B} = \overline{A} \cap \overline{B}$
  - $\overline{A \cap B} = \overline{A} \cup \overline{B}$

4.1.2 Relazioni

4.1.3 Funzioni

4.2 Teorie dei grafi

4.2.1 Nozioni base

4.2.2 Grafi ...

4.3 Esercizi

Chapter 5

Modello del Mondo - Rappresentazione Estensionale

Le teorie assertive e i modelli rappresentano un passo importante avanti, ma ci sono ancora tre limitazioni principali:

- Stiamo considerando solo i fatti del modello in esame. Cosa succede con i fatti che possono verificarsi negli altri possibili modelli, descrivendo situazioni possibilmente molto diverse?
- Il linguaggio utilizzato è molto limitato e consiste solo nell'insieme di affermazioni che descrivono i fatti del modello in esame. Cosa succede alle affermazioni che descrivono fatti in tutti gli altri modelli?
- Come conseguenza dei due fatti sopra, la definizione di funzione di interpretazione non è generale. Una nuova e diversa funzione di interpretazione deve essere definita per ogni nuova situazione.

5.1 Dominio

Secondo l'Equazione (3.1), un modello `M` è deifinito come `M = {f}`, dove `f` è un insieme di fatti che sono veri in una determinata situazione.

Definizione 5.1 Dominio (di interpretazione) Un dominio (di interpretazione) è un insieme di fatti `{f}`

$$\mathtt{D} = \mathtt{\{F\}} \tag{5.1}$$

Definizione 5.2 (Modello) Dato un dominio `D`, un modello `M` è un sott'insieme di `D`

$$\mathtt{M} = \mathtt{\{f\}} \subseteq D \tag{5.2}$$

Osservazione 5.1 (Dominio, modello) Un dominio è l'insieme di tutti i fatti che siamo disposti a considerare. Un modello è semplicemente il sottoinsieme di fatti che definiamo come rappresentativi di ciò che è il caso nella situazione attuale. La Definizione 5.2 generalizza in modo ovvio la Definizione 3.1.

Esempio 5.1 (I fatti del dominio rappresentato in Figura 3.1) Un dominio che raccoglie, tra gli altri, i fatti rappresentati nella Figura 3.1 potrebbe contenere, ad esempio, i seguenti fatti:

```
Sofia is a person       Sofia is a woman
Paolo is a man          Paolo is a person
Paolo is a dog          Rocky is a dog
Sofia is near Paolo     Rocky is the dog of Sofia
Rocky is the dog of Paolo . . .
```

... e molti altri

Osservazione 5.2 (Dominio) Mentre un modello è l'insieme di fatti che sono veri in una certa situazione, un dominio consiste nell'insieme di fatti che potenzialmente possono essere veri per tutte le possibili situazioni. Un dominio definisce tutto e solo ciò che può essere potenzialmente percepito.

Osservazione 5.3 (Fatti mutuamente inconsistenti in un dominio) Come dall'Esempio 5.1, un dominio, a differenza di un modello, può contenere fatti che, almeno intuitivamente, sono mutuamente inconsistenti. Dato un dominio, ci sono molti modelli potenziali, alcuni dei quali potenzialmente mutuamente inconsistenti. I domini devono consentire la possibile istanziazione di modelli distinti mutuamente inconsistenti, come avviene normalmente nel mondo.

5.2 Linguaggio asserzionale

Secondo l'Equazione (3.2), una teoria assertiva $\mathcal{T}_A$ è definita come $\mathcal{T}_A = \{a\}$, dove $\{a\}$ è un insieme di affermazioni che descrivono i fatti che sono veri in una determinata situazione.

Definizione 5.3 (Linguaggio assertivo) Un linguaggio assertivo $\mathcal{L}_A$ è un insieme di affermazioni $\{a\}$

$$\mathcal{L}_A = \{a\} \tag{5.3}$$

Definizione 5.4 (Teoria asserzionale) Data un linguaggio assertivo $\mathcal{L}_A$, una teoria assertvia $\mathcal{T}_A$ è un sottonsieme di $\mathcal{L}_A$

$$\mathcal{T}_A = \{a\} \subseteq \mathcal{L}_A \tag{5.4}$$

Osservazione 5.4 (Linguaggio assertivo) Mentre una teoria assertiva è l'insieme di affermazioni che descrive ciò che è vero in un certo modello, un linguaggio assertivo consiste nell'insieme di affermazioni che descrivono tutti i fatti che possono potenzialmente verificarsi. La Definizione 5.4 si estende in modo ovvio dalla Definizione (3.2).

Esempio 5.2 (Linguaggio assertivo) Il linguaggio dell'Esempio 3.3 esteso per includere tutti i fatti del dominio definito nell'Esempio 5.1 allo stesso modo (ossia, traduzioni italiane citate della frase in inglese che descrive un fatto) è un linguaggio assertivo.

Osservazione 5.5 (Completezza e correttezza di un linguaggio assertivo $\mathcal{L}_A$ rispetto a un dominio D) Un linguaggio assertivo non è necessariamente completo, ovvero non contiene necessariamente affermazioni per tutti i fatti in un dominio (che, tra le altre cose, sono in principio infiniti). La caratteristica chiave è che dovrebbe contenere tutte le affermazioni ritenute rilevanti. Viceversa, un linguaggio assertivo è richiesto di essere corretto, cioè di contenere solo affermazioni che denotano fatti nel dominio di riferimento, al fine di evitare affermazioni prive di senso.

Esempio 5.3 (Linguaggi assertivi):

1. Linguaggi che consentono solo affermazioni in linguaggio naturale della forma "<soggetto> <verbo> <oggetto>" descrivono fatti sul mondo. Il linguaggio utilizzato è una sequenza di affermazioni semplici senza frasi complesse.

2. I database relazionali (DB) descrivono fatti sul mondo. Il linguaggio utilizzato per descrivere i contenuti di un database relazionale sono le tabelle;

3. I modelli entity-relationship (ER) descrivono fatti generali sui contenuti dei database. Sono scritti utilizzando il linguaggio del diagramma ER, un linguaggio di grafi specificamente etichettato.

5.3 Funzione d'interpretazione

Definizione 5.5 (Funzione di interpretazione) Sia $\mathcal{L}_A$ un linguaggio di asserzioni e $\mathtt{D}$ un dominio. Quindi, una Funzione di Interpretazione di un linguaggio assertivo $\mathcal{I}_A$ è definita come:

$$\mathcal{I}_A : \mathcal{L}_A \to \mathtt{D} \quad (\mathcal{I}_A \subseteq \mathcal{L}_A \times \mathtt{D}) \tag{5.5}$$

Diciamo che un fatto $f \in M$ è l'interpretazione di $a \in I_A$, e scriviamo:

$$f = I_A(a) = a^{I_A} \tag{5.6}$$

Per indicare che a è una descrizione linguistica di f, diciamo che f è l'interpretazione di a, o equivalentemente, che a denota f.

Osservazione 5.6 (Funzione di interpretazione) La Definizione 5.5 si generalizza secondo le necessità e sostituisce la Definizione 3.3.

Osservazione 5.7 (Funzione di interpretazione, non ambiguità e sinonimia) Come dall'Osservazione 3.7, 3.8, 3.9, le funzioni di interpretazione, essendo funzioni, non sono ambigue e non consentono affermazioni e parole polisemiche, ma permettono la sinonimia.

Osservazione 5.8 (Funzione di interpretazione, totalità) Le funzioni di interpretazione sono totali. Ciò garantisce che ogni elemento del linguaggio abbia un'interpretazione.

Osservazione 5.9 (Funzione di interpretazione, non suriettività) Le funzioni di interpretazione non sono necessariamente suriettive. In altre parole, se $I_A : L_A \to D$, L_A potrebbe non essere in grado di denominare tutti i fatti in D. Questa proprietà è utile con domini infiniti o quando non si è interessati a menzionare tutti i fatti conosciuti. Vedi anche l'Osservazione 5.5.

Esempio 5.4 (Funzione di interpretazione) Prendendo come esempio 5.2, consideriamo L_A che contiene le affermazioni dell'Esempio 3.3, estese per descrivere tutti i fatti nel dominio D definito nell'Esempio 5.1. Prendiamo il dominio D definito nell'Esempio 5.1. Quindi, una funzione di interpretazione $I_A : L_A \to D$ può essere costruita estendendo in modo ovvio la funzione di interpretazione definita nell'Esempio 3.4.

5.4 Modello del mondo

Il viaggio è completo. Dobbiamo soltanto mettere tutto insieme.

Osservazione 5.10 (I ruoli di D,$\mathcal{L}, \mathcal{I}_A, M, \mathcal{T}_A$) Le definizioni fornite nelle sezioni precedenti possono essere riassunte dalla seguente figura.

$$\begin{tikzcd}
a \arrow[d, "\mathcal{I}_A"] & \mathcal{T}_\mathcal{A} \arrow[l, "\in"'] \arrow[r, "\subseteq"] \arrow[d, dashed, "\mathcal{I}_A"] & \mathcal{L}_A \arrow[d, "\mathcal{I}_A"] \\
\mathtt{f} & \mathtt{M} \arrow[l, "\in"'] \arrow[r, "\subseteq"] & \mathtt{D}
\end{tikzcd} \tag{5.7}$$

Nell'Equazione (5.7), $\mathtt{D}$ definisce l'insieme di fatti f di potenziale interesse, $\mathtt{M}$ l'insieme di fatti su cui ci stiamo concentrando, $\mathcal{L}_A$ l'insieme di affermazioni a di potenziale interesse e, infine, $\mathcal{T}_A$ è la teoria che descrive $\mathtt{M}$. In altre parole, come singole affermazioni a descrivono singoli fatti f da un lato, e come gli insiemi complessivi di affermazioni $\mathcal{L}_A$ descrivono $\mathtt{D}$ dall'altro lato, definiscono l'ambito entro il quale una teoria assertiva $\mathcal{T}_A$ può concentrarsi per descrivere modelli specifici $\mathtt{M}$. Affermazioni, linguaggi e teorie sono semplicemente i mezzi per specificare il modello desiderato enunciando i fatti desiderati tra quelli consentiti dal dominio di riferimento.

Definizione 5.6 (Modello del mondo)

$$\hat{\mathcal{W}} = \langle \mathcal{L}_A, \mathtt{D}, \mathcal{I}_A \rangle \tag{5.8}$$

è un **modello del mondo**

Osservazione 5.11 (Modello del mondo) Nell'Equazione (5.7), i componenti di un modello del mondo, cioè $\mathcal{L}_A, \mathtt{D}, \mathcal{I}_A$ definiscono le regole generali che vengono seguite durante la costruzione di una rappresentazione. Sono definiti a priori, di solito da esperti di modellizzazione e rappresentazione della conoscenza, come strumenti generali da utilizzare da parte degli operatori. Forniscono l'infrastruttura generale di modellazione che consente di rappresentare problemi del mondo reale. Forniscono anche un quadro uniforme con il quale è possibile confrontare e, eventualmente, anche unire due rappresentazioni. Gli sviluppatori di software di solito studiano questi modelli durante alcuni corsi di informatica o intelligenza artificiale e li utilizzano così come sono nello sviluppo di sistemi; si pensi, ad esempio, all'ampio utilizzo dei modelli ER (Entity-Relationship).

Osservazione 5.12 (Dalle rappresentazioni mentali ai modelli del mondo) I modelli del mondo sono spazi di rappresentazioni possibili, cioè teorie, progettati per minimizzare la possibilità di diverse rappresentazioni mentali della stessa teoria e del corrispondente modello che raffigura il mondo.

I modelli del mondo forniscono il quadro generale entro il quale le teorie assertive e i modelli possono essere definiti e confrontati.

Osservazione 5.13 (Teoria e modello) Ricorda che, dato un modello di un mondo $\hat{\mathcal{W}} = \langle \mathcal{L}_A, \mathtt{D}, \mathcal{I}_A \rangle$, abbiamo che $\mathtt{M} = \mathtt{\{F\}} \subseteq \mathtt{D}$ (equazione 5.1) e $\mathcal{T}_A = \{a\} \subseteq \mathcal{L}_A$ (equzione 5.3)

Osservazione 5.14 (Definire un modello attraverso una teoria) Il modo più comune per modellare il mondo è definire un insieme di affermazioni, ciò che chiamiamo una teoria. In altre parole, costruiamo un modello $\mathtt{M}$ selezionando qualsiasi sottoinsieme $\mathcal{T}_A$ di $\mathcal{L}_A$. Questo è l'approccio comune quando il compito è rappresentare da zero una parte data del mondo che è di interesse.

Tuttavia, a volte, viene fornita una teoria predefinita $\mathcal{T}_A$ e un modello predefinito $\mathtt{M}$ e si chiede come sono correlati. In tal caso, abbiamo quanto segue.

Definizione 5.7 Correttezza e completezza di una teoria assertiva $\mathcal{T}_A$ rispetto a un modello M Sia $\hat{\mathcal{W}} = \langle \mathcal{L}_A, \mathtt{D}, \mathcal{I}_A \rangle$ il modello di un mondo. Siano $\mathcal{T}_A \subseteq \mathcal{L}_A$ e $\mathtt{M} \subseteq \mathtt{D}$ rispettivamente una teoria assertiva e un modello. Allora abbiamo due situazioni possibili, come segue:

- **Correttezza**. Sia $a \in \mathcal{L}_A$ un'asserzione. Se per tutte le a, se $a \in \mathcal{T}_A$ allora $\mathcal{I}_A(a) = \texttt{f} \in \texttt{M}$ diciamo che $\mathcal{T}_A$ è corretta rispetto a M, o che M è un modello per AL.

- **Completezza**. Sia $\texttt{f} \in \texttt{M}$ un fatto. Se, per ogni f, se $\texttt{f} \in \texttt{M}$ allora esiste un'asserzione $a \in \mathcal{T}_A$ tale che $\mathcal{I}_A(a) = \texttt{f}$ diciamo che $\mathcal{T}_A$ è completa rispetto a M.

Le nozioni di **scorrettezza** e **incompletezza** sono definite in modo ovvio.

Osservazione 5.15 (Correttezza e completezza di una teoria assertiva $\mathcal{T}_A$ rispetto a un modello M) Una teoria assertiva potrebbe non essere completa, ovvero potrebbero esistere fatti nel dominio per i quali non contiene affermazioni. Descrizioni incomplete dei modelli sono la norma, a causa dell'ignoranza o anche per mancanza di interesse. Tuttavia, una teoria $\mathcal{T}_A$ deve contenere solo affermazioni riguardanti i fatti in M per M appartenente a $\mathcal{T}_A$. Questo al fine di evitare informazioni errate. Notare come questo requisito sia opposto a quello su linguaggi e domini come indicato nell'Osservazione 5.5.

Esempio 5.5 (Correttezza e completezza di una teoria assertiva $\mathcal{T}_A$ rispetto a un modello M) Considera l'Esempio 3.4. Una teoria assertiva $\mathcal{T}_A$ che contiene tutte e solo le affermazioni del dominio della funzione di interpretazione nell'Esempio 3.4 è corretta e completa rispetto al modello M che contiene tutti e solo i fatti che sono nel dominio della stessa funzione di interpretazione. Qualsiasi teoria assertiva che è un sottoinsieme di $\mathcal{T}_A$ è corretta rispetto a M. M non è un modello di alcun superset di $\mathcal{T}_A$.

Chapter 6

Modello del Mondo - Rappresentazione Intenzionale

I modelli del mondo ci permettono di sfruttare fatti e affermazioni su di essi come i principali componenti necessari per costruire una rappresentazione del mondo, da utilizzare successivamente per risolvere problemi.

Osservazione 6.1 (Modello del mondo, rappresentazione estensionale) I modelli del mondo, come dalla Definizione 5.6, sono rappresentazioni estensionali del mondo, ovvero sono definite come insiemi di affermazioni a e fatti $\mathtt{f}$, oltre a una funzione di interpretazione $\mathcal{I}_A$ che consente di definire quali affermazioni denotano quali fatti in uno o più modelli di riferimento.

Ma cosa si intende per "fatto"? Come costruiamo affermazioni sui fatti? La risposta a questa domanda richiede la definizione di una rappresentazione intenzionale dei modelli del mondo, ovvero i meccanismi di rappresentazione che consentono di costruire affermazioni e fatti a partire da un insieme finito di elementi componenti primitivi. In seguito, ogni volta che è necessario per evitare confusione, adottiamo la seguente notazione e terminologia.

Notazione 6.1 (Rappresentazione estensionale e intenzionale di un insieme) Sia $\mathcal{S}$ un insieme. Allora con:

- $\mathcal{S}^e$ intendiamo la rappresentazione estensionale di $\mathcal{S}$, ovvero come un insieme di elementi (ad esempio, fatti, affermazioni, ma non solo);
- $\mathcal{S}^i$ intendiamo la rappresentazione intenzionale di $\mathcal{S}$, in cui gli elementi di $\mathcal{S}^e$ sono definiti intenzionalmente, a partire da un insieme di componenti primitivi.

Gli apici vengono omessi quando non sorgono confusioni.

6.1 Dominio

L'Esempio 3.1, l'Osservazione 3.1 e anche l'Osservazione 3.5 forniscono indicazioni su come costruire una rappresentazione intenzionale dei fatti.

Intuizione 6.1 (Dominio, rappresentazione intenzionale) La rappresentazione intenzionale di un dominio è composta da tre componenti, come segue:

- *entità*, associate agli elementi della rappresentazione che possono essere isolati e distinti dal resto;
- *classi* (insiemi) di entità, caratterizzate dal fatto di avere alcune caratteristiche comuni non condivise dalle entità degli altri insiemi;
- *relazioni* tra entità, che raccolgono più entità che condividono una proprietà comune.

