TLN - Relazione Esercizio 2

Lorenzo Tabasso

Dipartimento di Informatica, Università degli Studi di Torino
lorenzo.tabasso@edu.unito.it

1 Introduzione

Questa relazione illustra la la risoluzione del secondo esercizio assegnato dal professor Mazzei, ovvero la realizzazione di un traduttore interlingua che traducesse dall'inglese all'italiano alcune frasi date in input. La particolarità di questo traduttore risiede nell'uso di una rappresentazione indipendente dal linguaggio (basata sulla semantica), la quale viene usata durante l'operazione di traduzione per generare la frase in output. Questo approccio alla traduzione offre molti vantaggi, come la possibilità di traduzione tra lingue molto diverse tra loro (ad esempio inglese ed arabo), a scapito però, di una crescente difficoltà riguardo alla definizione di una rappresentazione semantica su larga scala. Per questo motivo, nell'ambito di questo progetto mi sono concentrato solamente sulle tre frasi d'esempio riportate nella consegna del progetto. Ho infatti sviluppato prima una grammatica con una semantica associata per la traduzione delle tre frasi, e poi in seguito realizzato un Sentence Planner e un Realizer in grado di risolvere il task di traduzione interlingua.

2 Soluzione proposta

Il progetto sviluppato è suddiviso in tre parti eseguibili in cascata, le quali elaborano le tre frasi traducendole dall'inglese all'italiano. Queste tre parti corrispondono ai seguenti Task:

1. **Parsificazione della frase**: ogni frase viene parsificata utilizzando la grammatica sviluppata.
2. **Creazione del Sentence Plan**: utilizzando la rappresentazione semantica della frase ottenuta al passo precedente, il Sentence Planner genera un nuovo albero a dipendenze secondo un template basato sulla frase di partenza. Questo albero contiene già la traduzione 1:1 dei termini nella lingua di arrivo, ottenuta grazie all'ausilio di un dizionario.
3. **Realizzazione della frase**: il Realizer sviluppato, basato su *SimpleNLG-It*, converte il Sentence Plan ottenuto al passo precedente in una frase nel linguaggio di destinazione, e la riporta all'utente sia sul terminale del programma che in un file di testo.

2.1 Risorse utilizzate

Il progetto è stato sviluppato in **Python 3.7.7** (le prime due parti) e in **Java 1.8** (la parte del Realizer). Nello specifico, sono state impiegate le seguenti librerie:

1. **Python 3.7.7**
   (a) **NLTK**: usata nelle prime due parti. Essa fornisce svariati tool per l'interpretazione e la manipolazione del linguaggio naturale. In questo contesto, è stata utilizzata per parsificare il testo in input tramite la grammatica con annotatazioni semanticamente associata, e per attraversare l'albero di parsing creato da nltk (la libreria utilizza il lambda calcolo per ottenere una rappresentazione in logica del primo ordine della semantica della frase).
   (b) **AnyTree**: libreria usata per la creazione del Sentence Plan,
   (c) **JSON**: libreria usata per l'esportazione del Sentence Plan. Ho scelto di esportarlo in JSON per facilitare la manipolazione e la lettura dei dati.
2. **Java 1.8**
   (a) **SimpleNLG-It**: usata nella terza parte (Realizer). Essa fornisce i metodi necessari alla generazione della frase, permettendo di poter settare le features (tempo verbale, genere e numero) della singola frase da esportare.
   (b) **Jackson** e **SimpleJSON**: librerie usate per la manipolazione del Sentence Plan in input al Realizer.

3 Il metodo proposto

Nelle prossime sottosezioni sono listati tutti i dettagli del metodo implementato.

3.1 Grammatica

La gramatica del traduttore è stata costruita usando come simboli non terminali una parte dei Tag del Penn Treebank, i quali sono riportati nella tabella sottostante. Ad essi, sono stati aggiunti altri tre simboli non terminali, che rappresentano i costituenti della frase: *NP*, *VP* e *PP*.

Sono state inoltre inserite le seguenti *features* al livello semantico, con lo scopo di migliorare il processo di traduzione.

