Peer Reviewed Papers

Costruzione di Basi di Conoscenza AIML per chatter bot a partire da FAQ e Glossario

Giovanni De Gasperis

Dipartimento Ingegneria Elettrica e dell'Informazione, Università degli Studi dell'Aquila

giovanni.degasperis@univaq.it

Keywords: chatter bot human computer interaction

Abstract

I chatter bot sono programmi che emulano una conversazione umana e che possono dimostrare un comportamento molto simile a quello umano all'interno di un dominio di conoscenza ben definito. AIML, Artificial Intelligence Markup Language, è un linguaggio esteso dall'XML per descrivere basi di conoscenza ad uso dei chatter bot, in un contesto d'uso tramite algoritmi di ragionamento case-based e riconoscimento di pattern testuali. Viene qui descritta una metodologia di progetto e realizzazione di chatter-bot basata su un algoritmo di generazione automatica delle basi di conoscenza AIML a partire da un file di testo delle domande più frequenti nel dominio di conoscenza (FAQ) e un file di testo del glossario dei termini utilizzati. Un esempio di generazione completa di chatter bot è descritto e dimostrato.

1 Introduzione

I primi tentativi di creare degli agenti conversazionali risalgono agli anni '60-'70 con ELIZA (1966) e PARRY (1972) (Guzeldere & Franchi, 1995). Il loro funzionamento si basa sul riconoscimento di parole o frasi chiave date in input e su una serie di risposte pre-preparate e pre-programmate corrispondenti in output che possono far sembrare intelligente l'andamento della conversazione.

Attualmente le interfacce utente multi-modali (Pirrone & Cannella, 2008) possono includere un interprete in linguaggio pseudo-naturale che emula una conversazione umana allo scopo di rendere familiare e confortevole per l'utente l'interazione con strumenti di information retrieval. La conversazione è supportata da software denominati chatter bot (Many Authors, 2009); alcuni di essi hanno una base di conoscenza descritta in AIML e sono costituiti da un interprete AIML (AIML, 2005). Per questo tipo di agenti conversazionali la base di conoscenza è costituita da coppie (pattern, template), che possono essere connesse semanticamente e/o ricorsivamente dal costrutto SRAI (Symbolic Reduction in AIML).

A.L.I.C.E. è l'implementazione più conosciuta di un chatter bot generalista di lingua inglese, nelle sue diverse edizioni. Ancora oggi A.L.I.C.E. tuttavia si basa su delle tecniche di pattern-matching simili a quelle utilizzate da ELIZA dal 1966.

Tra i vari Alicebot, recentemente Wallace ha annunciato il servizio web SpellBinder (Wallace, 2009) per mezzo del quale una base di conoscenza AIML può essere generata a partire dalle trascrizioni dei dialoghi dei personaggi di opere cinematografiche o serie televisive, assumendo ed emulando la personalità e il modo di parlare dei personaggi stessi. Come esempio viene offerta la possibilità di interagire con il famoso personaggio James T Kirk, ottenuto dalle trascrizioni di tutti i dialoghi della prima serie TV Star Trek degli anni '60.

Il modello sottostante può essere riferito alle reti semantiche della teoria del case based reasoning (Smid, 20002), risolte tramite algoritmi di pattern-matching testuale (Wallace, 2007). Una base di conoscenza può essere rappresentata tramite un grafo del tipo illustrato in Fig.1.

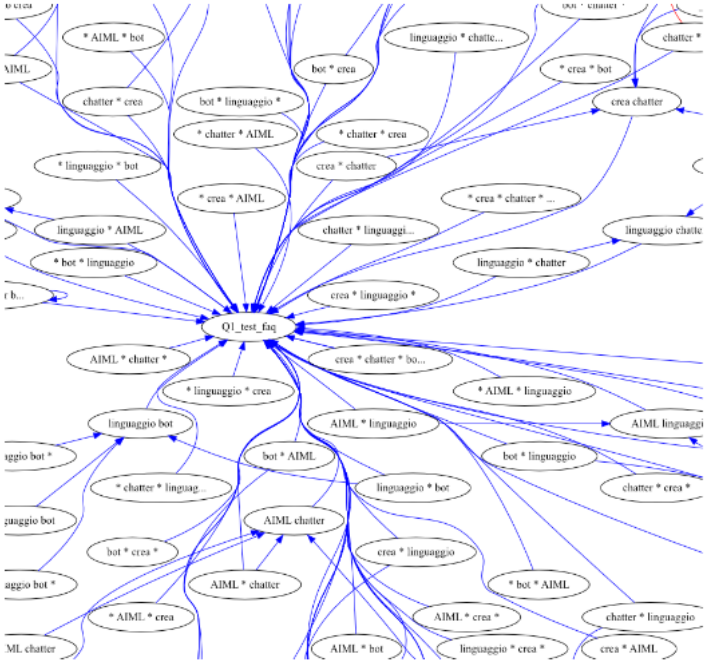

Fig. 1 - Particolare del grafo centrato sulla domanda Q1. Il grafo visualizza la rete di ricorsione delle categorie della base di conoscenza AIML ottenuta dalla generazione automatica a partire dal file FAQ e glossario discussi nel testo.