Definizione 6.1 (Dominio, rappresentazione intenzionale) La rappresentazione intenzionale $\mathtt{D}^i$ di un dominio $\mathtt{D}$ è definita come:

$$\mathtt{D}^i = \langle E, \{C\}, \{R\} \rangle \tag{6.1}$$

con

$$\begin{gathered} E = \{e\} \\ C \subseteq E \\ R \subseteq E \times \cdots \times E \text{ n times} \end{gathered}$$

dove $E = \{e\}$ è un set di **identità**, $\{C\}$ è un set di **classi** di identità, $\{R\}$ p un insieme di relazione n-arie R^n per qualche n. E viene chiamato l'universo di D^i o anche **l'universo dell'interpretazione**.

Definizione 6.2 (Fatto, rappresentazione intensionale) La rappresentazione intensionale di un fatto $\mathtt{f}$ assume una delle quattro seguenti forme.

$$\begin{aligned} &\mathtt{e} && \in \mathtt{C} \\ &\langle \mathtt{e}_1, \ldots, \mathtt{e}_n \rangle && \in \mathtt{R} \\ &\mathtt{C} && \subseteq \mathtt{E} \\ &\mathtt{R}^n && \subseteq \mathtt{C}_1 \times \cdots \times \mathtt{C}_n \end{aligned} \tag{6.2}$$

con $\mathtt{e},\ \mathtt{e}_i \in \mathtt{E}$ e $\mathtt{C},\ \mathtt{C}_i \subseteq \mathtt{E}$.

Osservazione 6.2 (Fatto, rappresentazione intensionale) L'intuizione dietro la Definizione 6.2 è la seguente. $e \in C$ significa che una certa entità e appartiene a una certa classe C, come nel caso dell'affermazione che Sofia è una persona. $< e_1, \ldots, e_n > \in R$ significa che n entità si trovano in una certa relazione, come nel caso dell'affermazione che Paolo si trova tra Sofia e Stefania. $C \subseteq E$ significa che C, ad esempio "persona", è una classe. $R_n \subseteq C_1 \times \ldots \times C_n$ significa che una relazione si applica solo a classi specifiche, come nel caso dell'affermazione che le persone sono amiche degli animali.

Osservazione 6.3 (Complessità dei fatti) Si potrebbe pensare a fatti più sofisticati, ad esempio il fatto $C_1 \subseteq C_2$. Ciò è naturalmente possibile, a costo di complicare il linguaggio assertivo, con un limite superiore nell'intuitività del modello di mondo risultante. Qui stiamo definendo il modello di mondo più semplice possibile. Tuttavia, i linguaggi di rappresentazione e le logiche possono comunque essere utilizzati per aumentare l'espressività del linguaggio di rappresentazione.

Esempio 6.1 (Fatto, rappresentazione intensionale ed estensionale) Consideriamo l'Esempio 5.1. Il fatto che Sofia sia una persona è costruito prendendo l'entità Sofia e affermando che essa è una persona. Il fatto che Sofia sia vicina a Paolo è costruito affermando che le entità Sofia e Paolo stanno in una relazione in cui una è vicina all'altra.

Definizione 6.3 (Dominio, rappresentazione estensionale) La rappresentazione estensionale D_e di un dominio D, la cui rappresentazione intensionale D_i è come da Definizione 6.1, è $D_e = \{f\}$ con f come da Definizione 6.2.

Osservazione 6.4 (Entità) Ma cosa si intende per entità? L'idea generale è che un'entità è qualcosa che può essere percepito e quindi rappresentato in una rappresentazione analogica di (una parte del) mondo. È un'entità un oggetto specifico? O una persona? O un animale? Sì, ma anche qualsiasi cosa che avvenga nel tempo, ad esempio un evento o un processo. Si assume che le entità abbiano nomi. Esempi di (nomi di) entità: Federico, Pussy, Garfield, Trento, l'ultimo Campionato del Mondo di calcio, e così via.

Intuizione 6.2 (Entità) Un'entità è qualsiasi cosa che può essere rappresentata in una rappresentazione (analogica o linguistica) del mondo e che ha un nome. Un'entità può essere rappresentata nella rappresentazione mentale del mondo ogni volta che è percepita o il suo nome è udito da un essere umano.

Intuizione 6.3 (Nome) Un nome è una stringa, scritta in qualche lingua, che consente di fare riferimento alle entità nelle rappresentazioni e, più specificamente, nei modelli di mondo.

Intuizione 6.4 (Entità nominata) Il vigneto di fronte a me non è un'entità (per me). Anche se posso vederlo chiaramente, non ho un nome che posso usare per fare riferimento ad esso, ad esempio quando parlo con gli altri. Per essere considerate entità, devono essere entità nominate.

Osservazione 6.5 (Dominio, rappresentazione intensionale) Modelliamo intenzionalmente il mondo intorno a noi in termini di entità. Le entità, a loro volta, sono raggruppate in classi specifiche, in base a determinate loro proprietà. Quindi abbiamo, ad esempio, le classi: umano, persona, donna, animale, cane, macchina, penna, computer, e così via. Le entità possono essere messe in relazione tra loro in base alla loro classe. Ad esempio: gli esseri umani sono amici degli esseri umani, o degli animali, mangiano cibo, una macchina è vicina a un semaforo e così via.

Osservazione 6.6 (Dominio, parzialità) Un dominio non deve contenere tutti e cinque i tipi di fatti elencati nella Definizione 6.2. La selezione dei fatti da includere dipende da ciò che si sta modellando. Grossolanamente, i domini possono essere suddivisi in due tipi. Il primo è il dominio dei dati, così chiamato perché descrive ciò che è percepibile e coinvolge i primi due tipi di fatti nella Definizione 6.2. Il secondo tipo è il dominio della conoscenza, così chiamato perché si concentra su affermazioni generali, non direttamente percebili, e descrive relazioni generali tra classi e relazioni. I domini della conoscenza coinvolgono gli ultimi tre tipi di affermazioni. Esistono anche quelli che qui chiamiamo domini misti, ovvero domini che coinvolgono tutti i tipi di fatti. I domini misti sono molto utili nella comunicazione, ad esempio nella comunicazione in linguaggio naturale tra le persone, poiché consentono di esprimere uniformemente la conoscenza sulle entità, le classi e le relazioni.

Definizione 6.4 (Dominio, dati, conoscenza, misto) Un dominio dei dati contiene solo fatti della forma $e \in C$ e $< e_1, \ldots, e_n > \in R$. Un dominio della conoscenza contiene solo fatti della forma $C_1 \subseteq E$ e $R_n \subseteq C_1 \times \ldots \times C_n$. Un dominio misto contiene tutti i tipi di fatti.

Esempio 6.2 (Dominio dei dati) Il dominio descritto nell'Esempio 5.1 è un dominio dei dati. Alcuni dei suoi fatti possono essere rappresentati intenzionalmente come segue:

$$\begin{aligned}
\text{sofia} &\in \text{Person} \\
< \text{rocky}, \text{paolo} > &\in \text{DogOf} \\
\text{sofia} &\in \text{Woman} \\
\text{paolo} &\in \text{Dog} \\
< \text{paolo}, \text{rocky} > &\in \text{HasDog} \\
\text{rocky} &\in \text{Dog} \\
< \text{sofia}, \text{paolo} > &\in \text{Near} \\
< \text{rocky}, \text{sofia} > &\in \text{DogOf} \\
\text{paolo} &\in \text{Man} \\
< \text{sofia}, \text{paolo} > &\in \text{FriendOf} \\
< \text{paolo}, \text{sofia}, \text{stefania} > &\in \text{Between}
\end{aligned}$$

Nella notazione, le entità sono scritte con le lettere iniziali in minuscolo, mentre nei concetti e nelle proprietà sono in maiuscolo.

Esempio 6.3 (Dominio della conoscenza) Un dominio della conoscenza che descrive la conoscenza alla base del dominio dei dati nell'Esempio 5.1 è un dominio dei dati che potrebbe contenere, ad esempio, i seguenti fatti:

$$
\begin{aligned}
\text{Person} &\subseteq \text{Entity} \\
\text{HasDog} &\subseteq \text{Person} \times \text{Dog} \\
\text{Dog} &\subseteq \text{Entity} \\
\text{DogOf} &\subseteq \text{Dog} \times \text{Person} \\
\text{Animal} &\subseteq \text{Entity} \\
\text{FriendOf2} &\subseteq \text{person} \times \text{person} \times \text{person} \\
\text{Near} &\subseteq \text{Entity} \times \text{Entity} \\
\text{FatherOf} &\subseteq \text{Person} \times \text{Person} \\
\text{ChildOf} &\subseteq \text{Person} \times \text{Person}
\end{aligned}
$$

Dove "Entity" sta per E.

Esempio 6.4 (Dominio misto) Un esempio di dominio misto può essere creato unendo i domini dell'Esempio 6.2 e 6.3.

6.2 Linguaggi assertivi

Definizione 6.5 (Linguaggio assertivo, rappresentazione intensionale) La rappresentazione intensionale L_{iA} di un linguaggio assertivo L_A è definita come

$$L_A^i = \langle E, \{C\}, \{P\} \rangle \tag{6.3}$$

dove $E = \{e\}$ è un insieme di (nomi di) entità, $\{C\}$ è un insieme di concetti, dove un concetto è un nome di una classe, e $\{P\}$ è un insieme di proprietà, dove una proprietà è un nome di una relazione.

Definizione 6.6 (Linguaggio assertivo, rappresentazione estensionale) La rappresentazione estensionale L_A^e di un linguaggio assertivo L_A, la cui rappresentazione intensionale L_A^i è come da Definizione 6.5, è $L_A^e = \{a\}$ con a che ha una delle seguenti quattro forme.

$$
\begin{aligned}
&C(e) \\
&\mathcal{P}_n(e_1, \ldots, e_n) \\
&C \\
&\mathcal{P}_n(C_1, \ldots, C_n)
\end{aligned}
\tag{6.4}
$$

Dove: $C(e)$ dovrebbe essere letto come l'entità di nome e appartenente alla classe di nome C, e $P(e_1, \ldots, e_n)$ come le entità di nome $e_1, \ldots, e_n$ coinvolte nella relazione di nome P.

Osservazione 6.7 (Sintassi astratta e concreta) Le affermazioni nella Definizione 6.6 devono essere lette come sintassi astratta, anziché sintassi concreta. Devono essere prese come segnaposti per molteplici modi di descrivere lo stesso fatto in diverse lingue assertive. Quindi, ad esempio, l'asserzione di sintassi astratta Person(Sofia) è la rappresentazione di sintassi astratta di molteplici rappresentazioni di sintassi concrete, come ad esempio Person(Sofia), Sofia è una persona, P(S), (Sofia)Person (notazione postfix) e, soprattutto, varie tipologie di notazioni grafiche (vedi sotto). Ciò che conferisce significato alle affermazioni è il loro significato inteso, come definito dalla funzione di interpretazione (vedi sotto). In concreto, la sintassi utilizzata nelle prime due affermazioni proviene dalla logica $\mathcal{LOE}$, mentre quella delle ultime tre proviene dalla logica $\mathcal{LOD}$.

Terminologia 6.1 (Alfabeto) La rappresentazione intenzionale, come definita nella Definizione 6.6, $\mathcal{L}^i_A$, di un linguaggio, in questo caso $\mathcal{L}_A$, definito estensionalmente, in questo caso $\mathcal{L}^e_A$, è anche chiamata l'alfabeto (di quel linguaggio).

Osservazione 6.8 (Alfabeto) Le affermazioni descrivono fatti. Il primo passo è associare a ciascun elemento del dominio un nome nell'alfabeto del linguaggio assertivo. Successivamente, le affermazioni vengono costruite compositamente seguendo come sono costruiti i fatti. Ciò crea una corrispondenza uno-a-uno tra i fatti e le affermazioni, che suggerisce immediatamente il fatto descritto da un'affermazione.

Osservazione 6.9 (Linguaggio assertivo, parzialità) L'Osservazione 6.6 si estende in modo ovvio ai linguaggi assertivi.

Definizione 6.7 (Linguaggio assertivo, dati, conoscenza, misto) Un linguaggio assertivo di dati contiene solo affermazioni della forma $C(e)$ e $P(e1, \ldots, en)$. Un linguaggio assertivo di conoscenza contiene solo fatti della forma C e $P_n(C1, \ldots, Cn)$. Un linguaggio assertivo misto contiene entrambi i tipi di affermazioni.

Esempio 6.5 (Un linguaggio assertivo di dati che descrive il dominio dei dati nell'Esempio 6.2) Alcune affermazioni sono le seguenti:

Person(sofia)	Woman(sofia)	Man(paolo)
Person(paolo)	Dog(paolo)	Dog(rocky)
Near(sofia, paolo)	DogOf(rocky, sofia)	FriendOf(sofia, paolo)
DogOf(rocky, paolo)	HasDog(sofia, rocky)	HasFriend(paolo, sofia)

Esempio 6.6 (Un linguaggio assertivo di conoscenza che descrive il dominio della conoscenza nell'Esempio 6.3) Alcune affermazioni sono le seguenti:

Person	HasDog(Entity,Entity)	Woman
Dog	FriendOf2(Man,Man,man)	Animal
Near(Entity, Entity)	HasFriend(Man,Dog)	FriendOf1(Person, Person)

dove "Entity" è il concetto che denota l'insieme Entity di tutte le entità nel dominio.

Esempio 6.7 (Un linguaggio assertivo misto che descrive il dominio misto negli Esempi 6.2, 6.3) Alcune affermazioni sono le seguenti:

Person	Near(rocky, paolo)	Person(paolo)
FriendOf1(Person, Person)	FriendOf1(sofia, paolo)	Dog
Dog(rocky)	Near(Entity, Entity)	Near(paolo, sofia)

Le affermazioni di conoscenza implicano affermazioni sui dati. Pertanto, ad esempio, si potrebbe dedurre che Sofia è una persona dal fatto che è un amico di un'altra persona. Questo richiede ragionamento, cioè ciò per cui servono le logiche.

Esempio 6.8 (Un linguaggio assertivo di conoscenza, utilizzando modelli ER, che descrive il dominio della conoscenza nell'Esempio 6.3) Consideriamo l'Esempio 6.6: concentriamoci sui seguenti concetti e ruoli:

Person	Animal	HasAnimal(Person, Animal)
OwnedBy(Animal, Person)	Woman	Man
Dog	HasDog(Man, Dog)	HasDog(Women, Dog)
OwnedBy(Dog, Man)	OwnedBy(Dog, Woman)	

Here is a sample ER Model describing our example:

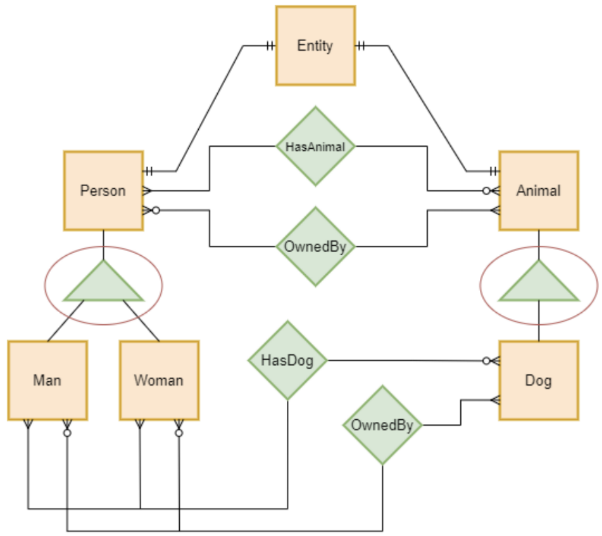
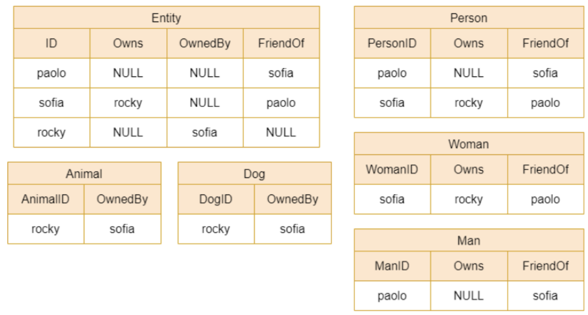


Figure 6.1: Modello ER

Notazione: i quadrati rappresentano concetti, i rombi rappresentano relazioni, i triangoli rappresentano una relazione IS-A (quindi, ad esempio, Uomo è un'istanza di Persona). Si noti che i domini, come li abbiamo definiti, non consentono le relazioni IS-A (per il momento, vedere Osservazione 6.3). Ecco perché questa relazione non deve essere considerata per i linguaggi ER da considerarsi linguaggi assertivi. Per quanto riguarda le frecce, quella con due stanghe rappresenta una relazione uno a uno, mentre le frecce a forchetta rappresentano una relazione molti a molti, in cui quelle con un punto bianco identificano l'esistenza facoltativa della relazione.

Esempio 6.9 (Un linguaggio assertivo di dati, utilizzando database relazionali, che descrive il dominio dei dati nell'Esempio 6.2) Considerando le affermazioni dell'esempio 6.1, ecco come i nostri dati sono rappresentati da un database:



Entity

ID	Owns	OwnedBy	FriendOf
paolo	NULL	NULL	sofia
sofia	rocky	NULL	paolo
rocky	NULL	sofia	NULL

Person

PersonID	Owns	FriendOf
paolo	NULL	sofia
sofia	rocky	paolo

Woman

WomanID	Owns	FriendOf
sofia	rocky	paolo

Animal

AnimalID	OwnedBy
rocky	sofia

Dog

DogID	OwnedBy
rocky	sofia

Man

ManID	Owns	FriendOf
paolo	NULL	sofia

Figure 6.2: Database con dati

Notiamo come alcune caselle siano istanziate come NULL, questo avviene per mancanza di informazioni.