- LOC per le espressioni locative,
- POSS per le espressioni possessive,
- PERS per le espressioni personali,
- NUM per indicare il numero (singolare o plurale),
- GEN per indicare il genere (maschile o femminile)

Tag	Description
NN	Noun, singular or mass
NNS	Noun, plural
DT	Determiner
EX	Existential *there*
IN	Preposition or subordinating conjunction
JJ	Adjective
PRP	Personal pronoun
PRP$	Possessive pronoun
RB	Adverb
VBG	Verb, gerund or present participle
VBP	Verb, non-3rd person singular present
VBZ	Verb, 3rd person singular present

3.2 Semantica

Nella semantica descritta all'interno della grammatica, ogni variabile è vincolata da un quantificatore esistenziale. Ho deciso di utilizzare una rappresentazione Neo-Davisoniana per gestire sia la presensa di aggettivi e avverbi che la presenza di verbi intransitivi. Nel primo caso, si otterrà una semantica del tipo $\lambda x.adj(x)$, mentre nel secondo caso si avrà *verb(e), agent(e,x), adverb(e)*. Per la gestione dei verbi transitivi, ho optato per una rappresentazione più semplice, del tipo *verb(x,y)*, dove x indica il soggetto e y l'oggetto. E infine, per gestire i termini con il ruolo di soggetto (il pronome personale "you" o nomi propri e comuni), ho utilizzato il *Type Raising*, definendo una semantica della forma $\lambda P.P(term)$. Un'esempio di quest'ultima casistica è la gestione del pronome personale "you", che ha la seguente semantica associata: $\lambda P.P(you)$.

3.3 Sentence Plan

Applicando la grammatica e la semantica descritte in precedenza alle tre frasi proposte in input, si ottengono le tre seguenti rappresentazioni semantiche in logica del prim'ordine:

1. exists z1.(thing(z1) & image(you,z1))
2. exists x.(exists e.(presence(e) & agent(e,x)) & exists z2.(my(z2) & head(z2) & exists z4.(price(z4) & x(z4)) & on(x,z2)))
3. exists x.(your(x) & big(x) & opportunity(x) & exists e.(fly(e) & agent(e,x) & out(e) & exists y.(from(e,y) & here(y))))

Tali rappresentazioni corrispondono in ordine alle tre frasi: *"You are imagining things"*, *"There is a price on my head"*, e *"Your big opportunity is flying out of here"*.

Nella seconda e terza rappresentazione, è evidente l'uso della rappresentazione Neo-Davisoniana nell'elaborazione dei due terminali "there is", (la cui semantica prodotta dal sistema è *exists x.(exists e.(presence(e) & agent(e,x)) &*

...) e nella gestione della catena di aggettivi come "[...] big opportunity [...]" e di avverbi "[...]out of here[...]".

Ponendo l'attenzione sul termine *"is"*, si nota che esso ricopre due ruoli diversi all'interno delle ultime due frasi, motivo per il quale è stato necessario disambiguare i due ruoli usando due regole grammaticali differenti. In questo modo, è stato possibile sia gestire il legame dell'*is* all'esistenziale *"there"* nella seconda frase, che gestire il ruolo di ausiliare del verbo *"fly"* nella terza frase.

Nella creazione delle due regole appena descritte, ho deciso di assegnare al terminale lo stesso PoS-Tag: **VBZ**, per semplicità e coerenza verso il significato del PoS-Tag all'interno del Penn Treebank. Come svantaggio però, il numero di alberi di parsing prodotto dalla grammatica è aumentato, ottenendo in output anche delle formule non ben formate. Perciò, per porre una soluzione al problema, ho fatto uso di tre espressioni regolari per catturare l'espressione semantica corretta.

Un'alternativa alla scelta progettuale effettuata potrebbe essere quella di utilizzare una testa differente per entrambe le regole, che permetterebbe di eliminare l'ambiguità nella scelta.

3.4 Lessicalizzazione

Il passo finale della costruzione del Sentence Plan consiste nella traduzione del singolo termine dall'inglese all'italiano. Per semplicità, ho scelto di utilizzare un dizionario che associa in modo univoco (senza tenere conto di ambiguità) una parola in inglese con il corrispettivo termine in italiano. L'applicazione della traduzione del termine tramite il dizionario citato poc'anzi avviene appena prima del popolamento dell'albero a dipendenze. Appena dopo la creazione della foglia viene applicata la traduzione 1:1, e il valore risultante viene inserito all'interno della foglia appena istanziata. La struttura intermedia (creata usando la libreria *AnyTree*) che rappresenta un'astrazione ad alto livello dell'albero a dipendenze, contiene nelle foglie i termini della frase tradotti dall'inglese all'italiano, e nei nodi interni i costituenti della frase. I figli di un nodo interno sono gli attributi che *SimpleNLG* richiede per quel tipo di nodo. Terminata la costruzione dell'albero esso viene codificato in formato JSON e passato al Realizer.