2 Metodologia di Progetto della Base di Conoscenza

L'insieme delle domande frequenti restringe semanticamente il problema nel dominio di conoscenza desiderato da fornire al chat bot; sono facili da scrivere e da ottenere perché spesso su molti argomenti sono già disponibili dal web. Diversamente, il glossario è uno strumento meno diffuso, ma può essere derivato dalle numerose risorse online disponibili, ammesso di poter relazionare i termini utilizzati con il dominio di conoscenza di interesse. Dato che la conoscenza strettamente necessaria che delimita il dominio è esplicitamente ed implicitamente inclusa nei file di FAQ e glossario, è' importante poter considerare questi ultimi come punti di partenza del processo di generazione di un software esperto capace di rispondere alle stesse domande presenti nelle FAQ e di fornire le definizioni del glossario, come illustrato in Fig.2.

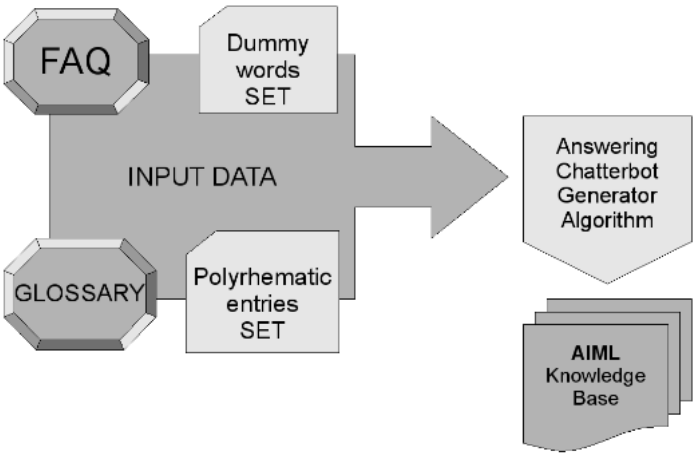

Fig 2 - Flusso del processo di generazione del chatter bot

Bisogna creare un file di testo delle FAQ, indicato con T_{FAQ}, in un formato tale che ciascuna domanda sia associata alla relativa risposta, e un altro file contenente il glossario, T_{GLO}, in un formato nel quale un termine è associato con la rispettiva definizione. Al fine di ottenere un chatter bot AIML accurato al termine del processo di generazione, devono essere compiuti i seguenti passi principali:

1. definizione ed implementazione dell'algoritmo di generazione del chatter bot.

In funzione del dominio di conoscenza:

2. definizione di un insieme di parole irrilevanti
3. definizione di un insieme di elementi polirematici specifici del dominio

dove il termine irrilevante indica che la parola è utilizzata solo ad un scopo sintattico e strutturale della frase, ma con nessuna relazione semantica con il dominio di conoscenza; l'elemento polirematico si riferisce a combinazioni linguistiche e/o sequenze di parole che possono essere associate ad un unica entità semantica. Questi due insiemi sono utilizzati per filtrare o selezionare le parole del corpus delle FAQ e del glossario, definendone il gruppo strettamente necessario alla generazione dell'insieme dei pattern che vanno a costituire la base di conoscenza AIML finale. E' possibile anche adottare delle tecniche di

linguistica computazionale per ridurre l'insieme di queste parole; queste metodologie sono utili per individuare le parole irrilevanti, come ad esempio quelle che sono più frequenti nel corpus costituito da tutte le parole del dominio di conoscenza, come viene proposto in (Shawar & Atwell, 2008), al fine di non inserirle nel data base AIML.

2.1 Definizione dell'INPUT

I dati in ingresso sono organizzati nel seguente modo:

FILE FAQ F_{in}:

```
Q <spazio> <testo della domanda>
A <spazio> <testo della risposta>
```

FILE Glossario G_{in}:

```
G <spazio> <testo della voce (polirematica>)
D <spazio> <testo della definizione >
```

Insieme parole irrilevanti D_{in}:
Un file testo con una parola per ogni linea

Insieme di elementi polirematici Pin:
Un file testo con tutte le parole componenti l'elemento per ogni linea

3 Algoritmo di generatore di chatter bot

Dato l'insieme dei dati in ingresso definito da $\{F_{in},G_{in},D_{in},P_{in}\}$, l'uscita dell'algoritmo è costituita dal grafo Z_{out} delle associazioni pattern template e associazioni ricorsive dal quale è possibile generare la base di conoscenza AIML (versione 1.0.1 (AIML, 200%)) finale; essa può essere successivamente utilizzata in unione con un interprete AIML 1.0.1 per l'implementazione del chatter bot.