Esempio 6.10 (Un linguaggio assertivo di conoscenza, utilizzando schemi di database relazionali, che descrive il dominio della conoscenza nell'Esempio 6.2) Invece, in questo esempio vediamo le relazioni delle diverse chiavi nel database (continuando a considerare le affermazioni nell'esempio 6.1): Notiamo come

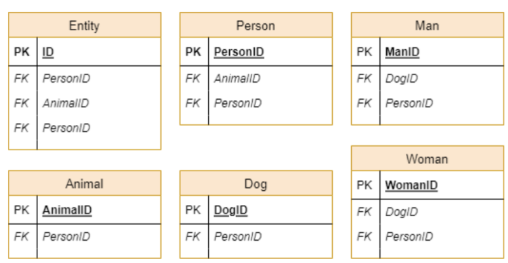


Figure 6.3: Relazione del Database

alcune caselle siano istanziate come NULL, questo avviene per mancanza di informazioni.

Osservazione 6.10 (Linguaggi assertivi, dati, conoscenza e misti) I tre esempi chiariscono lo scopo e lo scopo dei linguaggi e dei domini di dati, conoscenza e misti. I linguaggi di dati, che descrivono fatti nel mondo reale, formalizzano il contenuto dei database (relazionali) o rappresentazioni analoghe (file CSV, JSON, XML). I linguaggi di conoscenza specificano la conoscenza di riferimento, ad esempio, relazioni, attributi, dei linguaggi di dati. I linguaggi misti vengono utilizzati per specificare entrambi.

6.3 Funzione d'interpretazione

Definizione 6.8 (Funzione di interpretazione, interpretazione intensionale) La rappresentazione intenzionale I_i^A di una funzione di interpretazione $I_A : L_A \to D$ di un linguaggio assertivo è definita come segue:

$$\mathcal{I}_A^i = \langle \mathcal{I}_e, \mathcal{I}_C, \mathcal{I}_P \rangle \tag{6.5}$$

con:

$$\begin{aligned} \mathcal{I}_e &: \mathcal{E} && \to \mathtt{E} \\ \mathcal{I}_C &: \{\mathcal{C}\} && \to \{\mathtt{E}\} \\ \mathcal{I}_P &: \{\mathcal{P}^n\} && \to \{\mathtt{E}\} \times \cdots \times \{\mathtt{E}\} \end{aligned} \tag{6.6}$$

e di conseguenza:

$$\begin{aligned} I_{\mathcal{A}}(C(e)) &= I_{\mathcal{C}}(C)(I_{\mathcal{E}}(e)) && = e \in C \\ I_{\mathcal{A}}(P_n(e_1, \ldots, e_n)) &= I_{\mathcal{P}_n}(I_{\mathcal{E}}(e_1), \ldots, I_{\mathcal{E}}(e_n)) && = \langle e_1, \ldots, e_n \rangle \in R_n \\ I_{\mathcal{A}}(C) &= I_{\mathcal{C}}(C) && = C \subseteq E \\ I_{\mathcal{A}}(P_n(C_1, \ldots, C_n)) &= I_{\mathcal{P}_n}(I_{\mathcal{C}}(C_1), \ldots, I_{\mathcal{C}}(C_n)) && = R_n \subseteq C_1 \times \cdots \times C_n \end{aligned} \tag{6.7}$$

dove: I_e è la funzione di interpretazione delle entità, I_C è la funzione di interpretazione dei concetti e I_P è la funzione di interpretazione delle proprietà.

Notazione 6.2 (Notazione alternativa per la funzione di interpretazione) L'equazione (6.7) è talvolta scritta come segue:

$$\begin{aligned} I_{\mathcal{A}}(C(e)) &= C^I(e^I) \\ I_{\mathcal{A}}(P(e_i, e_j)) &= P^I(e_i^I, e_j^I) \\ I_{\mathcal{A}}(C) &= C^I \\ I_{\mathcal{A}}(P(C_i, C_j)) &= P^I(C_i^I, C_j^I) \end{aligned} \tag{6.8}$$

Osservazione 6.11 (Funzione di interpretazione) L'equazione (6.7) mostra come I_A viene applicata ricorsivamente applicandola ai componenti della sua affermazione di input fino all'interpretazione delle entità. In questo processo, i suoi componenti vengono applicati come necessario, cioè I_e alle entità, I_C ai concetti e I_P alle proprietà. La corrispondenza uno a uno tra il linguaggio e la struttura del dominio per costruire il significato di un'affermazione componendolo a partire dal significato dei suoi componenti.

Osservazione 6.12 (Funzione di interpretazione, interpretazione intenzionale) Quando D_i e L_i^A vengono costruiti seguendo le regole evidenziate nell'Osservazione 6.8 e I_A è come nell'Equazione (6.7), allora la rappresentazione del mondo è intuitiva e autoesplicativa, mappata uno a uno con la rappresentazione analoga che sta descrivendo. Ciò consente allo sviluppatore della rappresentazione di concentrarsi solo sulle rappresentazioni linguistiche, pensando implicitamente alla corrispondente rappresentazione analogica.

Definizione 6.9 (Funzione di interpretazione, dati, conoscenza, misto) Una funzione di interpretazione dei dati mappa un linguaggio assertivo dei dati in un dominio dei dati. Una funzione di interpretazione della conoscenza mappa un linguaggio assertivo della conoscenza in un dominio della conoscenza. Una funzione di interpretazione mista mappa un linguaggio misto in un dominio misto.

Esempio 6.11 (Una funzione di interpretazione dei dati dal linguaggio dei dati nell'Esempio 6.5 al dominio dei dati nell'Esempio 6.2) Un'istanza di applicazione è

$$\begin{aligned} I_A(FriendOf1(sofia, paolo)) &= I_P(FriendOf1)(I_e(sofia), I_e(paolo)) \\ &=< sofia, paolo > \in FriendOf1 \end{aligned}$$

Esempio 6.12 (Una funzione di interpretazione della conoscenza dal linguaggio assertivo nell'Esempio 6.6 al dominio della conoscenza nell'Esempio 6.3) Un'istanza di applicazione è

$$I_A(FriendOf1) = I_P(FriendOf1) = FriendOf1 \subseteq Person \times Person$$

Esempio 6.13 (Un'interpretazione della conoscenza dal linguaggio del modello ER nell'Esempio 6.1 al dominio della conoscenza nell'Esempio 6.3) Un'istanza di applicazione è NECESSITA' DI IMMAGINE CON MAPPING DI UN CONCETTO E UN ELEMENTO DI PROPRIETÀ

Esempio 6.14 (Una funzione di interpretazione dei dati dal linguaggio del database nell'Esempio 6.9 al dominio dei dati nell'Esempio 6.2) Un'istanza di applicazione è NECESSITA' DI IMMAGINE CON MAPPING DI UN ELEMENTO DI DATI COMMENTO SU COME MAPPARE NULL

Esempio 6.15 (Una funzione di interpretazione della conoscenza dal linguaggio dello schema del database nell'Esempio 6.15 al dominio della conoscenza nell'Esempio 6.2) Un'istanza di applicazione è NECESSITA' DI IMMAGINE CON MAPPING DI UN ELEMENTO DI DATI

6.4 Modello del mondo

Definizione 6.10 (Dominio del Mondo, rappresentazione intenzionale) Dato un Dominio del Mondo $\hat{W}$ definito come

$$\hat{W} =< L_A, D, I_A >$$

la sua rappresentazione intenzionale $\hat{W^i}$ è definita come

$$\hat{W^i} =< L_A^i, D_i, I_A^i > \tag{6.9}$$

con

$$L_A^i =< E, \{C\}, \{P\} >$$
$$D_i =< E, \{C\}, \{R\} >$$
$$I_A^i =< I_e, I_C, I_P >$$

dove $\hat{W}^i$ è (chiamato) il modello di mondo stencil, D_i è lo stencil del dominio, L_A^i è lo stencil del linguaggio (assertivo) e I_A^i è lo stencil della funzione di interpretazione.

Osservazione 6.13 (Lo stencil del modello del mondo) $\hat{W}^i$ è tutto ciò di cui abbiamo bisogno per definire un modello del mondo e per utilizzarlo per implementare teorie e modelli corrispondenti. Vedi le Osservazioni 5.13 e 5.14 e la Definizione 5.7.

Terminologia 6.2 (Data, conoscenza, misto) Parliamo di modelli di mondo, teorie e modelli di dati, conoscenza e misti con il significato ovvio.

Osservazione 6.14 (Modello di mondo) La Definizione 6.10 ci dice che un modello di mondo è composto da un linguaggio L_A di affermazioni su un dominio di riferimento D, in cui il significato delle affermazioni in L_A è univoco nel senso che il fatto denotato da un'affermazione è definito in modo univoco dal ruolo chiave della funzione di interpretazione I_A. In questo senso, vale la pena riassumere le proprietà chiave delle funzioni di interpretazione, come introdotte e discusse sopra.

- Non consente la polisemia: un elemento nel linguaggio denota uno e un solo elemento del dominio. L'ambiguità denotazionale non è possibile;
- Consente la sinonimia: un elemento del dominio può essere denotato da più di un elemento del linguaggio;
- È totale: ogni elemento del linguaggio ha una denotazione nel dominio di riferimento. Le affermazioni prive di significato non sono consentite;
- Non è (necessariamente) surgettivo: possono esistere elementi del dominio per i quali non esistono elementi nel linguaggio che li denotano.
- È compositivo: il significato di un'affermazione è calcolato mediante la composizione funzionale dei suoi elementi costitutivi. Il significato delle affermazioni è codificato intenzionalmente nella loro struttura.

Osservazione 6.15 (Definizione di un modello di mondo) La costruzione di un modello di mondo $\widehat{W} =< L_A, D, I_A >$ richiede di passare attraverso la definizione dei suoi tre componenti. La motivazione è di solito la necessità di modellare un determinato dominio di problema. Il modellatore di solito inizia con una comprensione intuitiva del dominio, anche se non è lavorata in dettaglio. Il processo di solito segue i seguenti passaggi:

1. Definire L_A, dove l'idea chiave è che L_A dovrebbe catturare gli aspetti chiave del dominio di destinazione;

2. Specificare D assicurandosi che sia una rappresentazione analogica adeguata del dominio di destinazione;

3. Definire I_A, in questo processo convalidando il fatto che L_A e D siano adeguatamente definiti e allineati reciprocamente.

Spesso gli ultimi due passaggi, almeno inizialmente, vengono eseguiti solo in modo intuitivo, facendo affidamento sulla composizione della funzione di interpretazione. Questi passaggi di solito vengono eseguiti completamente solo quando il modello di mondo è stato utilizzato per un po' di tempo e le sue ambiguità, se presenti, sono diventate esplicite. Nel complesso, il processo di costruzione dei modelli di mondo è complesso, richiede molta esperienza. Potrebbe richiedere anni per definire completamente un modello di mondo.

Esempio 6.16 (Definizione di un modello di mondo) In tutta questa sezione, a partire dalla Sezione 2, abbiamo costruito modelli di mondo. Modelli di mondo semplici possono essere costruiti raccogliendo linguaggio, dominio e funzione di interpretazione da questi esempi e allineandoli in modo adeguato. Tuttavia, si noti che, come già menzionato in precedenza, la definizione dei modelli di mondo è un compito molto complesso che richiede molta competenza, validazione nel campo e, in ultima analisi, tempo. Come già menzionato in precedenza, i modelli ER, i modelli UML, sottoinsiemi del linguaggio naturale arricchiti con adeguate funzioni di interpretazione, sono esempi di modelli di mondo.

Osservazione 6.16 (Utilizzo di un modello di mondo - costruzione e ragionamento su una teoria del mondo) Il problema è come descritto nella Sezione 1.3 e nella Sezione 2.2. Abbiamo un modello di mondo e dobbiamo utilizzare la sua struttura per costruire un insieme di affermazioni, cioè una teoria del mondo, e dobbiamo assicurarci di cosa essa rappresenta, ovvero il modello di riferimento, e le sue proprietà. Una volta che una teoria è articolata con riferimento a un modello di mondo, diventa possibile studiare le sue proprietà attraverso una metodologia generale e algoritmi definiti nella prossima sottosezione.

L'aspetto chiave nella definizione o selezione di un modello di mondo è la decisione sul linguaggio assertivo L_A e il livello di ambiguità che esso permette.

Intuizione 6.5 (Linguaggio, informale, semi-formale, logico) I linguaggi dei modelli di mondo sono di tre tipi, come segue:

- **Linguaggi informali**, ossia linguaggi in cui la sintassi di L_A è definita in modo informale, ad esempio, in linguaggio naturale e senza l'uso di regole di produzione.

- **Linguaggi semi-formali**, ossia linguaggi in cui la sintassi di L_A è definita formalmente.

- **Linguaggi formali**, chiamati anche linguaggi logici, ossia linguaggi in cui L_A e I_A sono definiti formalmente.

Esempio 6.17 (Linguaggio, informale, semi-formale, logico) Con riferimento all'esempio precedente, abbiamo quanto segue:

- Linguaggi informali: tutte le lingue naturali;
- Linguaggi semi-formali: il linguaggio relazionale dei database, la notazione ER e UML;
- Linguaggi logici: i linguaggi utilizzati nella logica.

Osservazione 6.17 (Linguaggio, informale, semi-formale, logico) Abbiamo quanto segue.

- I linguaggi informali sono più facili da usare in quanto sono conosciuti da tutti. Nell'Informatica, vengono tipicamente utilizzati nella stesura dei requisiti iniziali, ad esempio, da mostrare ai clienti. La principale difficoltà è che c'è una elevata probabilità di fraintendimento. Ora vengono utilizzati anche nelle interazioni con i ChatBot.
- I linguaggi semi-formali vengono tipicamente utilizzati nell'Ingegneria del Software quando si scrivono requisiti avanzati. Riducono il livello di ambiguità e sono molto efficaci nel lavoro collaborativo tra gli Ingegneri del Software. Possono anche essere utilizzati nella generazione automatica di codice. Possono sorgere problemi a causa della presenza di un certo livello di ambiguità semantica.
- I linguaggi logici hanno due principali utilizzi: (i) La specifica di software e hardware altamente critici (ad esempio, sistemi critici per la sicurezza o la sicurezza) e (ii) l'implementazione di sistemi di ragionamento, tipicamente sistemi di Intelligenza Artificiale, in grado di calcolare le conseguenze da ciò che è noto.

6.5 Utilizzare un modello di mondo

Vediamo come i modelli di mondo possono essere utilizzati nella pratica per rappresentare e ragionare sul mondo. Ci concentriamo sui modelli di mondo logici.

Intuizione 6.6 (La risoluzione dei problemi mediante l'uso di modelli di mondo) L'uso dei modelli di mondo può essere caratterizzato come una soluzione alla necessità di rispondere a domande (in linguaggio naturale) o query (in qualche linguaggio formale, ad esempio SQL) sul mondo, grazie allo sfruttamento di un meccanismo di ragionamento che consente di calcolare conseguenze da un corpo di conoscenza esistente sul mondo, cioè una teoria. Questo processo di solito si articola in tre compiti principali, come segue (vedi anche Figura 6.4):

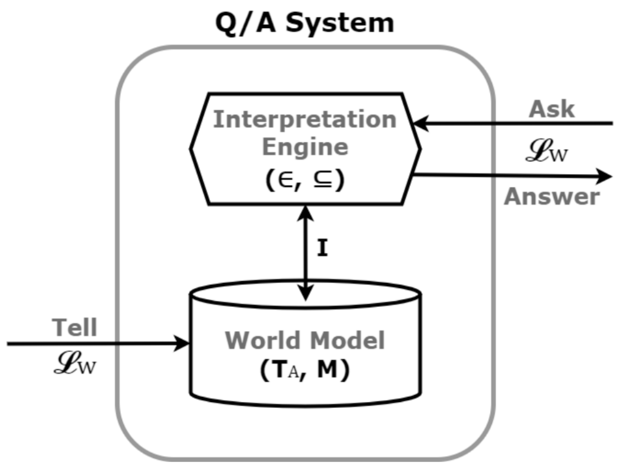


Figure 6.4: Risolvere problemi utilizzando il modello di mondo

- Comunicare Dati e Conoscenza, da un responsabile di sistema al sistema. Ciò richiede la specifica di un linguaggio L_T che può essere utilizzato per specificare i fatti del modello di mondo in uso.
- Porre una Domanda, da un utente al sistema, ovvero le domande / query specifiche su cui l'utente vuole conoscere le risposte. Ciò richiede la specifica di un linguaggio L_Q utilizzato per scrivere domande / query.
- Rispondere a una Domanda, dal sistema all'utente, riguardo alla domanda posta precedentemente, scritta in un linguaggio L_A.

Alcune osservazioni. Il responsabile e l'utente sono di solito persone diverse. Di solito il processo di fornire al sistema ulteriori informazioni è continuo nel tempo. Di solito L_Q e L_A sono lo stesso linguaggio L. Molto spesso il linguaggio L_T è diverso sia da L_Q che da L_A. Ci sono tre principali motivazioni per questo: efficienza del ragionamento, intuitività dell'interazione con l'utente e, non da ultimo, la necessità che L_T sia più adatto per la specifica del modello di mondo. Ad esempio, come sarà chiaro nel seguito, i modelli basati su grafi sono molto adatti per la specifica del modello di mondo, mentre L_Q e L_A sono spesso in (alcuni frammenti di) linguaggio naturale. La traduzione tra questi linguaggi è sempre automatica e implementata all'interno del sistema di domande e risposte.

Terminologia 6.3 (Ne Il linguaggio del modello di mondo) Nelle seguenti assumiamo che $L_Q = L_A = L_T = L_W$, dove L_W è il linguaggio di rappresentazione del modello di mondo.