3.5 Realizzazione frase

Per convertire la struttura del Sentence Plan in una frase in italiano, ho siviluppato un Realizer in Java usando la libreria *Simple-NLG-It*. Quest'ultimo componente del progetto, prende in input il JSON contenente il Sentence Plan costruito al passo precedente, e per ogni suo elemento procede come segue. Per ogni nodo all'interno del Sentence Plan, crea un oggetto PhraseElement che in base alla tipologia del costituente può essere un NPPhraseSpec, un VPPhraseSpec oppure un PPPhraseSpec. Ad ognuno di questi nodi principali vengono quindi aggiunti i nodi figli o le foglie a seconda del ruolo che svolgono nel Sentence Plan. Ad esempio un oggetto di tipo NPPhraseSpec avrà due figli: *specifier* e

noum, i quali possono essere impostati attraverso i metodi *.setSpecifier()* e *.setNoum()*. Come passo finale, tramite il metodo *.setFeature()*, vengono aggiunte le proprietà al nodo (tempo verbale, genere e numero) che sono state catturate nella fase di parsing.

4 Risultati

In questa sezione vengono analizzati i risultati del progetto.

4.1 Alberi sintattici e semantica

Come già accennato in precedenza, le tre rappresentazioni semantiche ottenute dal PoS-Tagger di NLTK, usando la grammatica da me definita sono le seguenti. Ad ogni rappresentazione semantica, allego il corrispettivo albero sintattico, all'interno del quale sono presenti sia la semantica che il relativo λ-calculus (la sintassi riportata all'interno dell'albero è quella utilizzata da NLTK).

1. *You are imagining things*: **exists z1.(thing(z1) & image(you,z1))**
   In questa rappresetazione vi sono due predicati principali, il predicato unario *thing(z1)* e quello binario *image(you,z1))*, il quale mette in relazione l'oggetto, rappresentato dalla variabile vincolata *z1* e il soggetto *you*. L'albero sintattico di questa frase è visionabile nella figura 1.

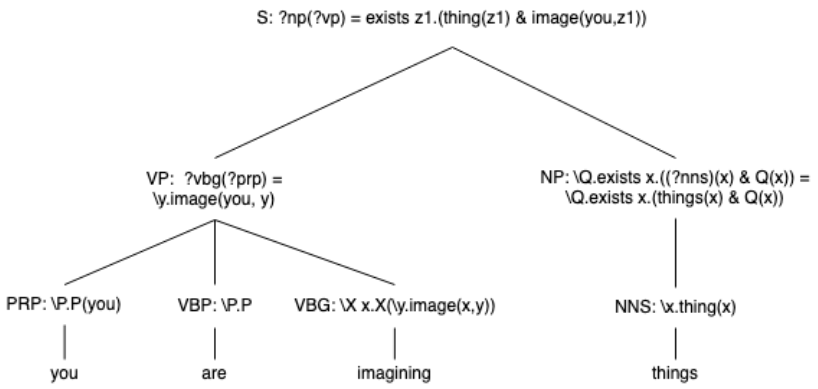

Figura 1. L'albero sintattico della frase "You are imagining things"

2. *There is a price on my head*: **exists x.(exists e.(presence(e) & agent(e,x)) & exists z2.(my(z2) & head(z2) & exists z4.(price(z4) & x(z4)) & on(x,z2)))**

All'interno di questa formula logica, il predicato *presence(e)* (dove *e* è l'evento) fa riferimento ai termini "there is", il cui significato è legato al verbo essere, inteso nel senso di "esistere", "essere presente". Il soggetto invece, indicato dal predicato *agent(e,x)*, è rappresentato dalla variabile x ed è associato ai termini *price* e *on* tramite i predicati *x(z4)* e *on(x,z2)*. Inoltre, il predicato *on(x,z2)* lo mette in relazione anche gli altri predicati *my(z2)* e *head(z2)*, che insieme rappresentano rispettivamente il modificatore (*"my"*) e l'oggetto (*"head"*). L'albero sintattico di questa frase è visionabile nella figura 2.

3. *Your big opportunity is flying out of here*: **exists x.(your(x) & big(x) & opportunity(x) & exists e.(fly(e) & agent(e,x) & out(e) & exists y.(from(e,y) & here(y))))**
   Analogamente all'esempio precedente, in quest'ultima formula *fly* è il verbo principale, il quale è modificato dall'avverbio *out* e messo in relazione (tramite il predicato *from(e,y)*) con il predicato *here*. Il soggetto in questo caso è identificato da *x*, ovvero *opportunity*, ed è modificato da *your* e *big*. L'albero sintattico di questa frase è visionabile nella figura 3.

4.2 Traduzione finale

Le traduzioni finali ad opera del Realizer scritto con Simple-NLG-It sono riportate di seguito:

1. You are imagining things → Tu stai immaginando cose.
2. There is a price on my head → Un prezzo esiste sulla mia testa.
3. Your big opportuniy is flying out of here → La tua opportunità grande sta volando fuori da qui.