3.1 Generazione del FAQ-AIML

L'algoritmo generatore è stato messo a punto tramite programmazione in linguaggio Python per un totale di circa 500 linee di codice. I passi principali possono essere riassunti come segue:

Algoritmo 1: Fasi principali della generazione dell'AIML

F1. estrazione delle liste di categorie rilevanti dalle domande in FAQ F_{in}
F2. calcolare tutte le possibili ramif icazioni
F3. estrarre le risposte associate alle domande
F4. generare Z_{out} e l'AIML

3.1.1 Dettaglio di F1

Una singola categoria, così come definita nell'AIML, corrisponde ad una coppia pattern-template (P-T). Il pattern deve coincidere con una o più parole prese da quelle contenute nella domanda in Fin, in maniera tale che possano essere utilizzate nella fase di analisi della domanda fatta dall'utente per determinare quale risposta occorre fornirgli. L'algoritmo è descritto in Algoritmo 2.

Algoritmo 2: Generazione delle categorie AIML

Sia D_w l'insieme delle parole irrilevanti
Sia P_w l'insieme degli elementi polirematici
FOR tutte le domande q in FAQ F_{in} DO
    sia L la lista costruita dalla sequenza delle parole rilevanti w_i
        (i.e. Filtrare tutte le w_i in D_w and usare le w_i in P_w)
    inizializzare una lista vuota di categorie C
    FOR tutte le parole w_i in L DO
        accodare w_i in C combinata con quelle già presenti prese 2 a 2
    END FOR
    costruire una lista di categorie M con le parole rilevanti trovate in q
    accodare C ed M all'insieme delle liste di categoria S_c
END FOR

3.1.2 Dettaglio di F2

Il metodo F2 calcola tutte le possibili ramificazioni uscenti da una categoria che possono portare a risposte diverse, come illustrato dall'Algoritmo 3.

Algoritmo 3: Estrazione delle ramificazioni uscenti dalla categoria

Sia OUT il dizionario in uscita che mappa una categoria ad una lista di interi
FOR tutte le liste di categorie C_1 in S_c DO
    sia A_i la risposta la cui domanda Q_i ha generato C_1
    FOR tutte le categorie c_i in C_1 DO
        accoda l'intero i alla lista OUT[c_i]
    END FOR
END FOR
ritorna OUT

Nell'implementazione risulta cruciale la struttura dati "dizionario" come ci viene offerta dal linguaggio Python, dove `OUT[<categoria>]` è fondamentale nella determinazione delle ramificazioni uscenti da quella stessa categoria.

3.1.3 Dettaglio di F4

In questa sezione viene generato il grafo Zout per la successiva generazione dell'AIML, cercando di ottenere tutte le parole rilevanti dall'enunciato utilizzato dall'utente e confrontandole con quelle rilevanti contenute nelle domande delle FAQ Fin. In questo modo si possono generare tutte le ricorsioni SRAI tra categorie, seguendo lo standard AIML 1.0.1 (AIML, 2005).

3.2 Generazione del GLOSSARIO-AIML

Al fine di generare il glossario AIML, per ogni voce filtrata dall'insieme delle parole irrilevanti e selezionata dall'insieme degli elementi polirematici, uno SRAI viene associato con la relativa definizione del glossario.

Algoritmo 4: Generazione del grafo Zout e dell'AIML

```
FOR tutte le domande Qi DO
        data la lista di categorie C1 generata da Qi
        sia Ta il template SRAI che contiene il testo della risposta
        FOR tutte le categorie Ci in C1 DO
                IF Ci è una combinazione di due parole THEN
                        genera tutti i possibili archi (SRAI) verso Ta
                ELSE IF Ci ha una sola ramif icazione THEN
                        genera un arco (SRAI) verso Ta
                        IF Ci è costituita da singola parola THEN
                                genera un arco (SRAI)
                                        alla def inizione del glossario
                        END IF
                END IF
        END FOR
END FOR
```

4 Risultato AIML

Il codice AIML del chatter bot viene generato principalmente a partire dai file FAQ e glossario. Viene poi generato ulteriore codice AIML che operando una riduzione semantica degli elementi presenti negli enunciati degli utenti formulati in testo libero in modo tale da aiutare a selezionare le parole rilevanti, in modo tale da ottimizzare la ricerca della risposta giusta alla domanda posta.