Definizione 6.11 (Interpretazione e conseguenza) Sia $\hat{W} =< L_A, D, I_A >$ un modello di mondo. Sia $T \subseteq L_A$ una teoria e $M \in D$ un modello di $\hat{W}$. Sia

$a \in T$ un'asserzione. Allora scriviamo

$$\begin{aligned} &M \models_{L_A} a \text{ to mean } I_A(a) \in M \\ &M \models_{L_A} T \text{ to mean } I_A(a) \in M \text{ per ogni } a \in T \end{aligned} \tag{6.10}$$

e diciamo che M implica T, o anche che M implica a. La notazione per il linguaggio L_A viene omessa quando non è necessaria.

Intuizione 6.7 (Modelli di mondo, problemi di ragionamento) Ma quali domande e quali risposte? Tutti i modelli di mondo forniscono risposte a quattro (fondamentali) domande fondamentali che enunciamo di seguito come problemi di ragionamento. Supponiamo di avere un modello di mondo $\hat{W}$ definito intensionalmente, cioè $\hat{W}^i =< L_i, D_i, I_i >$ e che abbiamo un insieme M di modelli con $M \subseteq D$ e un insieme di teorie $T \subseteq L_A$. Allora abbiamo quanto segue (notare che un'asserzione si comporta come con un elemento):

Problema di ragionamento 6.1 (Model checking) Given T and M, check whether $M \models T$.

Osservazione 6.18 (Verifica del modello e correttezza della teoria) La verifica del modello è la stessa cosa della verifica della correttezza di una teoria rispetto a un modello, come dalla Definizione 5.7. È sufficiente verificare se le asserzioni in T si verificano in M.

Problema di ragionamento 6.2 (Satisfiability) Given T , check whether there exists M such that $M \models T$.

Osservazione 6.19 (Soddisfacibilità) Qualsiasi teoria che non rappresenti informazioni negative, come più spesso accade, è soddisfacibile. Se le informazioni negative sono consentite, è sufficiente verificare se la teoria contiene due fatti che si contraddicono reciprocamente.

Osservazione 6.20 (Risposta alle query nei database) La risposta alle query nei database è una forma sofisticata di verifica del modello / soddisfacibilità. Il contenuto del database è il modello di mondo di riferimento, la query è la teoria da verificare, la risposta è l'insieme di istanze che rendono corretta la teoria di input.

Problema di ragionamento 6.3 (Validity) Given T, check whether for all M, $M \models T$.

Osservazione 6.21 (Validità) Applicare la verifica del modello a tutti i modelli M.

Problema di ragionamento 6.4 (Insoddisfacibilità) Dato T, verificare se non esiste alcun M tale che $M \models T$.

Osservazione 6.22 (Insoddisfacibilità) Se le informazioni negative sono consentite, verificare due asserzioni contraddittorie.

Intuizione 6.8 (Un'architettura per la risoluzione dei problemi mediante l'uso di modelli di mondo) L'architettura che supporta l'uso dei modelli di mondo, come specificato nell'Intuizione 6.6, è rappresentata nella Figura 6.4. Possiamo identificare due componenti principali, come segue:

- Un motore di modelli di mondo (inferenza) che codifica i dati e le conoscenze disponibili sul mondo e consente un ragionamento minimo su di essi (vedere i problemi di ragionamento nell'Intuizione 6.7);
- Un motore di interpretazione (inferenza) che implementa uno o più dei problemi di ragionamento definiti nell'Intuizione 6.7.

Nota che il sistema viene selezionato all'inizio. La scelta dipende dalle specifiche del problema da risolvere.

Osservazione 6.23 (Motore di inferenza del modello di mondo) Il ragionamento in un modello di mondo equivale a verificare se una query (un'unica asserzione o un insieme di asserzioni come parte di una teoria) appartiene al modello. Query più sofisticate, come quelle implementate nei database relazionali, possono essere implementate in cui è possibile lasciare alcuni elementi della query non specificati. Il motore di inferenza troverà quindi il corretto s.

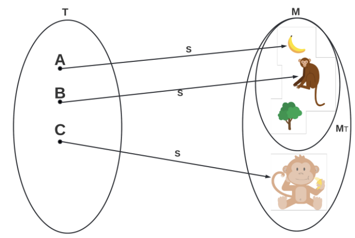
6.6 Esercizi

Esercizio 6.1 (ER Creation) Create an ER Model from this theory:

- There is a tree
- There is a banana
- The monkey is eating a banana
- The monkey is sitting on a tree*
- The monkey is scratching his head*

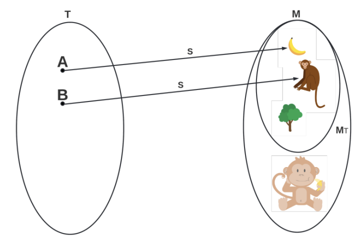
Esercizio 6.2
Considera le frasi e la modellazione della teoria. Dimmi se è completa, corretta, completa e corretta, incompleta o scorretta.

- A = ”There is a banana”
- A = ”There is a monkey”
- C = ”There is a tree”
- D = ”The monkey is eating a banana”



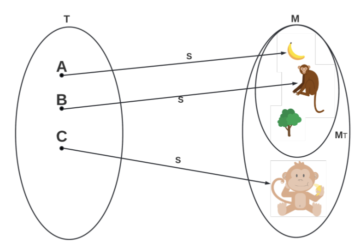
Esercizio 6.3
Considera le frasi e la modellazione della teoria. Determina se è completa, corretta, completa e corretta, incompleta o errata.

- A = ”There is a banana”
- A = ”There is a monkey”
- C = ”There is a tree”
- D = ”The monkey is eating a banana”



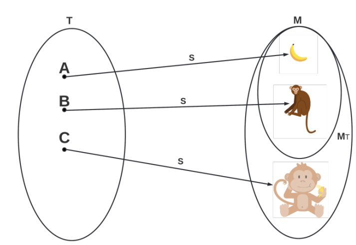
Esercizio 6.4
Considera le frasi e la modellazione della teoria. Determina se è completa, corretta, completa e corretta, incompleta o errata.

- A = ”There is a banana”
- A = ”There is a monkey”
- C = ”The monkey is eating a banana”
- D = ”There is a tree”



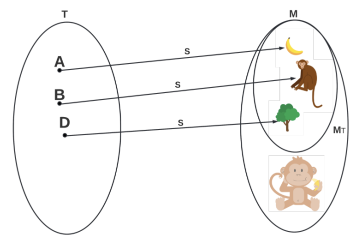
Esercizio 6.5
Considera le frasi e la modellazione della teoria. Determina se è completa, corretta, completa e corretta, incompleta o errata.

- A = ”There is a banana”
- A = ”There is a monkey”
- C = ”There is a tree”
- D = ”The monkey is eating a banana”



Esercizio 6.6
Considera le frasi e la modellazione della teoria. Determina se è completa, corretta, completa e corretta, incompleta o errata.

- A = "There is a banana"
- A = "There is a monkey"
- C = "There is a tree"
- D = "The monkey is eating a banana"



Chapter 7

Grafi di conoscenza - rappresentare il mondo come un grafo

Rappresentiamo i modelli del mondo $\hat{W} = \langle L_A, D, I_A \rangle$ come modelli di knowledge graph formalmente rappresentati come $\hat{\mathcal{KG}} = \langle L_{KG}, D, I_{KG} \rangle$, ossia come tipi speciali di grafi con etichette in cui ogni tripla $\langle$nodo, arco, nodo$\rangle \in L_{KG}$ è un'affermazione $a \in L_{KG}$ che descrive un fatto $f \in D$, con $f = I_{KG}(a)$. Chiamiamo le teorie in L_{KG} knowledge graphs e utilizziamo la notazione $KG \subseteq L_{KG}$. Analogamente ai modelli del mondo, distinguiamo tra i modelli $\hat{\mathcal{KG}}$ che rappresentano dati, conoscenza o informazioni miste.

Definizione 7.1 (Entità, etipo e mixed $\hat{\mathcal{KG}}$):

- Un $\hat{\mathcal{KG}}$ che è un modello di mondo dei dati è chiamato *entity graph model* ($\hat{\mathcal{EG}}$).
- Un $\hat{\mathcal{KG}}$ che è un modello di mondo della conoscenza è chiamato *etype graph model* ($\mathcal{E}\hat{\mathcal{T}}\mathcal{G}$).
- Un $\hat{\mathcal{KG}}$ che è un modello di mondo misto è chiamato *etype entity graph model* ($\mathcal{E}\hat{\mathcal{E}}\mathcal{G}$).

Questa terminologia si estende in modo ovvio a tutti i componenti di $\hat{\mathcal{KG}}$, così come a tutti i $\hat{\mathcal{KG}}$ e modelli correlati definiti all'interno di un $\hat{\mathcal{KG}}$.

7.1 Dominio

Sia `KG = D`$^i = \langle E, \{C\}\{R\} \rangle$ lo stencil di un dominio KG. Definiamo i suoi componenti.

Definizione 7.2 (L'Universo di Interpretazione E di un KG) L'Universo di Interpretazione E di un KG è definito come

$$\texttt{E} = \texttt{ET} \cup \texttt{DT} \tag{7.1}$$

con $\texttt{ET} \cap \texttt{DT} = \emptyset$, dove $\texttt{ET} = \{\texttt{E}_\text{T}\}$ con $\texttt{E}_\text{T} = \{\texttt{e}\}$, e $\texttt{DT} = \{\texttt{D}_\text{T}\}$ con $\texttt{D}_\text{T} = \{\texttt{v}\}$ $\texttt{E}_\text{T}$ è un **entity type** (**etype**) è $\texttt{D}_\text{T}$ è un **datatype** (**dtype**). Gli elementi degli etype sono chiamati entità, quelli dei dtipi sono chiamati valori (dati).

Osservazione 7.1 (Etype, dtype) In un KG, E è strutturato in un insieme di sotto-universi, cioè etipi e dtipi. In astratto, ciascun sotto-universo è simile a una classe $C \in \{C\}$, ovvero un sottoinsieme di E. La differenza fondamentale è che etipi e dtipi sono tipi che, come nei linguaggi di programmazione, vengono definiti quando si definisce LKG e sono quindi indipendenti dall'applicazione. In quanto tali, questi tipi sono dotati di determinate proprietà e operatori di tipo incorporati, in particolare: un insieme di costruttori che consentono di creare gli elementi di un tipo, un riconoscitore in grado di determinare se un certo elemento appartiene a un certo tipo e una relazione di equivalenza che consente di decidere se due elementi di quel tipo sono uguali.

Esempio 7.1 (Etype) Un esempio di etipo è: `Location`, in cui intuitivamente una `location` è un etipo che contiene spazialmente altre entità. Le locations di solito non cambiano la loro posizione rispetto ai loro sistemi di riferimento delle coordinate. Le loro coordinate spaziali sono quindi un importante indicatore per decidere se due locations (ossia due entità che appartengono all'etipo `Location`) sono effettivamente la stessa location. Ci sono molti etipi che sono casi speciali (sottoetipi) di Location, ad esempio: `Mountain`, `City`, `Street`, `Home` e molti altri. Altri etipi importanti sono: Entity, il più generale etipo, quello che contiene tutti gli elementi in ET (la sua proprietà più evidente è che ha un nome, implementando così il requisito che tutte le entità devono avere un nome); `Event`, le cui proprietà più caratterizzanti sono i tempi di inizio e di fine, `Person`, le cui proprietà più caratterizzanti sono il nome, la data di nascita e i genitori; e molti altri.

Osservazione 7.2 (Dtype)

I dtipi hanno le stesse proprietà degli etipi, ma ne aggiungono altre due:

- l'insieme dei loro membri, cioè dei loro valori, è predefinito
- i nomi dei valori sono gli stessi dei valori stessi (cioè i valori dei dati si denotano da soli, quindi, ad esempio, il numero (propriamente chiamato numero) 3,14 è il nome del numero 3,14).

Esempio 7.2 (Dtype) La seguente è una lista non esaustiva di tipi di dati:

`Dtype, Float, Integer, Boolean, String, SpaceTime, Identifier` dove `Float, Integer, Boolean, String` definiscono rispettivamente lo spazio dei numeri reali, degli interi, dei valori booleani e delle stringhe. SpaceTime è l'insieme di valori utilizzati per descrivere spazio e tempo. Pertanto, i sotto-tipi di SpaceTime includono GeoCoordinate, Distance, XYCoordinate ma anche Date, Time, DateTime e così via. Dtype è l'insieme di tutti i valori dei dati.

Osservazione 7.3 (Set di classi C di un KG) Le classi di KG sono classi di modelli del mondo "così come sono", vedere la Definizione 6.1.

Definizione 7.3 (Le relazioni binarie tra oggetti e dati R di un KG.) L'insieme di relazioni $\{R\} = \{OR\} \cup \{DR\}$ è un insieme di relazioni binarie di un KG tale che

$$\mathtt{R} \subseteq \mathtt{E}_{\mathrm{T}_s} \times \{\mathtt{E}_{\mathrm{T}_T} \cup \mathtt{D}_{\mathrm{T}_t}\} \tag{7.2}$$

con $\mathtt{ET}_s$, $\mathtt{ET}_t \in \mathtt{ET}$ e $\mathtt{DT}_t \in \mathtt{DT}$. Se R è definito come:

$$\mathtt{R} \subseteq \mathtt{E}_{E_t} \times \mathtt{E}_{E_t} \tag{7.3}$$

allora diciamo che R è una relazione oggetto binaria $\mathtt{OR} \in \{\mathtt{OR}\}$. Se R è definito come:

$$\mathtt{R} \subseteq \mathtt{E}_{tT} \times \mathtt{D}_{T_t} \tag{7.4}$$

allora diciamo che R è una relazione dati binaria $\mathtt{DR} \in \{\mathtt{DR}\}$.

Osservazione 7.4 (Arità di una relazione R) In un KG ci sono solo relazioni binarie, ciò consente l'uso di grafi con archi di ingresso e uscita singoli. La rappresentazione di un modello del mondo con archi più complessi richiede la sua riformulazione per consentire solo archi uno-a-uno.

Osservazione 7.5 (Argomenti di una relazione R) Diversamente dalla Definizione 6.1, le relazioni prendono come argomenti etipi e dtipi. Vedere l'Osservazione 7.1 per una spiegazione. Successivamente vedremo come reintrodurre relazioni che prendono come argomenti concetti definiti dall'utente.

Osservazione 7.6 (Relazione tra oggetti) Le relazioni tra oggetti sono relazioni tra entità, rappresentando quindi come le entità interagiscono. Pertanto, ad esempio, alcuni esempi sono:

`Near(Person, Tree), HasFather(Person, Person), TalksTo(Person, Dog)`.

Si noti che `Person, Tree, Person, Dog` sono etipi, piuttosto che concetti.

Osservazione 7.7 (Relazione dati) Le relazioni dati rappresentano le proprietà delle entità come tali, indipendentemente dalle loro interazioni con altre entità. Ad esempio, alcuni esempi sono:

`Height(Person, Float), HasName(Person, String), HasId(Entity, Identifier)`.

Si noti che `Float, String, Identifier` sono dtipi, piuttosto che concetti, come era il caso nella Sezione 5.

Osservazione 7.8 (Relazione, cardinalità) La cardinalità di una relazione può essere: 1-a-1, 1-a-n, m-a-n (dove uno tra m o n può anche essere 0, il che significa che possono esistere entità che non appartengono a nessuna relazione).

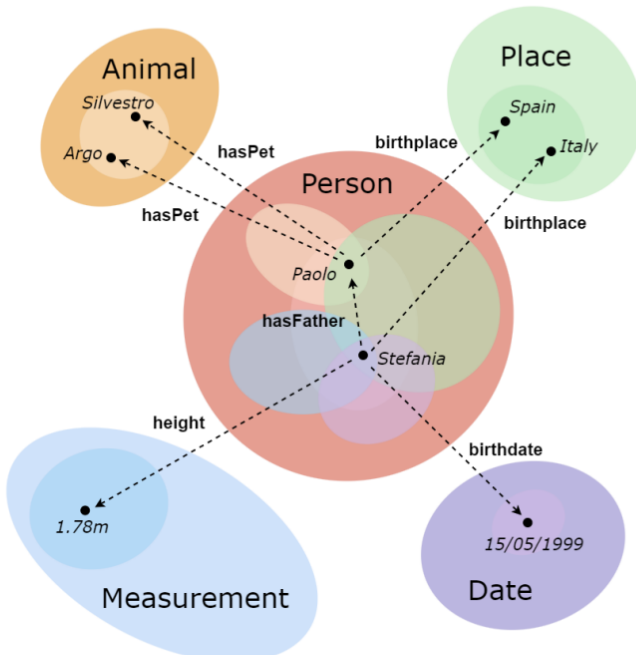


Figure 7.1: diagramma di Venn di un EG

Esempio 7.3 (Dominio in un $\hat{\mathcal{EG}}$, Diagramma di Venn) Rappresentiamo i domini degli EG come nella Figura 7.1. Le relazioni sono rappresentate come collegamenti tra entità dell'etipo appropriato e entità o valori dell'etipo o del dtipo appropriato, rispettivamente.

Esempio 7.4 (Dominio in $\mathcal{E}\hat{\mathcal{T}}\mathcal{G}$, Diagramma di Venn) Rappresentiamo i domini degli ETG come nella Figura 7.1. Le relazioni sono rappresentate come collegamenti tra etipi ed etipi/dtipi.