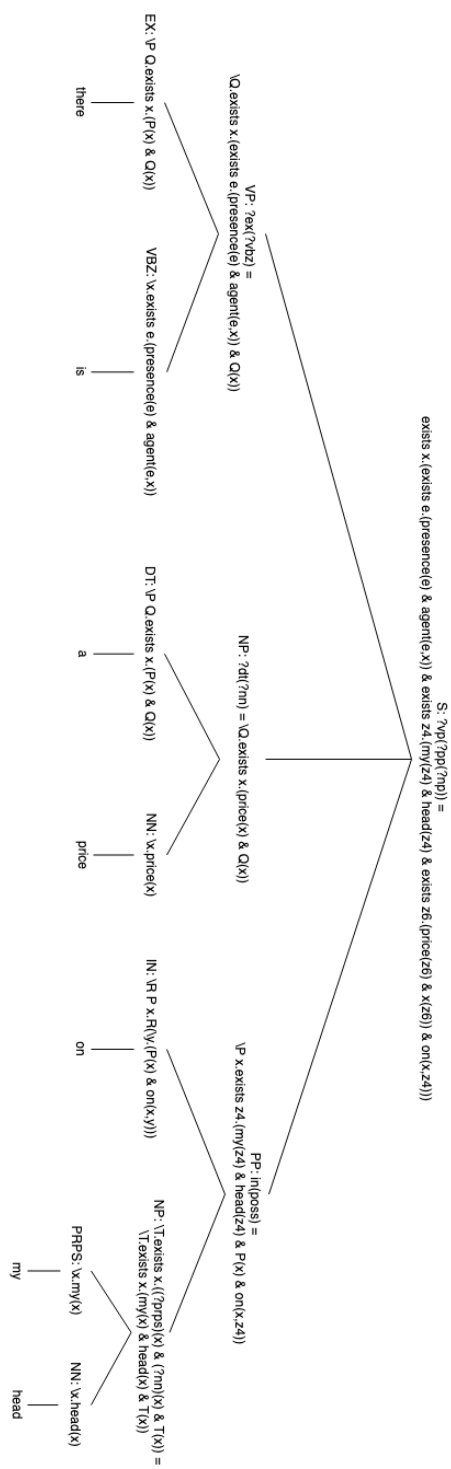

Figura 2. L'albero sintattico della frase "There is a price on my head"

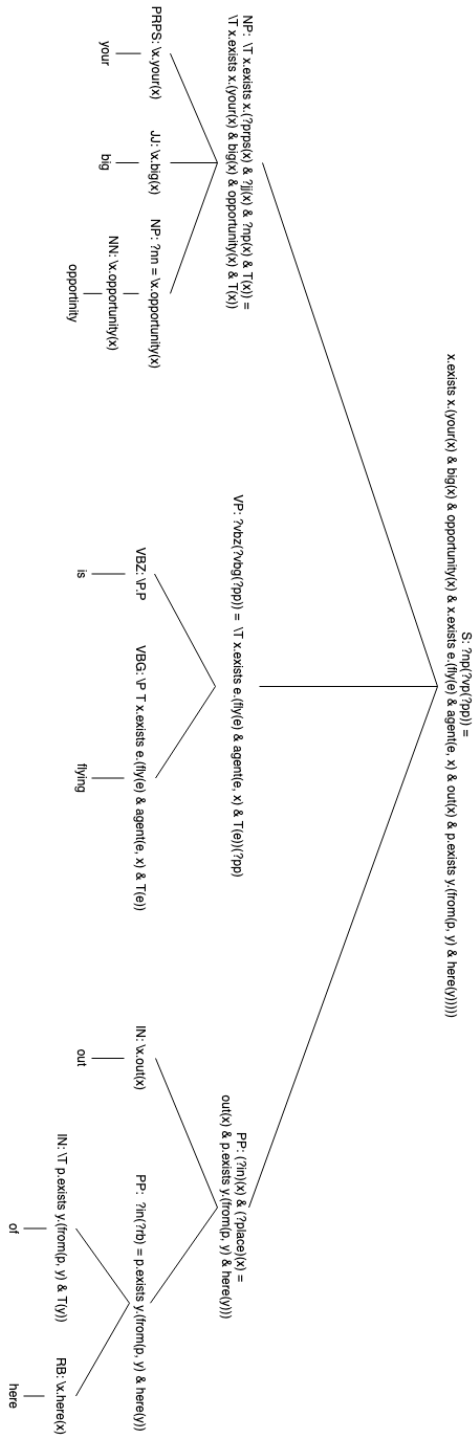

Figura 3. L'albero sintattico della frase "Your big opportunity is flying out of here"