5 Caso di studio: mini chatter bot in italiano

Il caso di studio proposto è volutamente minimalista ai fini della trattabilità e della rappresentabilità grafica. E' composto di sole due domande Q1 e Q2. Un semplice chatter bot può quindi essere generato utilizzando il seguente file di input:

5.1 File di FAQ

Q Come si crea un chatter-bot con il linguaggio AIML?

A Prima devi imparare il linguaggio e i suoi principali costrutti. Ovviamente devi prima aver imparato l'XML perche' ne e' un suo derivato. Tra i costrutti sintattici pi potenti ti suggerisco lo SRAI, che ti permette di connettere coppie pattern-template tra loro, come ad esempio nel caso dei sinonimi. Poi devi costruire bene la base di conoscenza in AIML e darla in pasto a pandorabots.com.

Q Come si usa un chatter-bot?

A Per usare il chatter-bot devi inviare i file AIML presso un server ove sia residente un interprete AIML. Quindi devi editare una pagina HTML che contenga i campi codificati per la domanda e la risposta verso l'utente finale.

5.2 File di Glossario

G chatter bot

D un chatter-bot e' un software che simula una conversazione in linguaggio naturale

G AIML

D AIML, Artificial Intelligence Markup Language, e' il linguaggio per la definizione di una tipologia di chatter bot introdotto nel 2000 da Richard Wallace

G Alice

D E' il miglior chatter-bot conosciuto basato su AIML

G Richard Wallace

D Dr. Richard Wallace e' l'inventore del linguaggio AIML e del chatter bot Alice

5.3 Base di conoscenza generata

Date le 5 linee di testo, 116 parole e 700 caratteri del file di FAQ si ottiene un file AIML comprendente 195 categorie. Il file di glossario raggruppa ulteriori 19 categorie, date dalle 4 originali voci di glossario e una combinazione di parole rilevanti che possono essere presenti nella domanda. Il codice AIML non può essere illustrato qui per mancanza di spazio, ma è possibile richiederlo all'autore tramite email. Il chatter bot generato completo di tutta la base di conoscenza è disponibile online (De Gasperis, 2009) per sessioni libere di pseudo-conversazione.

Conclusioni

In questo lavoro è stata dimostrata una metodologia per la generazione automatica di un chatter bot in tecnologia AIML. Le possibili applicazioni nel settore dell'eLearning potrebbero facilitare l'interazione con l'utente o la navigazione all'interno del materiale didattico sotto forma di assistente digitale personale antropomorfizzato tramite avatar parlante. Ad esempio, in una tipica sessione di formazione a distanza, i cui contenuti del modulo didattico possano essere riassunti tramite FAQ e un glossario, lo studente che fruisce del materiale didattico online secondo canoni convenzionali, potrebbe anche consultare l'assistente digitale, realizzato con le tecniche qui esposte, ponendogli direttamente domande in testo libero relative ai quei contenuti per i quali ha bisogno di chiarimenti tramite una risposta ad una domanda, ove possibile.

Altre possibili applicazioni potrebbero essere sviluppate nel campo della robotica personale e in generale dei sistemi utilizzanti linguaggio pseudo naturale che vogliono interagire in un contesto multi-modale con l'utente (Pirrone & Cannella, 2008).

BIBLIOGRAFIA

Pirrone R., Cannella V., R.G. (2008), *Gaiml: A new language for verbal and graphical interaction in chatbots*. In: International Conference on Complex, Intelligent and Software Intensive Systems, 2008, 715–720.

Many Authors (2009), *Chatterbot entry*. http://en.wikipedia.org/wiki/Chatterbot (November 2009) Wikipedia.

AIML 1.0.1 reference (2005), http://www.alicebot.org/TR/2005/WD-aiml (2005) ALICE Artificial Intelligence Foundation.

Wallace R. (2009), *Pandorabots announces the availability of bespoke pandorabots spellbinder service*. Web page (October 2009) Pandorabots.com, http://pandorabots.com/pandora/pics/spellbinder/index.html.

Smid K. P.I. (2002), *Conversational virtual character for the web*. In: Proceedings of Computer Animation 2002, Geneva, Swtizerland (2002) 240.

Wallace R. (2007), *AIML pattern matching simplified*. http://www.alicebot.org/documentation/matching.html (2007) ALICE Artificial Intelligence Foundation.

Shawar B.A., Atwell E. R. A. (2008), *Faqchat as an information retrieval system*. http://www.comp.leeds.ac.uk/andyr/research/papers/ltc05-faqchat.pdf (2008) FAQchat.

De Gasperis G. (2009), *Italian generated example chatter-bot*. http://www.pandorabots.com/pandora/talk?botid=f0a3e607de36aa16 (2009) Hosted on pandorabots.com.

Güzeldere G., Franchi S. (1995), *Dialogues with colorful personalities of early AI*. http://www.stanford.edu/group/SHR/4-2/text/dialogues.html (24 July 1995).