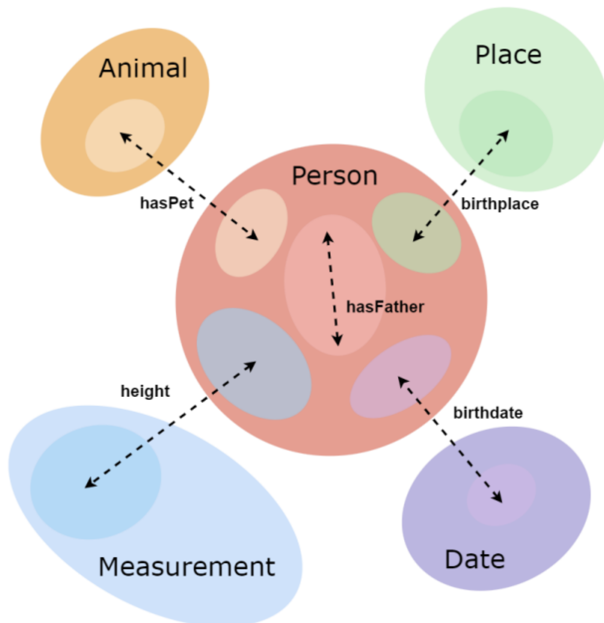


Figure 7.2: diagramma di Venn di un ETG

7.2 Linguaggio assertivo

Sia $\mathcal{L}^i_{\mathcal{KG}} = \mathcal{L}^i_{\mathcal{A}} = \langle \mathcal{E}, \{\mathcal{C}\}\{\mathcal{P}\}\rangle$ lo stencil del $\hat{\mathcal{KG}}$ linguaggio $L^i_{\mathcal{KG}}$. La definizione di $L^i_{\mathcal{KG}}$ segue direttamente dalla definizione di $\mathtt{D}^i$ nella Sezione 7.1.

Definizione 7.4 (Concetto) Abbiamo

$$\{\mathcal{C}\} = \mathcal{ET} \cup \mathcal{DT} \tag{7.5}$$

dove $\mathcal{ET} = \{\mathcal{E}_{\mathcal{T}}\}$ e $\mathcal{DT} = \mathcal{D}_{\mathcal{T}}$ sono rispettivamente (nomi dei) **etipi** e **dtipi** in $\hat{\mathcal{KG}}$

Definizione 7.5 (Proprietà degli oggetti e dei dati)

$$\{\mathcal{P}\} = \{\mathcal{OP}\} \cup \{\mathcal{DP}\} \tag{7.6}$$

dove $\{\mathcal{OP}\}$ e $\{\mathcal{DP}\}$ sono definiti come segue:

$$\begin{aligned} \{\mathcal{OP}\} &\subseteq \{\mathcal{E}_{\mathcal{T}}\} \times \{\mathcal{E}_{\mathcal{T}}\} \\ \{\mathcal{OP}\} &\subseteq \{\mathcal{E}_{\mathcal{T}}\} \times \{\mathcal{E}_{\mathcal{T}}\} \end{aligned} \tag{7.7}$$

Gli elementi di $\{\mathcal{OP}\}$ sono chiamati proprietà degli oggetti, quelli di $\{\mathcal{DP}\}$ proprietà dei dati.

Esempio 7.5 ($\mathcal{EG}$) L'$\mathcal{EG}$ di questo esempio rappresenta il dominio descritto nell'Esempio 7.3. La chiave dell'osservazione sta nel fatto che il numero di nodi corrisponde al numero di entità e valori, mentre il numero di archi corrisponde ai valori di proprietà istanziate. Si noti come i vincoli di cardinalità n-a-m consentano agli archi etichettati con la stessa proprietà di uscire dallo stesso nodo di etipi. Si noti anche come i nodi di dati siano e possano essere solo nodi foglia.

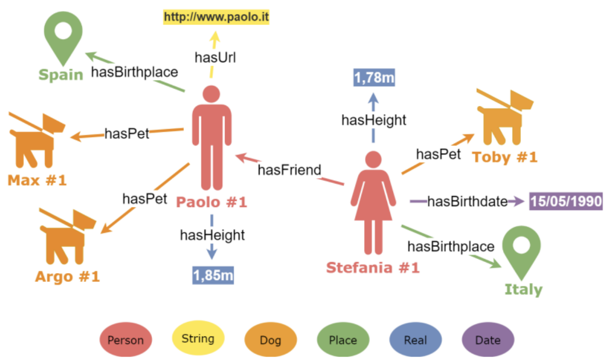


Figure 7.3: Esempio di un EG che rappresenta informazioni riguardo agli esseri umani.

Esempio 7.6 $\mathcal{ETG}$ L'$\mathcal{ETG}$ di questo esempio rappresenta il dominio descritto nell'Esempio 7.4. C'è un nodo per etipo e tipo di dati. L'$\mathcal{ETG}$ codifica alcune meta-informazioni, ad esempio i vincoli di cardinalità, che possono guidare e controllare la creazione dell'$\mathcal{EG}$ a partire da un $\mathcal{ETG}$. In modo simile agli $\mathcal{EG}$, i nodi dei dtipi sono nodi foglia.

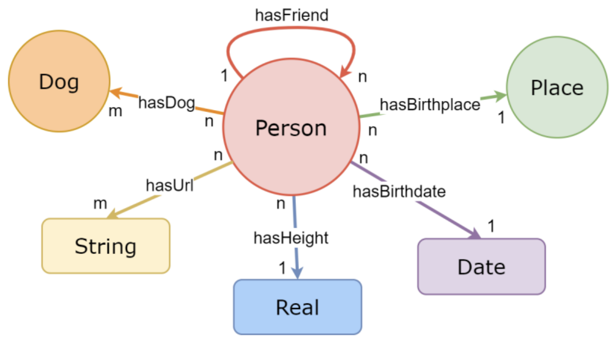


Figure 7.4: Esempio di un ETG che rappresenta informazioni riguardo agli esseri umani.

Osservazione 7.9 ($\hat{\mathcal{EG}}, \mathcal{E}\hat{\mathcal{TG}}, \mathcal{E}\hat{\mathcal{EG}}$) Nelle affermazioni di $\hat{\mathcal{KG}}$, le triple sono ¡ nodo, arco, nodo ¿ $\in \mathcal{L}_A$. Nei nodi di $\hat{\mathcal{KG}}$, i nodi $\hat{\mathcal{EG}}$ rappresentano o entità con il loro etipo o valori con il loro dtype. Nei nodi di $\mathcal{E}\hat{\mathcal{TG}}$, i nodi rappresentano etipi. Nei nodi di $\mathcal{E}\hat{\mathcal{EG}}$, ci sono entrambi i tipi di nodi. Gli archi sono etichettati con nomi di proprietà e rappresentano relazioni. Gli archi da etipi/entità a etipi/entità rappresentano relazioni tra oggetti. Gli archi da etipi/entità a dtipi/valori rappresentano relazioni tra dati.

7.3 Funzione di interpretazione

La funzione di interpretazione di un $\hat{\mathcal{KG}}$ è una mappatura diretta dal linguaggio assertivo $\mathcal{L}^i_{\mathcal{KG}}$ al dominio di interpretazione di destinazione $\mathtt{D}$. Come si può vedere dagli esempi nella Sezione 7.1 e nella Sezione 7.2, esiste una mappatura quasi diretta tra il linguaggio e il dominio di interpretazione. In pratica, ciò significa che una volta che è stata chiarita la denotazione degli elementi del singolo linguaggio di $\mathcal{L}_{\mathcal{KG}}$ e ci si è assicurati che la funzione di interpretazione soddisfi tutti i requisiti (vedi Sezione 6.4), il significato previsto di un $\mathcal{KG}$ può essere letto direttamente dal $\mathcal{KG}$ stesso.

7.4 Grafo di conoscenza

I modelli di grafo della conoscenza del mondo $\hat{\mathcal{KG}} = \langle \mathcal{L}^i_{\mathcal{KG}}, \mathtt{D}^i, \mathcal{I}^i_{KG} \rangle$ sono una rappresentazione universale, intuitiva, autoesplicativa ed efficiente dal punto di vista computazionale del mondo. Una volta fornito un $\hat{\mathcal{KG}}$, è sufficiente per costruire il proprio $\mathcal{KG}$ preferito e tutte le operazioni descritte nella Sezione 6.5 sono disponibili.

Osservazione 7.10 **(Tipi di grafi di conoscenza)** Come discusso anche nell'Osservazione 6.3, i modelli del mondo e i grafi di conoscenza introdotti finora hanno un'espressività minima. In pratica, i grafi di conoscenza sono spesso arricchiti con ulteriori costruttori che consentono la descrizione di fatti più complessi. Questa è un'operazione del tutto valida con l'avvertenza che è necessario fare attenzione nella selezione del giusto compromesso tra complessità, intuitività e complessità computazionale della rappresentazione selezionata.

Osservazione 7.11 **(Dai grafi di conoscenza alla logica)** I $\mathcal{KG}$ consentono una facile incorporazione nella logica più appropriata con l'obiettivo di consentire il ragionamento su di essi. Questo sarà l'argomento delle sezioni successive.

7.5 Esercizi

Esercizio 7.1 **(Crea un diagramma di insiemi)** Considera questa mappa della metropolitana di Milano:

Figure 7.5: Mappa della metropolitana di Milano

Cosa deve essere fatto:

- Estrarre insiemi rilevanti di oggetti: Linee, Stazioni, Incroci, Terminali, Numero di stazioni, Colori, ...
- Istanziare elementi del dominio: Linea rossa, Cadorna, RHO fiera, Milano, Giallo
- Estrarre relazioni rilevanti: haColore, appartieneA, faParteDi, vicinoA, ...

Chapter 8

Logica - Rappresentazione estensionale

I modelli del mondo ci consentono di specificare i principali componenti necessari per costruire una rappresentazione del mondo, sia estensionalmente che intensionalmente, da utilizzare successivamente per risolvere problemi. Ma ciò non è sufficiente.

Osservazione 8.1 (Linguaggi assertivi, limitazioni) La descrizione di domini e modelli utilizzando solo affermazioni è molto limitata. Ci piacerebbe avere modi più flessibili per descriverli. Un esempio di descrizione linguistica più ricca è la descrizione in lingua naturale riportata nella Figura 3.1 nell'Esempio 8.1.

Esempio 8.1 (Una ricca descrizione linguistica) Considera la rappresentazione analitica nella Figura 3.1. Una buona teoria informale che ne descrive il contenuto può essere articolata come segue.

"Paolo, Stefania e Sofia, tre grandi amici, si trovano in un incantevole parco, circondato dalla bellezza della natura. Ognuno di loro ha con sé quattro adorabili cani, pronti per una passeggiata rilassante. Paolo, con il suo sorriso contagioso, tiene i guinzagli di quattro cani amichevoli: Argo, che sembra essere l'animale più energetico del gruppo; Luna, dal pelo scuro e occhi luminosi; Max, un cane meticcio amichevole; e Penny, un piccolo cane affettuoso. Stefania, con la sua passione per gli animali, si gode la compagnia dei suoi quattro fedeli compagni: Toby, un cane affettuoso con una coda in movimento continuo; Ginger, un dolce cane marrone e bianco con orecchie appuntite; Rocky, dall'aspetto robusto ma dal cuore tenero; e Luna, un cucciolo carino e giocoso. Sofia, con il suo portamento tranquillo, cammina elegantemente seguita dai suoi quattro adorabili cani: Balto, un cane a pelo lungo e orecchie appuntite; Rex, un'anima protettiva e leale; Bella, una cagnolina elegante e affettuosa; e Charlie, un simpatico incrocio di razza con la lingua fuori, pronto a fare amicizia."

Essere in grado di esprimere descrizioni linguistiche come quella dell'Esempio

8.1 richiede un'estensione sostanziale della capacità espressiva dei linguaggi utilizzati per descrivere i modelli del mondo, i cosiddetti linguaggi di rappresentazione. Ciò ci costringe a passare dai modelli del mondo alla logica.

8.1 Dominio

L'obiettivo è avere un linguaggio più espressivo, se confrontato con quello di un modello del mondo, per descrivere un dominio di interpretazione, lasciando quest'ultimo invariato. Vedi la Sezione 5.1.

Osservazione 8.2 (Dominio, modello) Come dalla Definizione 5.2, abbiamo $M = \{f\} \subseteq D$, dove $\{f\}$ è un fatto, M è un modello e D è un dominio di interpretazione.

8.2 Sistema di rappresentazione

Definizione 8.1 (Sistema di rappresentazione, formule atomiche, formule complesse, funzione di interpretazione della rappresentazione) Sia $\hat{W} =< L_A, D, I_A >$ un modello del mondo con $I_A : L_A \to D$. Sia L_a tale che $L_A \subseteq L_a$ e tale che esista una funzione di interpretazione della rappresentazione $I : L_a \to D$ con $I_A \subseteq I$. Allora, un linguaggio di rappresentazione L è definito come

$$L = \{w\} = L_a \cup L_c, \text{ con } L_a \subset L_c \tag{8.1}$$

dove: I_A è come definito nella Definizione 6.7, $w \in L$ è una formula (ben formata), $w \in L_a$ è una formula atomica (ben formata) e $w \in L_c$ è una formula complessa (ben formata). A loro volta, L_a e L_c rappresentano il linguaggio delle formule atomiche e delle formule complesse, rispettivamente.

Osservazione 8.3 (Linguaggio) La Definizione 8.1 del (linguaggio di rappresentazione) formalizza l'idea informale di linguaggio fornita nell'Intuizione 2.6.

Osservazione 8.4 (Funzione di interpretazione della rappresentazione) Una funzione di interpretazione della rappresentazione I estende una funzione di interpretazione I_A, cioè $I_A \subseteq I$, nel senso che si comporta allo stesso modo di I_A per tutte le affermazioni $a \in L_a$. Le funzioni di interpretazione delle rappresentazioni sono definite nella Definizione 8.4.

Osservazione 8.5 (Linguaggio di rappresentazione e modello del mondo) Dato un modello del mondo $\hat{W} =< L_A, D, I_A >$, la tupla $\hat{W} =< L, D, I >$ non è un modello del mondo per due motivi. Il primo è che, come dettaglieremo di seguito, L_a contiene formule (quelle in $L_a - L_A$) che non possono essere interpretate da una funzione di interpretazione. Il secondo è che L_c contiene formule (quelle in $L_c - L_a$) per le quali non esiste alcuna interpretazione. Come discusso nell'Osservazione 8.1 ed esemplificato nell'Esempio 8.1, l'obiettivo generale è fornire maggiore flessibilità nel modo in cui vengono descritti i modelli del mondo. Ciò si realizza in due modi: attraverso le formule atomiche estendendo il campo di applicazione delle funzioni di interpretazione, mantenendo comunque lo stesso dominio; e attraverso le formule complesse che elaborano linguisticamente sui contenuti del dominio. Leggendo il testo nell'Esempio 8.1 è facile scoprire che gran parte di quel testo ha una relazione solo marginale con l'immagine nella Figura 3.1.

Definizione 8.2 Dato un linguaggio di rappresentazione L, una teoria T è definita come

$$T = \{w\} \subseteq L \tag{8.2}$$

Osservazione 8.6 (Teoria) La Definizione 8.2 generalizza in modo ovvio la Definizione 5.4.

Osservazione 8.7 (Cosa rappresentano i linguaggi di rappresentazione) Come evidenziato anche dall'Esempio 8.1, l'esistenza dei linguaggi di rappresentazione è motivata dal fatto che ci sono informazioni che non possono essere facilmente e intuitivamente rappresentate in un modello del mondo. Per fornire alcuni esempi:

- *Informazioni negative.* I modelli del mondo di solito non rappresentano ciò che non è il caso nella parte del mondo rappresentata. E c'è effettivamente una buona ragione per questo. Per qualsiasi cosa che viene rappresentata, ci sono infinitamente molte cose che questa cosa non è. Quindi, ad esempio, se vedi una donna con i capelli biondi e varie caratteristiche, questa donna non è un uomo, non è un animale, non ha i capelli neri, non ha i capelli rossi, non è a casa, ecc.;
- *Informazioni parziali.* Percepite una persona da una certa distanza e non potete distinguere se è una donna o un uomo;
- *Informazioni conseguenziali.* Dal momento che vedete una donna, sapete, come conseguenza, che è anche una persona.
- *Informazioni equivalenti.* Dal momento che vedete una macchina gialla con una targa sul tetto, sapete che è un taxi.
- e molto altro.

Le descrizioni in lingua naturale come quella nell'Esempio 8.1 di solito integrano la descrizione del modello del mondo con molte informazioni aggiuntive che, si spera, faciliteranno la creazione di un modello mentale corretto e non divergente del modello del mondo che viene descritto. Il messaggio chiave è che i modelli del mondo sono rappresentazioni linguistiche dirette di rappresentazioni analogiche mentali. Come tali, difficilmente possono rappresentare informazioni che non sono esplicitamente codificate in una rappresentazione analogica, ma che riguardano piuttosto come gli osservatori si relazionano alle rappresentazioni analogiche.

Osservazione 8.8 (Cosa i linguaggi di rappresentazione non rappresentano) I linguaggi di rappresentazione descrivono i modelli del mondo, ma non catturano la pragmatica di come vengono utilizzate le descrizioni linguistiche, né il testo aggiuntivo utilizzato per imporre questa pragmatica, ad esempio: il fatto che piaccia ciò che si vede, se qualcuno è arrabbiato, se qualcuno sta cercando di convincere qualcun altro della veridicità di ciò che sta descrivendo e molto altro.

Terminologia 8.1 (Linguaggio di rappresentazione e funzione di interpretazione) Per semplicità, quando non sussiste confusione, nei seguenti parliamo di "linguaggio" e "funzione di interpretazione", intendendo rispettivamente "linguaggio di rappresentazione" e "funzione di interpretazione della rappresentazione".

Osservazione 8.9 (Formule atomiche) Le affermazioni sono formule atomiche, ma alcune formule atomiche non sono affermazioni. La proprietà chiave che le formule atomiche condividono con le affermazioni è che sono interpretate da una funzione di interpretazione. In altre parole, il significato delle formule atomiche, come quello delle affermazioni, può essere calcolato direttamente dal dominio.

Definizione 8.3 (Affermazioni atomiche, affermazioni complesse) Dato un linguaggio $L = L_a \cup L_c$, L_a è definito come

$$L_a = L_A \cup L_{AC} \tag{8.3}$$

dove: L_A è un linguaggio assertivo, che chiamiamo anche un linguaggio di affermazioni atomiche, e L_{AC} è un linguaggio di affermazioni complesse.

Osservazione 8.10 (Formule atomiche, operatori di conoscenza e dati) Le formule atomiche sono distinte in base all'uso di operatori che generano.

Esempio 8.2 (Formule atomiche, operatori di conoscenza) Possiamo costruire formule atomiche complesse nel seguente modo. Se C_1 e C_2 sono concetti, allora (seguendo la notazione del LOD), $C_1 \cap C_2$ è anche una descrizione di concetto, dove C_i può essere un'affermazione atomica così come un'affermazione complessa. Esempi di affermazioni in questo linguaggio sono (";" è utilizzato per separare diverse formule):

"being a person" ; "having blond hair" ; "having a dog";

"having blond hair" $\sqcap$ "being a person";

"being a person" $\sqcap$ "having blond hair" $\sqcap$"being a person"

... e così via, con formule atomiche complesse di lunghezza indefinita. Come dal LOD, l'interpretazione di $C_1 \cap C_2$ è l'intersezione delle interpretazioni di C_1 e C_2. Quindi, ad esempio, il concetto atomico nella seconda riga denota una persona con i capelli biondi.

Esempio 8.3 (Formule atomiche, operatori di dati) !!DA FARE!! FUNZIONI ESEMPIO CON "friendOf" e "nearTo"

Osservazione 8.11 (Formule complesse) A differenza delle formule atomiche, le formule complesse non sono interpretate da una funzione di interpretazione. L'idea è che le formule complesse consentano di comporre formule atomiche in articolazioni complesse più lunghe il cui significato può essere dedotto in qualche modo dal significato delle formule atomiche costituenti tramite implicazione, vedere di seguito.

Esempio 8.4 (Formule complesse) Possiamo costruire formule atomiche complesse come segue. Se A_1 e A_2 sono formule, allora (seguendo la notazione di LOP), A_1 xor A_2 è anche una formula in cui A_i può essere una formula atomica o complessa. Esempi di affermazioni in questo linguaggio sono (";" viene utilizzato per separare diverse formule):

"Sofia è una persona" ;

"Sofia è una persona" xor "Sofia è una persona";

"Sofia è una persona" xor "Paolo è un uomo" ;

("Sofia è una persona" xor "Paolo è un uomo") xor "Paolo è un cane" ;

. . . e così via, con formule complesse di lunghezza indefinita. L'intuizione è che A_1 xor A_2 contiene un'unica e sola informazione tra le informazioni denotate da A_1 e A_2.

Esempio 8.5 (Linguaggi di rappresentazione) Di seguito sono riportati esempi di linguaggi di rappresentazione:

1. Tutte le lingue naturali, come utilizzate dalle persone nella vita di tutti i giorni;

2. Il linguaggio dell'aritmetica, che descrive come effettuare operazioni di addizione e sottrazione su numeri naturali. Il linguaggio dell'aritmetica è una lingua naturale semplificata che consente di menzionare, tra le altre cose, numeri, variabili, più, meno, moltiplicazione e la possibilità di comporre frasi più complesse;

3. I linguaggi dei database relazionali (DB) non si estendono ai linguaggi di rappresentazione;

4. I linguaggi entità-relazione (ER) non si estendono ai linguaggi di rappresentazione.

Osservazione 8.12 (Linguaggi basati su grafi e linguaggi di rappresentazione) I linguaggi basati su grafi (ad esempio, ER o linguaggi di database) in generale sono difficili da estendere ai linguaggi di rappresentazione a causa della difficoltà intrinseca nel mantenere l'intuitività, che è la caratteristica chiave dei linguaggi basati su grafi. I linguaggi di rappresentazione si adattano più facilmente ai linguaggi che hanno una struttura sequenziale simile a quella delle lingue naturali.

8.3 Funzione di interpretazione

Definizione 8.4 (Funzione d'interpretazione) Dato un linguaggio $L = L_a \cup L_c$, con $L_a = L_A \cup L_{AC}$. Sia D un dominio. Allora una Funzione di Interpretazione I per L_a è definita come segue:

$$I : L_a \to D \ (I \subseteq L_a \times D) \tag{8.4}$$

con

$$I = I_A \circ I_{AC} \tag{8.5}$$

dove

$$\begin{aligned} I_{AC} &: L_a \to L_A \\ I_A &: L_A \to D \end{aligned} \tag{8.6}$$

dove: I_A è una funzione di interpretazione per affermazioni atomiche e I_{AC} è una funzione di interpretazione per affermazioni complesse. I_A è come definita nella Definizione 5.5 e 6.8. Inoltre, diciamo che un fatto $f \in M$ è l'interpretazione di $w \in L_a$, e scriviamo

$$f = I(a) = a^I \tag{8.7}$$

per indicare che w è una descrizione linguistica di f. Diciamo che f è l'interpretazione di w, o, equivalentemente, che w denota f.

Osservazione 8.13 (Funzione di interpretazione) La Definizione 8.4 estende la Definizione 5.5 per applicarsi alle formule atomiche che non sono affermazioni. Tutte le considerazioni fatte nella Sezione 5.3 si applicano.

8.4 Entailment

Il significato dei linguaggi di rappresentazione è calcolato riducendo il significato delle formule complesse a quello delle loro formule atomiche costituenti.

Definizione 8.5 (Relazione di implicazione) Siano $M \subseteq D$ e $w \in L$ una formula. Allora, $\models$, da leggersi come "implica", è una relazione di implicazione definita come

$$\models \subseteq D \times L \tag{8.8}$$

Scriviamo anche

$$M \models_L T \quad (M \models_L w) \tag{8.9}$$

con $\{M\}$ che rappresenta un insieme di modelli M e T un insieme di formule w. $M \models T$ sta per $M \models w$ per ogni $w \in T$. La notazione per il linguaggio L viene omessa quando non è necessaria. Diciamo che M implica w, o anche che M implica T.

Definizione 8.6 (Implcazione di una formula atomica) Se w è una formula atomica, allora abbiamo

$$M \models w \text{ se e solo se } I(w) \in M \tag{8.10}$$

Osservazione 8.14 (Implicazione di formule atomiche) L'implicazione di formule atomiche si riduce alla loro interpretazione.

Osservazione 8.15 (Implicazione di formule complesse) L'implicazione di formule complesse opera in due passaggi, in modo simile a come le funzioni di interpretazione operano sulle formule atomiche complesse. Nel primo passaggio, riduce l'implicazione di una formula complessa a quella delle sue formule atomiche componenti. Nel secondo passaggio, applica la funzione di interpretazione alle formule atomiche.

Osservazione 8.16 (Relazione di implicazione) L'interpretazione è una funzione. L'implicazione è una relazione. È una relazione uno-a-molti. Può esserci un numero multiplo di formule atomiche e/o complesse che denotano un fatto e, simmetricamente, per la stessa formula ci possono essere fatti multipli implicati da essa (l'ultima proprietà è quella che rende l'implicazione una relazione).

Osservazione 8.17 (Relazione di implicazione e funzione di interpretazione) La differenza chiave tra l'interpretazione come funzione e l'implicazione come relazione è che con l'implicazione permettiamo a una formula (teoria) di avere significati multipli, cioè di denotare fatti multipli (insiemi di fatti), mentre l'interpretazione non consente affermazioni polisemiche. La motivazione chiave di ciò è che certe logiche, in particolare quelle utilizzate per formalizzare la presa di decisioni, ad esempio LOP, consentono di modellare la conoscenza parziale, ovvero il fatto che una persona non abbia conoscenza completa del mondo, una situazione intrinseca alla conoscenza umana. Nel caso della conoscenza parziale, una formula non presente nel modello può essere (non sempre) indifferentemente considerata vera o falsa. Vedere anche l'Osservazione 8.18.

Osservazione 8.18 (Relazione di implicazione e funzione di interpretazione, formula atomica e complessa) Si potrebbe pensare di distinguere tra fatti atomici e complessi e dire che sono denotati rispettivamente da formule atomiche e complesse. Non facciamo questo passo perché non esiste una cosa come un fatto complesso. I fatti sono nel mondo. Ciò che è descritto da una formula complessa non è nel mondo, è nella mente, è in come una persona descrive il mondo. Pensate, ad esempio, a una formula che afferma la mia conoscenza parziale sulla situazione attuale, ad esempio il fatto che sulla tavola ci sia una penna o una matita (dove dico questo semplicemente perché non sono abbastanza vicino e non posso vedere chiaramente). Nel mondo non esiste una penna o una matita. Sulla tavola c'è o una penna o una matita, dato che posso percepire chiaramente che c'è un solo oggetto!

Definizione 8.7 (Interpretazione e implicazione) Se un'interpretazione I è un modello per una teoria T (o una formula w), allora diciamo che I implica T (o w) e scriviamo

$$I \models T \quad (I \models w) \tag{8.11}$$

Esempio 8.6 (Formule complesse) Consideriamo le formule complesse come definite nell'Esempio 8.4, cioè formule della forma A_1 xor A_2, dove A_i è qualsiasi formula. Supponiamo che A_1 e A_2 siano formule atomiche. Allora A_1 xor A_2 sarà denotato da un modello M che contiene la denotazione di A_1 o da uno che contiene la denotazione di A_2. In formule:

$$\begin{array}{ll} I(A_1) & \models A_1 \\ I(A_1) & \models A_1 \mathrm{xor} A_2 \\ I(A_1) & \not\models A_1 \mathrm{xor} A_2 \\ \{I(A_1), I(A_2)\} & \not\models A_1 \mathrm{xor} A_2 \end{array}$$

L'esempio sopra mostra quanti modelli possono denotare le stesse formule (prime tre equazioni) e anche come la stessa formula può denotare modelli multipli (prima e ultima equazione).

L'implicazione è come formalizziamo il ragionamento (umano).

Definizione 8.8 (Logical entailment) Siano $M \subseteq D$ un modello e $T1, T2 \subseteq L$ due teorie e $w \in L$ una formula. Allora scriviamo

$$T_1 \models_{\{M\}} T_2 \quad (T_1 \models_{\{M\}} w) \tag{8.12}$$

e diciamo che $T1$ (logicamente) implica $T2$ (w) rispetto all'insieme di modelli M se

$$\text{per tutti } M \in \{M\}, \quad \text{se } M \models T_1 \quad \text{dunque} \quad M \models T_2 \quad (M \models w)$$

Terminologia 8.2 (Implicazione logica v.2, v.3) A volte, l'implicazione logica è definita come

$$T_1 \models T_2 \quad (T_1 \models w) \tag{8.13}$$

per indicare che ogni modello $M \subseteq D$, cioè

$$\text{per ogni } M \subseteq D, \quad \text{se } M \models T_1 \quad \text{quindi} \quad M \models T_2 \quad (M \models w)$$

A volte, l'implicazione logica è definita come

$$T_1 \models_T T_2 \quad (T_1 \models_T w) \tag{8.14}$$

per significare tutti i modelli che implicano T, cioè

$$\text{per ogni M tale che } M \models T, \quad \text{se } M \models T_1 \quad \text{quindi} \quad M \models T_2 \quad (M \models w)$$

Utilizzeremo tutte queste definizioni in modo interscambiabile, poiché rappresentano diverse formalizzazioni della stessa nozione.

Osservazione 8.19 (Implicazione logica) L'implicazione logica deve essere interpretata nel seguente modo. Prendi uno o più modelli M. Quindi mantieni tutti i modelli che implicano $T1$ e scarta gli altri. Se tutti questi modelli implicano anche $T2$, allora $T1$ implica $T2$. Nota il ruolo chiave di $T1$ nell'obbligare l'esclusione dei modelli in cui non vale. $T1$ focalizza solo su un insieme preciso di modelli.

Osservazione 8.20 (Implicazione logica e ragionamento) L'implicazione logica è la formalizzazione del processo di ragionamento. Iniziamo con alcune ipotesi, che modelliamo come uno o più modelli, e verifichiamo se, sotto queste stesse ipotesi, vale anche $T2$. Questo è esattamente come ragioniamo intuitivamente. Nota che il viceversa non vale. Vale a dire, se $M \not\models T1$, allora potremmo avere $M \models T2$ o $M \not\models T2$.

Esempio 8.7 (Ragionamento tramite implicazione) Ad esempio, si potrebbe voler dedurre dal fatto che se piove, le persone non escono di casa e dal fatto che sta piovendo, il fatto che le persone non escono di casa. O che se i miei capelli sono biondi, non sono neri. O che essere vicino a te esclude essere lontano da te. E così via. Naturalmente, il tipo di conclusioni che si possono trarre dipende dalle specifiche del modo in cui è definita la relazione di implicazione. La definizione delle logiche che si concentrano su tipi specifici di ragionamento sarà l'argomento delle sezioni seguenti.

Terminologia 8.3 (Implicazione, assioma, teorema) Considera la seguente forma di implicazione: $T1 \models w$. Storicamente, il insieme finito di formule $w \in T$ viene chiamato assiomi, mentre le loro infinite conseguenze logiche vengono chiamate teoremi. Gli assiomi sono garantiti di essere veri, essere delle informazioni a priori nel nostro caso, ad esempio, rappresentate da una teoria $\hat{KG}$.

Terminologia 8.4 (Ragionamento deduttivo) Il tipo di ragionamento implementato dall'implicazione logica è chiamato ragionamento deduttivo.

Terminologia 8.5 (Ragionamento, in avanti e all'indietro, obiettivo) Il ragionamento in avanti è il processo di ragionamento con cui si cerca di dimostrare un teorema, chiamato l'obiettivo, derivando conseguenze logiche dagli assiomi. In modo duale, con il ragionamento all'indietro si inizia con l'obiettivo e si cerca di ridurre il possesso dell'obiettivo al possesso degli assiomi, che si sa essere veri.

Osservazione 8.21 (Ragionamento, in avanti e all'indietro) All'inizio dell'arte, nell'ambito dell'Intelligenza Artificiale, tutti i motori di inferenza operano all'indietro. La ragione è che in questo modo il motore di inferenza può sfruttare le informazioni codificate nella struttura dell'obiettivo. Indipendentemente dalle specifiche di una relazione di implicazione e in modo simile alle funzioni di interpretazione, si richiede che le relazioni di implicazione soddisfino un certo insieme di principi che vincolano il comportamento desiderato del ragionamento. Siano $\Gamma = \{w\}$, $\Sigma = \{w\}$ insiemi di formule e w, w_1, w_2 siano formule.

Intuizione 8.1 (Riflessività)

$$w \models w \tag{8.15}$$

Osservazione 8.22 (Riflessività) Ogni fatto implica se stesso. La conoscenza afferma se stessa come conoscenza. Questa è l'essenza di ciò che riguarda la conoscenza.

Intuizione 8.2 (Taglio)

$$\text{Se } \Gamma \models w_1 \text{ e } \Sigma \cup \{w_1\} \models w_2, \text{ allora } \Gamma \cup \Sigma \models w_2 \tag{8.16}$$

Osservazione 8.23 (Taglio) Esistono due modi di interpretare il taglio. Il primo e più comune è che il ragionamento può essere reso efficiente eliminando risultati intermedi non rilevanti. Il secondo è la transitività, cioè il fatto che il ragionamento può essere composto concatenando sessioni di ragionamento indipendenti, qualcosa che le persone fanno continuamente nella vita di tutti i giorni.

Intuizione 8.3 (Compattezza)

$$\text{Se } \Gamma \models w, \text{ allora esiste un sottoinsieme finito } \Gamma_0 \subseteq \Gamma \text{ tale che } \Gamma_0 \models w \quad (8.17)$$

Osservazione 8.24 (Compattezza) Consideriamo l'infinità come la possibilità di descrivere un altro fatto nel processo di ragionamento. Ad esempio, i numeri naturali sono infiniti, e non importa quanti numeri siano stati utilizzati finora, è sempre possibile fornirne uno nuovo. La compattezza afferma che la conseguenza logica deve essere calcolata utilizzando un insieme finito di assunzioni. La conseguenza logica per un insieme ipoteticamente infinito di formule non è un comportamento considerato di interesse.

Intuizione 8.4 (Monotonia)

$$\text{Se } \Gamma \models w, \text{ allora } \Gamma \cup \Sigma \models w \quad (8.18)$$

Osservazione 8.25 Monotonia La monotonia implementa una proprietà fondamentale ed intuitiva della conoscenza, ad esempio della conoscenza scientifica. Se la conoscenza aumenta, ciò che può essere derivato da essa attraverso il ragionamento può solo aumentare. Al massimo può rimanere invariato se il nuovo pezzo di conoscenza era implicito in ciò che è già noto.

Intuizione 8.5 (Non Monotonia)

$$\text{Se } \Gamma \models w \text{ e } \Gamma \cup \Sigma \not\models w \quad (8.19)$$

Osservazione 8.26 (Non Monotonia) Nonostante la sua intuitività, la monotonia è una proprietà che nella maggior parte dei casi non si verifica. Questo è ampiamente il caso del ragionamento di buon senso, un argomento ampiamente studiato nell'ambito dell'Intelligenza Artificiale. Quante volte il fatto di apprendere qualcosa di nuovo ci ha costretti a cambiare idea? Un esempio storico di ragionamento di buon senso non monotonico nell'ambito dell'IA è che la credenza che tutti gli uccelli volino può essere smentita dal fatto che i pinguini sono uccelli e non volano. La conoscenza scientifica è ricca di esempi in cui le conoscenze precedenti sono state smentite da prove successive, e molte teorie nella filosofia della scienza sono state costruite su questo principio. L'esempio standard della non monotonia della conoscenza scientifica è la scoperta che è la Terra a ruotare attorno al Sole, e non il contrario. Dal punto di vista pratico, tutte le logiche che formalizzano il ragionamento matematico e vengono utilizzate nei metodi formali, ad esempio nei linguaggi di programmazione, sono logiche monotoniche, mentre la maggior parte delle logiche definite nell'ambito dell'IA sono non monotoniche. La negazione per mancato successo, come implementata nei database relazionali, è non monotonica.

8.5 Logica

Il viaggio è completo. Dobbiamo solo mettere insieme tutto.

Osservazione 8.27 **(I ruoli di D, L, I, $\models$, M, T)** Le definizioni fornite nelle sezioni precedenti possono essere riassunte nella seguente figura.

$$\begin{tikzcd}
a \arrow[d, "\mathcal{I}"] & \mathcal{T} \arrow[l, "\in"'] \arrow[r, "\subseteq"] \arrow[d, dashed, "\models"] & \mathcal{L}_a \arrow[d, "\mathcal{I}"] \\
\mathtt{f} & \mathtt{M} \arrow[l, "\in"'] \arrow[r, "\subseteq"] & \mathtt{D}
\end{tikzcd} \tag{8.20}$$

con $L = L_a \cup L_c$.

Confronta anche con la Figura 5.7 per i modelli del mondo. Nell'Equazione (8.20), D definisce l'insieme dei fatti f di potenziale interesse, L_a l'insieme delle affermazioni a di potenziale interesse, L le formule w che possiamo utilizzare per descrivere i fatti in M, $\models$ le condizioni in base alle quali qualsiasi $w \in L$ è implicato da M, M è l'insieme di fatti su cui ci stiamo concentrando e, infine, T è la teoria che descrive M. La differenza chiave rispetto all'equazione corrispondente del modello del mondo è che qui la relazione tra i modelli è stata estesa dalla funzione di interpretazione alla relazione di implicazione.

Definizione 8.9 **(Logica)** Prendi qualsiasi rappresentazione come da Equazione (8.20). Allora

$$\hat{L} = \langle L, D, I, \models \rangle \tag{8.21}$$

è una logica.

Osservazione 8.28 **(Logica)** Nell'Equazione (8.20), i componenti di una logica, cioè D, L, I, $\models$, definiscono le regole generali che vengono seguite nella definizione dei meccanismi per la rappresentazione e il ragionamento sui modelli del mondo, cioè le rappresentazioni. Sono definiti a priori, estendendo i modelli del mondo, di solito da esperti di rappresentazione della conoscenza e logica, come strumenti generali da utilizzare da parte dei professionisti. Forniscono l'infrastruttura generale di modellazione e ragionamento che consente di rappresentare e ragionare sulle rappresentazioni dei problemi del mondo reale. Forniscono anche i meccanismi attraverso i quali è possibile confrontare e, eventualmente, unire due rappresentazioni. Gli sviluppatori di software solitamente studiano questi modelli durante corsi di informatica o intelligenza artificiale e li utilizzano così come sono nello sviluppo di sistemi; pensate, ad esempio, all'ampio utilizzo di modelli ER e UML, così come di logiche.

Osservazione 8.29 **(Logica e Modello del Mondo)** Se $\hat{L} = \langle L, D, I, \models \rangle$ e $L = L_A$, allora $\hat{L} = \hat{W}$ con $\hat{W} = \langle L_A, D, I_A \rangle$. In effetti, la relazione di implicazione si riduce alla funzione di interpretazione, vedere Equazione (8.11).

Osservazione 8.30 (Da rappresentazioni mentali a modelli del mondo a logiche) I modelli del mondo forniscono meccanismi generali per generare rappresentazioni non ambigue, cioè teorie di rappresentazioni mentali. Le logiche forniscono i meccanismi per ragionare sulle teorie e sui modelli che descrivono.

Osservazione 8.31 (Definire un modello tramite una teoria) Il modo più comune per modellare il mondo è definire un insieme di affermazioni, ciò che chiamiamo una teoria. In altre parole, costruiamo un modello M selezionando un sottoinsieme T di L. Questo è l'approccio comune quando il compito consiste nella rappresentazione da zero di una parte specifica del mondo di interesse.

Tuttavia, talvolta, ci viene data una teoria predefinita T e un modello predefinito M e ci viene chiesto come sono correlati. In tal caso, abbiamo quanto segue.

Definizione 8.10 (Correttezza e completezza di una teoria T rispetto a un modello M) Sia $L^{\wedge} =< L, D, I, \models>$ una logica. Siano $T \subseteq L$ e $M \subseteq D$ una teoria e un modello, rispettivamente. Allora abbiamo due situazioni possibili:

- Correttezza: $M \models T$, in tal caso diciamo che T è corretta rispetto a M, o che M è un modello per T.
- Completezza: Se, per tutti i fatti $f \in M$, esiste una formula $w \in L_A$ tale che $I(w) = f$, in tal caso diciamo che T è completa rispetto a M.

Le nozioni di incorrettezza e incompletezza sono definite in modo ovvio.

Chapter 9

Logica - Rappresentazione intenzionale

Dobbiamo fornire i dettagli su come calcolare l'implicazione, partendo da una descrizione di come L e I sono costruiti.

9.1 Dominio

L'obiettivo è avere un linguaggio più espressivo, se confrontato con quello di un modello del mondo, per descrivere un dominio di interpretazione, lasciando quest'ultimo invariato. Vedere la Sezione 6.1.

9.2 Linugaggio di rappresentazione

Definizione 9.1 (Linguaggio, rappresentazione intensionale) Sia $L = L_a \cup L_c$ un linguaggio. La sua rappresentazione intensionale è data da

$$L^i = \langle L_a^i, L_c^i \rangle \tag{9.1}$$

dove L_a^i è il linguaggio delle formule atomiche, definito in modo intensionale, e L_c^i è il linguaggio delle formule complesse, definito in modo intensionale. L_a^i e L_c^i sono anch'essi coppie.

Definizione 9.2 (Linguaggio di formule atomiche, rappresentazione intensionale) Sia $L^i = \langle L_a^i, L_c^i \rangle$ un linguaggio definito in modo intensionale. Allora la rappresentazione intensionale di L_a^i è data da

$$L_a^i = \langle A_a^i, \{FR\}_a^i \rangle \tag{9.2}$$

dove: A_a^i è l'alfabeto di L_a^i e $\{FR\}_a^i$ è l'insieme di regole di formazione per L_a^i con

$$A_a^i = \langle E, \{C\}, \{R\} \rangle$$
$$L_a^e = \{w : w \in C(\{FR\}_a, A_a)\}$$

dove

- E è un insieme di (nomi di) entità,
- $\{C\}$ è un insieme di concetti, dove un concetto è il nome di una classe,
- $\{P\}$ è un insieme di proprietà, dove una proprietà è il nome di una relazione, e infine,
- $C(\{FR\}_a, A_a)$ è la chiusura transitiva di $\{FR\}_a$ su A_a.

Definizione 9.3 (Regola di formazione) i limitiamo a linguaggi con grammatiche context-free (grammatiche libere dal contesto). Di conseguenza, prendiamo $\{FR\}_a = \{R_a\}$, dove ciascuna regola di formazione R_a ha la forma

```
<expression> ::= -expression-
```

dove, seguendo la notazione[1] BNF:

- <expression> è un'espressione non terminale, e le espressioni non terminali sono racchiuse tra parentesi angolari ¡¿.
- I simboli che non appaiono sul lato sinistro di una regola sono chiamati terminali.
- -expression- è composta da una o più sequenze di simboli, sia terminali che non terminali.
- ::= permette a <expression> di essere sostituito con una sequenza che si trova in -expression-.
- Le sequenze in -expression- sono separate dalla barra "|", che indica una scelta nella sostituzione.

Osservazione 9.1 (Regola di formazione generica) Una generica regola di formazione R_{ia} può essere visualizzata come

$$\mathtt{<NT>\ ::=\ <NT_1>\ |\ \ldots\ |\ <NT_1>\ |\ T_1\ |\ \ldots\ |\ T_m|}$$

con la possibilità di nessuna occorrenza di sequenze terminali o non terminali.

Osservazione 9.2 (Chiusura transitiva) $C(\{FR\}_c, L_a)$ è l'insieme minimo di formule che possono essere ottenute applicando ricorsivamente le regole di

$\{FR\}_c$ ai loro stessi risultati, partendo da L_a. Le formule atomiche sono "scatole nere" per $\{FR\}_c$, nel senso che le regole in $\{FR\}_c$ possono comporle in formule complesse, ma non possono modificarne la struttura interna.

Osservazione 9.3 (Infinità della chiusura transitiva) Le regole di formazione, come da Definizione 9.3, sono ricorsive, vedi anche Osservazione 9.1, nel senso che possono esserci non terminali che appaiono sia a sinistra che a destra di una regola di produzione. Ciò apre la possibilità di elementi di lunghezza indefinita. Quindi, l'insieme $C(\{FR\}_c, L_a)$ nel caso di regole di produzione ricorsive, come è spesso il caso, è infinito.

Osservazione 9.4 (Alfabeto) L'alfabeto A_a è infinito.

Esempio 9.1 (Linguaggio di formule atomiche, regole di generazione della conoscenza) Considera il linguaggio definito nell'Esempio 8.2 che consente formule complesse atomiche della forma $C_1 \wedge C_2$ dove C_i è un concetto. La BNF che genera questo linguaggio è composta dalle seguenti due regole di formazione:

`<awff> ::= <concept>`

`<awff> ::= <awff>` $\sqcap$ `<awff>`

Dove ¡concept¿ è un simbolo non terminale che rappresenta qualsiasi elemento C dell'alfabeto. $\sqcap$ è un simbolo terminale che, come tale, non può essere ulteriormente scomposto.

Esempio 9.2 (Linguaggio di formule atomiche, regole di generazione dei dati) TODO by prf

Osservazione 9.5 (Applicazione delle regole di formazione, generazione, riconoscimento) Le regole di formazione possono essere applicate per due motivi. Il primo, come nel caso dell'Esempio 9.1 e dell'Esempio 9.2, è nel processo di generazione della chiusura; il secondo è nel processo di riconoscimento per determinare se una certa formula di input appartiene alla chiusura. Le due serie di regole di formazione, sebbene catturino la stessa intuizione, lavorano in direzioni opposte, la prima partendo dai terminali e sviluppando elementi sempre più complessi della chiusura (generando, nel caso di regole di produzione ricorsive, un insieme infinito), la seconda scomponendo l'input complesso fino agli elementi terminali. Il fallimento in questo processo significa che l'input non appartiene alla chiusura. Questo processo verrà utilizzato di seguito nella definizione della funzione di interpretazione e della relazione di implicazione.

Esempio 9.3 (Regole di formazione, generazione e riconoscimento) Le seguenti due regole di formazione per la generazione della chiusura:

`<awff> ::= <concept>`

`<awff> ::= <awff>` $\sqcap$ `<awff>`

possono essere riscritte come le seguenti regole di riconoscimento per la stessa chiusura:

$$\begin{aligned} &W(\texttt{<awff>}) && ::= W(\texttt{<concept>}) \\ &W(\texttt{<awff1>} \sqcap \texttt{<awff2>}) && ::= W(\texttt{<awff1>}) \text{ e } W(\texttt{<awff2>}) \end{aligned}$$

Dove W verifica la correttezza dell'input della formula. W restituisce "true" se il processo di riconoscimento si conclude correttamente e "false" altrimenti.

Definizione 9.4 (Linguaggio di formule complesse, regole di formazione) Sia $L^i = \langle L_a^i, L_c^i \rangle$ un linguaggio definito in modo intenzionale. Allora la rappresentazione intenzionale di L_c^i è data da

$$L_c^i = \langle L_a^e, \{FR\}_c \rangle \tag{9.3}$$

dove $\{FR\}_c$ è l'insieme di regole di formazione per L_c^e con

$$L_c^e = \{w : w \in C(\{FR\}_c, L_a^e)\}$$

dove: L_{ea} è come definito in Definizione 9.2, e $C(\{FR\}_c, L_a^e)$ è la chiusura transitiva di $\{FR\}_c$ su L_a.

Esempio 9.4 (Linguaggio di formule complesse, rappresentazione intenzionale) Considera il linguaggio definito nell'Esempio 8.6 che consente formule complesse atomiche della forma A_1 xor A_2 dove A_i è una qualsiasi formula. La BNF che genera questo linguaggio è composta dalle seguenti due regole di formazione:

```
<wff> ::= <awff>
<wff> ::= <wff> xor <wff>
```

Dove <awff> è un simbolo non terminale che può essere istanziato con qualsiasi formula atomica in L_a.

9.3 Funzione di interpretazione

Definizione 9.5 (Funzione di interpretazione, rappresentazione intenzionale) Sia $L = L_a \cup L_c$ un linguaggio con $L_a = L_A \cup L_{AC}$. Sia la funzione di interpretazione $I : L_{ia} \to D$ definita come $I = I_A \circ I_{AC}$, con $I_{AC} : L_a \to L_A$ e $I_A : L_A \to D$ (Definizione 8.4). Allora, la rappresentazione intenzionale di I è data da

$$I^i = \langle L_a, \{FR\}_I \rangle \tag{9.4}$$

dove $\{FR\}_I$ è l'insieme di regole di formazione per I^e con

$$I^e = \{< w, f >: w \in L_a, f \in D, < w, f > \in C(\{FR\}_I, L_a)\}$$

dove $C(\{FR\}_I, L_a)$ è la chiusura transitiva di $\{FR\}_I$ su L_a.

Osservazione 9.6 (Funzione di interpretazione, regole di riconoscimento) Le regole di riconoscimento per la funzione di interpretazione $R_I \in \{FR\}_I$ sono definite come segue:

$$\begin{aligned} I_{AC}(T) \quad & ::= I_A(< T >) \\ I_{AC}(< NT >) \quad & ::= I_{AC}(< NT1 >)| \ldots |I_{AC}(< NT1 >) \end{aligned} \tag{9.5}$$

dove $< NT >$ sono simboli non terminali per I_{AC}, T sono simboli terminali per I_{AC} e $< T >$ sono simboli non terminali per I_A.

Osservazione 9.7 (Funzione di interpretazione, regole di riconoscimento) Nelle regole di riconoscimento della funzione di interpretazione, I_{AC} si applica ai non terminali, mentre I_A si applica ai terminali di I_{AC}, che sono effettivamente non terminali per I_A. In questo contesto, vedere la prima regola di produzione nell'Equazione (9.5), dove la stessa espressione T viene trasformata da terminale di I_{AC} a non terminale di I_A.

Esempio 9.5 (Funzione di interpretazione, regole di riconoscimento, operatori di dati) TODO by prof

Esempio 9.6 (Formule atomiche, regole di riconoscimento, operatori di conoscenza) Considera l'insieme di formule definite nell'Esempio 8.2, che sono nella forma $C_1 \sqcap C_2$, dove C_i può essere una affermazione atomica così come un'affermazione complessa. Considera le regole di formazione che le generano. Formalizziamo l'intuizione che $C_1 \sqcap C_2$ denoti l'intersezione delle interpretazioni di due affermazioni atomiche come segue:

$$\begin{aligned} I_{AC}(< \text{concept} >) \quad & ::= I_A(< \text{concept} >) \\ I_{AC}(< \text{awff} > \sqcap < \text{awff} >) \quad & ::= I_{AC}(< \text{awff} >) \cap I_{AC}(< \text{awff} >) \end{aligned}$$

Le sequenze di riscrittura possono essere implementate come segue:

$$I(C1 \sqcap C2) = I_{AC}(C1 \sqcap C2) = I_A(C1) \cap I_A(C2) = C1 \cap C2$$

Pertanto, ad esempio:

$$\begin{aligned} & I((person \sqcap woman) \sqcap dog) = \\ & I_{AC}((person \sqcap woman) \sqcap dog) = \\ & I_{AC}(person \sqcap woman) \cap I_A(dog) = \\ & I_A(person) \cap I_A(woman) \cap dog = \\ & (person \cap woman) \cap dog = \\ & woman \cap dog = \emptyset \end{aligned}$$

dove abbiamo supposto di sapere che le donne sono persone e i cani sono disgiunti dalle persone.

Osservazione 9.8 (Applicazione delle regole di formazione dell'interpretazione) Seguendo quanto menzionato nell'Osservazione 6.11, l'Equazione (6.7) mostra come I venga applicato in modo ricorsivo applicandolo ai componenti della sua affermazione di input fino alle affermazioni. In questo processo, i suoi componenti vengono applicati secondo necessità, ovvero I_e alle entità, I_C ai concetti e I_P alle proprietà. L'Esempio 9.6 mostra (seconda riga) come il processo menzionato nell'Osservazione 6.11 si generalizzi alle formule atomiche complesse. L'idea generale è che I_{AC} viene applicato finché non si arriva agli elementi singoli dell'alfabeto, dove, in quel momento, viene applicata la corretta funzione di interpretazione del componente.

Osservazione 9.9 (Annidamento delle regole di formazione) Il processo evidenziato sopra può essere annidato a qualsiasi livello. In effetti, I_A potrebbe essere nuovamente un insieme di regole di formazione che consentono regole atomiche complesse più raffinate e così via, per qualsiasi livello di annidamento. Ciò consente la generazione di manipolazioni di modelli del mondo sempre più complesse.

Esempio 9.7 (Formule atomiche, regole di formazione, operatori di dati) [DA COMPLETARE]

9.4 Entailment

Definizione 9.6 (Relazione di implicazione, rappresentazione intenzionale) Sia $M \subseteq D$ un modello e $T \subseteq L$ una teoria. Sia la relazione di implicazione definita come

$$\models \; \subseteq M \times T$$

come da Definizione 8.5. Allora, la rappresentazione intenzionale di $\models$ è

$$\models^i = \langle L, D, \{FR\}_{\models} \rangle \tag{9.6}$$

dove $\{FR\}_{\models}$ è l'insieme di regole di formazione per $\models_e$, con

$$\models^e = \{< f, w >: f \in M, w \in L, < f, w > \in C(\{FR\}_{\models}, D, L)\}$$

dove $C(\{FR\}_{\models}, L)$ è la chiusura transitiva di $\{FR\}_{\models}$ su L.

Definizione 9.7 (Relazione di implicazione, regola di formazione) Una regola di formazione per la relazione di implicazione $R \models \in \{FR\}_{\models}$ è definita come:

$$\begin{array}{ll} M \models T & ::= I(< T >) \\ M \models < NT > & ::= M \models < NT_1 > | \ldots | M \models (< NT_1 >) \end{array} \tag{9.7}$$

dove $< NT >$ sono simboli non terminali per $\models$, T sono simboli terminali per $\models$, e $< T >$ sono simboli non terminali per I.

Osservazione 9.10 (Relazione di implicazione, regole di formazione) Leggi Osservazione 9.7. Le stesse considerazioni si applicano, mutatis mutandis.

Esempio 9.8 (Implicazione) Prendi L_a, D, I come da Esempio 9.5. Prendi $L = L_a \cup L_c$ per essere il linguaggio di rappresentazione definito in 9.4 (con L_a uguale a quanto in Esempio 9.5), dove le regole di formazione per L_{ic} sono come segue (come da Esempio 9.4 riportato qui per completezza):

$$
\begin{aligned}
&< wff > ::=< awff > \\
&< wff > ::=< wff > \text{ xor } < wff >
\end{aligned}
$$

Quindi definiamo la relazione di implicazione con le seguenti regole di riconoscimento:

$$
\begin{aligned}
&M \models < wff > ::= I(< awff >) \\
&M \models < wff1 > \text{ xor } < wff2 > \quad ::= M \models < wff1 > \text{ and } M \not\models < wff2 > | \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad M \not\models < wff1 > \text{ and } M \models < wff2 >
\end{aligned}
$$

Abbiamo i seguenti esempi (dove a, a_i sono formule atomiche).

$$
\begin{aligned}
&M \models a \text{ se } I(a) \in M \\
&M \not\models a \text{ se } I(a) \notin M \\
&M \models a_1 \text{ xor } a_2 \text{ se } I(a_1) \in M \text{ and } I(a_2) \notin M | I(a_1) \in M \text{ and } I(a_2) \notin M \\
&M \models \{w_1, w_2\} \text{ se } M \models w_1 \text{ and } M \models w_2
\end{aligned}
$$

Osservazione 9.11 (Teoria) La nozione di teoria, come da Definizione 8.2, e la sua caratterizzazione come un insieme di assiomi, come da Terminologia 8.3, non cattura l'idea di una teoria costituita da tutti i teoremi, cioè tutte le affermazioni che sono considerate valide come parte della chiusura rispetto alle regole di formazione di implicazione.

Definizione 9.8 (Teoria, presentazione finita) Una teoria è un insieme di formule chiuso sotto la conseguenza logica, cioè T è una teoria se e solo se $T \models w$ implica che $w \in T$. L'insieme di assiomi T_0 da cui T è generato tramite implicazione è chiamato presentazione finita di T. Le due nozioni di teoria vengono usate in modo interscambiabile. Tuttavia, non tutte le teorie possono essere generate da un insieme finito di assiomi.

Definizione 9.9 (Teoria, assiomatica finita) Una teoria T è finitamente assiomatizzabile se può essere generata da un insieme finito di assiomi.

9.5 Logica

Definizione 9.10 (Logica, rappresentazione intenzionale) Sia $W^i = \langle L_a^i, D^i, I_a^i \rangle$ un modello del mondo con $D^i = \langle E, \{C\}, \{R\} \rangle$. Allora, sia

$$\hat{L} = \langle L, D, I, \models \rangle$$

una logica definita per lo stesso dominio di interpretazione D_i di $\hat{W^i}$. Quindi, la rappresentazione intenzionale $\hat{L}^i$ di $\hat{L}$ è definita come:

$$\hat{L}^i = \langle L^i, D^i, I^i, \models \rangle, \text{ dove } \; L^i = \langle L_a^i, L_c^i \rangle \tag{9.8}$$

e

$$\begin{aligned} L_a^i &= \langle A_a, \{FR\}_a \rangle \\ L_c^i &= \langle L_a^e, \{FR\}_c \rangle \\ I^i &= \langle L_a^e, \{FR\}_I \rangle \\ \models^i &= \langle L^e, \{FR\}_\models \rangle \end{aligned}$$

$\hat{L}^i, L_a^i, L_c^i, I^i, \models^i$ sono la struttura della logica $\hat{L}$, del linguaggio delle formule atomiche L_a, del linguaggio delle formule complesse L_c, della funzione di interpretazione I e della relazione di implicazione $\models$, rispettivamente.

Terminologia 9.1 (Dati, conoscenza, logica mista) Parliamo di logiche, teorie e modelli di dati, conoscenza e logiche miste con il significato ovvio.

Osservazione 9.12 (Logica, vincoli di progettazione e scelte) Lo stencil della logica L^i è tutto ciò di cui abbiamo bisogno per implementare una logica e usarla per eseguire il ragionamento. Tuttavia, ci sono vincoli da tenere presente, in particolare quelli relativi alla funzione di interpretazione e alla relazione di implicazione. Ci sono anche scelte importanti da effettuare, in particolare su quale modello del mondo e logica dovrebbero essere selezionati, come implementarli e come integrarli.

Osservazione 9.13 (Definizione di una logica) La definizione di una logica è articolata nei seguenti passi:

1. Definire A_a. Questo fornisce il linguaggio per fare affermazioni sui fatti nel mondo. In seguito, A_a sarà definito in termini di un $\hat{KG}$.
2. Definire L_a^i e, in particolare, $\{FR\}_a$. Questo fornisce il linguaggio per produrre descrizioni complesse dei fatti nel mondo.
3. Convalidare la definizione di A_a e L_a^i. Ciò viene fatto tramite la definizione di I_i dove l'obiettivo è assicurarsi che la notazione utilizzata per A_a e L_a^i sia intuitiva e rappresentativa dei fatti del mondo e che tutti i vincoli su I^i siano soddisfatti.

4. Definire L_c^i. Questo fornisce il linguaggio per costruire descrizioni complesse dei fatti nel mondo, come rappresentato nel $\hat{KG}$. Questo di solito viene fatto utilizzando un linguaggio logico che formalizza una parte mirata del linguaggio naturale.

5. Definire $\models^i$, cioè le regole per il ragionamento, basate sulla definizione di L_c^i. In particolare, l'idea chiave è che la definizione dell'implicazione è composita, seguendo la composizionalità delle regole di generazione per L_c^i.

Dato un sistema logico, è possibile usarlo per ragionare sul mondo. Questo è l'argomento della prossima sezione.

9.6 Usare la logica

La logica consente di effettuare ragionamenti complessi sui modelli del mondo. Vediamo come questo può essere implementato nella pratica.

Intuizione 9.1 (Risolvere problemi usando la logica) Questo è molto simile a quanto accade con il modello del mondo. Visto dall'esterno, l'utente non noterà differenze, ma solo un aumento nella capacità di ragionamento. Vedi Intuition 6.6.

Terminologia 9.2 (Linguaggio del modello del mondo, linguaggio del ragionamento) Nelle seguenti assumeamo che $L_Q = L_A = L_W$, dove L_W è il linguaggio di rappresentazione del modello del mondo, e che $L_T = L_R$, dove L_R è il linguaggio di rappresentazione del ragionamento. Si noti che, rispetto alla Terminology 6.3, assumiamo l'esistenza di LR e che $L_W \neq LR$. Questo è motivato dal fatto che il compito di rappresentare il mondo nel modello del mondo e il compito di ragionare sul mondo sono di solito implementati da due sistemi indipendenti con un terzo sistema esplicitamente dedicato a colmare i due sistemi e a tradurre L_W in L_R. Vedi anche Intuition 9.3.

Intuizione 9.2 (Logiche, problemi di ragionamento) Ma quali domande e quali risposte? Tutte le logiche, indipendentemente dalle specifiche della loro definizione, come da Definizione 9.10, forniscono risposte a sei domande fondamentali che enunciamo di seguito come problemi di ragionamento. Supponiamo di avere lo stencil $\hat{L}^i =< L^i, D^i, I^i, \models^i >$ e che abbiamo un insieme M di modelli con $M \subseteq D$ e un insieme di teorie $T \subseteq L_A$. Allora abbiamo quanto segue:

Problema di ragionamento 9.1 (Verifica del modello) Dati T e M, verificare se $M \models T$.

Problema di ragionamento 9.2 (Soddisfacibilità) Dato T, verifica se esiste un M tale che $M \models T$.

Problema di ragionamento 9.3 (Validità) Dato T, verifica se esiste un M tale che $M \models T$.

Problema di ragionamento 9.4 (Insoddisfacibilità) Dato T, verifica se non esiste alcun M tale che $M \models T$.

Problema di ragionamento 9.5 (Conseguenza Logica) Dati T1, T2 e un insieme di modelli di riferimento M, verifica se $T1 \models^M T2$.

Problema di ragionamento 9.6 (Equivalenza Logica) Dati T1, T2 e un insieme di modelli di riferimento M, verifica se $T1 \models^M T2$ e $T2 \models^M T1$.

Proposizione 9.1

Se una formula è valida, allora è anche soddisfacibile e non insoddisfacibile. In altre parole:

$$\text{Validità} \supset \text{Soddisfacibilità} \supset \text{not Insoddisfacibilità}$$

Proposizione 9.2

Se una formula è insoddisfacibile, allora non è nemmeno soddisfacibile e nemmeno valida. In altre parole:

$$\text{Insoddisfacibilità} \supset \text{not Soddisfacibilità} \supset \text{not not Validità}$$

Osservazione 9.14 (Logiche vs. Modelli del Mondo) I modelli del mondo (vedi Intuizione 6.7) affrontano i primi quattro problemi di ragionamento della logica, esposti nello stesso modo. La differenza chiave e fondamentale è che nei modelli del mondo, l'implicazione si riduce all'inclusione di insiemi, ovvero alla verifica se l'interpretazione di una formula appartiene a un modello (vedi Definizione 6.11). Sebbene molto più semplice del ragionamento logico, fornisce la base su cui è implementato il ragionamento logico.

Osservazione 9.15 (Problemi di Ragionamento, Dipendenza dalla Logica) Come le sezioni seguenti chiariranno, diverse logiche presentano istanze specifiche e diverse dei problemi di ragionamento definiti nell'Intuizione 9.2. Questo è in realtà il motivo principale per cui esistono modelli del mondo multipli. Nonostante il fatto che risolvano tutti gli stessi sei problemi fondamentali, lo fanno in contesti specifici e spazi problemi per i quali sono tarati. Mostrare e discutere queste specificità sarà l'argomento delle sezioni seguenti.

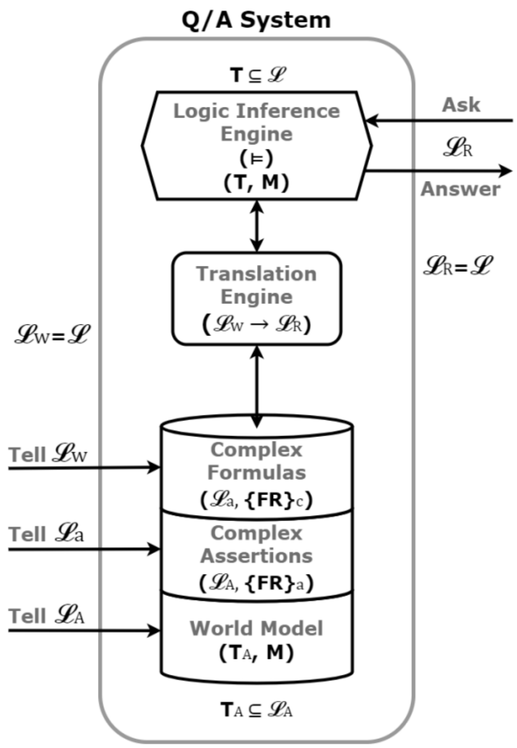


Figure 9.1: Risolvere problemi utilizzando la logica

Intuizione 9.3 (Un'architettura per risolvere problemi usando la logica) L'architettura che supporta l'uso della logica, come specificato in Intuizione 9.1, è rappresentata nella Figura 9.1. Possiamo identificare tre componenti principali, come segue:

- Un modello del mondo che codifica i dati e le conoscenze disponibili sul mondo e consente una ragionevole riflessione su di essi (vedi i problemi di ragionamento in Intuizione 6.7);
- Un motore di inferenza logica che implementa uno o più dei problemi di ragionamento definiti in Intuizione 6.7;
- Un motore di traduzione del linguaggio di rappresentazione che implementa la traduzione bidirezionale tra LW e LR.

Alcune osservazioni. Il modello del mondo e i motori di ragionamento vengono selezionati all'inizio, prima che il sistema venga messo in funzione. La scelta del modello del mondo e del modello logico dipende dalle specifiche del problema da risolvere. L'Osservazione 9.18 di seguito descrive le questioni e i compromessi che devono essere presi in considerazione quando si effettua la scelta del modello logico.

Osservazione 9.16 (Procedura decisionale, motore di inferenza logica) Una procedura decisionale è un algoritmo che, per una determinata logica, risolve uno dei sei problemi fondamentali specificati in Intuizione 6.7. Un motore di inferenza è di solito una procedura decisionale con due caratteristiche aggiunte:

- Può risolvere più di un problema di ragionamento, sfruttando il fatto che, in alcune logiche (ma non tutte), la soluzione di un problema di ragionamento può essere ridotta alla soluzione di un altro problema di ragionamento.
- Implementa diverse euristiche il cui obiettivo principale è accelerare il tempo di calcolo.

Osservazione 9.17 (Motore di traduzione del linguaggio di rappresentazione) Come sarà discusso in dettaglio, il lavoro sui motori di inferenza è più avanzato per alcune logiche in cui esistono implementazioni open source molto veloci e pronte all'uso. In questi casi, il motore di traduzione viene utilizzato per riscrivere un problema di ragionamento enunciato in un modello del mondo o in una logica di origine in un problema di ragionamento enunciato nella logica di destinazione.

Intuizione 9.4 (Il processo di utilizzo della logica) Le logiche sono utilizzate seguendo il modello Tell, Ask, Reason, Answer definito in Intuizione 9.1. Tuttavia, prima di ciò, il sistema rappresentato in Figura 9.1 deve essere specificato (e implementato). Il modello di specifica affinché ciò avvenga è il seguente:

1. Selezione del modello del mondo W e, in particolare, del linguaggio di rappresentazione LW e della funzione di interpretazione corrispondente. Ciò comporta a sua volta i seguenti tre passaggi:
   - Selezione del linguaggio assertivo L_A (e della funzione di interpretazione) utilizzato per specificare il dominio di interpretazione;
   - Selezione del linguaggio delle formule atomiche L_a (e della funzione di interpretazione) utilizzato per specificare formule atomiche e il corrispondente insieme di regole di formazione R_a;
   - Selezione del linguaggio di rappresentazione L utilizzato per specificare le formule e il corrispondente insieme di regole di formazione R_a.
2. Selezione della relazione di implicazione $\models$, che comporta i seguenti due passaggi:

- Selezione del linguaggio di rappresentazione del ragionamento L_R;
- Selezione della funzione di interpretazione per L_R;
- Selezione della relazione di conseguenza $\models L_R$ utilizzata per implementare il ragionamento e i problemi di ragionamento corrispondenti;
- Selezione del motore di inferenza utilizzato per risolvere i problemi di ragionamento target.

3. Selezione della procedura di traduzione utilizzata per tradurre le affermazioni $a \in L_a$ che descrivono i fatti memorizzati nel modello del mondo alle formule $w \in L$ del linguaggio del ragionatore.

4. Selezione del motore di inferenza.

Osservazione 9.18 (Logica, trade-off nella selezione) Qualsiasi logica può essere caratterizzata da due parametri principali:

- Espressività, ovvero il livello di dettaglio con cui il problema è espresso, a seconda della sintassi del linguaggio della logica;

- Efficienza computazionale, cioè quanto costa, in termini di spazio e tempo, ragionare e rispondere a interrogazioni in quel linguaggio.

Maggiore espressività consente una modellazione più raffinata e precisa del problema, ma genera anche formule più lunghe e complesse. Esiste un trade-off cruciale, in quanto maggiore è l'espressività di una logica, meno è computazionalmente gestibile.

Il modellatore deve trovare il giusto equilibrio tra espressività e complessità computazionale. In questo contesto, la scelta del linguaggio di rappresentazione $L =< L_a, L_c >$ è cruciale. La complessità computazionale sia di L_a che di L_c varia infatti da polinomiale a esponenziale e oltre. Sorge anche una questione di (in)decidibilità, cioè la possibilità per il ragionatore, su alcune interrogazioni, di entrare in un ciclo infinito, senza mai terminare e, di conseguenza, senza mai restituire una risposta.

Tuttavia, questa questione, sebbene molto rilevante nella logica matematica e nella teoria della computazione, non ha rilevanza pratica in questo contesto, principalmente perché nei domini di interpretazione dell'IA sono (quasi) sempre finiti, garantendo così la terminazione, anche nel caso di LOL, una delle logiche più espressive che consideriamo.

Chapter 10

$\mathcal{LOE}$ A Logic of Entities