Introduzione (slide 1)

Ciao a tutti. In quest’ultima parte dello Speech tratteremo il tema dei Chatbot e scopriremo come questi riescono a comunicare con gli utenti attraverso il linguaggio naturale, ovvero il linguaggio che le persone utilizzano tra di loro nella comunicazione di tutti i giorni.

Questa parte ha un’impronta prevalentemente tecnica dove vedremo un esempio concreto di applicazioni di intelligenza artificiale, o meglio una branca di essa che è chiamata “Natural Language Processing” generalmente abbreviato con l’acronimo NLP. In particolare vedremo i punti salienti relativi allo sviluppo di un progetto che stiamo realizzando in Onit; ovvero OnAssistant.

Che cos’è OnAssistant in due parole? Un assistente conversazionale che ci permette di prenotare prestazioni mediche attraverso l’utilizzo di un’applicazione di messaggistica istantanea che ormai tutti abbiamo sui nostri smartphone.

TEMPO:

Dominio del problema (slide 2)

Analizziamo brevemente il dominio del problema per capire l’utilità che può apportare il nostro agente conversazionale.

Attualmente per prenotare prestazioni mediche erogate dal Servizio Sanitario Nazionale (quindi dallo stato Italiano) si possono utilizzare queste modalità.

* Prenotazione allo sportello; quindi la persona si deve presentare negli uffici appositi con la ricetta cartacea rilasciata dal medico, negli orari specifici (perché aperti un numero ristretto di ore giornaliere), spesso rimanere in fila svariati minuti (e a volte anche oltre).
* Call Center: non sono noti per la loro efficienza e anch’essi disponibili solo in alcune parti di giornata.
* Portali OnLine: implicano una registrazione su questi portali, effettuare una navigazione sul web oppure il download di applicazioni specifiche se parliamo di utilizzo mobile.

Uno studio dell’Agenzia della Tutela della Salute di Milano, ha analizzato le prenotazioni del 2018 ed è emerso che circa l’80% di esse vengono ancora prenotate tramite la prenotazione classica allo sportello (quella più obsoleta).

TEMPO:

Vantaggi (slide 3)

Perché è stato scelto un Chatbot come soluzione? Con la realizzazione di OnAssistant si cerca di implementare un sistema che possa estrarre gli aspetti positivi delle modalità di prenotazione appena descritte lasciandone ovviamente le criticità.

* Efficienza:
  + Servizio sempre disponibile tutti i giorni in tutte le ore del giorno.
  + Durata di prenotazione media è nell’ordine di qualche minuto.
* Usabilità:
  + Uso facile ed immediato delle App. di messaggistica, unito alla possibilità di usare il linguaggio naturale, ne permette l’utilizzo ad ogni categoria di utente (il procedimento è equivalente allo scambio di messaggi con un amico)
* Praticità:
  + Interfacciamento con le applicazioni di messaggistica più “famose” (Telegram, Facebook messanger, Skype e molte altre). Purtroppo non è ancora possibile l’interfacciamento con Whapp, poiché non sono state rilasciate delle API fruibili dagli sviluppatori (ma arriveranno a breve).

Diversi studi dimostrano come l’interesse degli utenti verso l’utilizzo di messaggistica istantanea sia in costante aumento a discapito dell’uso delle applicazioni classiche.

TEMPO:

Scenario di utilizzo

Procederei con una *Demo* relativa alla prenotazione di una prestazione medica così sarà più facile comprendere i concetti che tratteremo più avanti.

TEMPO:

Architettura generale (slide 4)

Il prototipo è stato realizzato utilizzando una piattaforma fornita da Microsoft, chiama BotFramework SDK. In questa slide possiamo vedere l’infrastruttura generale e il flusso di conversazione che va da sinistra a destra e viceversa.

* Da questa parte abbiamo il lato Client (quindi il lato utente). Possiamo vedere alcune app di messaggistica che gli utenti possono utilizzare per mandare messaggi al nostro bot.
* Questo componente è il Dialog Manager, quindi la parte che si occupa dello stato e del flusso della conversazione. In pratica decide come e dove instradare la conversazione in base ai messaggi ricevuti dall’utente.
* Bot framework service è un servizio di Microsoft che permette l’interfacciamento tra il nostro bot e i client. Ogni messaggio ricevuto dal servizio, estrare le info più importanti tra le quali:
  + Il testo del messaggio.
  + L’Id dell’utente e del canale dal quale proviene il messaggio (per esempio per evitare che il bot riceva un messaggio da Luca su Messenger e risponda a Gianni su Telegram).
* Interfacciamento con LUIS.ai: ulteriore framework di Microsoft che consente l’applicazione effettiva di NLP. Consiste nel cervello del sistema ed è quel componente che permette l’utilizzo della comunicazione naturale. Se ci pensiamo un Chatbot nativamente non è nulla di eccezionale, poiché è pensato per rispondere a determinati input (comandi). Se si vuole realizzare un sistema più complesso è quindi necessario “collegare” un cervello che processi i messaggi ricevuti.
* Interfacciamento con OnHealth: un altro software di Onit a cui vengono demandati tutti i compiti di prenotazione visite / gestione pazienti ecce.

TEMPO:

Luis.ai (slide 5)

Il Natural Language Processing si basa su 3 concetti principali:

* Intent: rappresenta il significato attribuito ad una frase.
* Entities: parole chiave all’interno della frase che dobbiamo estrarre perché hanno un contenuto informativo
* Role: alcune entità assumono un ruolo all’interno della frase. Per esempio, se prenoto un volo da Milano a Londra, sono due entità di tipo “città” ma che assumono ruoli differente (città di partenza e città di destinazione).

Basandosi su questi concetti, gli algoritmi di NLP riescono a elaborare dati non strutturati (come una frase), in dati strutturati come per esempio il formato JSON (formato adatto all’interscambio di dati tra applicazioni clienti/server).

TEMPO:

Moduli LUIS.ai (slide 6)

Un aspetto molto importante riguarda la classificazione di intent ed entity e il loro inserimento in moduli appositi. Con modulo si intende un insieme di intent ed entity.

Perché ciò è molto importante? Facciamo l’esempio nel quale in nostro bot offra il servizio di prenotazione di appuntamenti e un altro servizio di visualizzazione dei nostri referti. Nel momento in cui l’utente entra nel flusso di una prenotazione, possiamo tranquillamente ignorare tutti gli intent che non sono collegati a quella funzionalità.

Quindi il modulo al quale ci stiamo riferendo viene continuamente switchato in background dal sistema in base a dove ci troviamo durante la conversazione. Questo viene fatto per evitare di avere un unico macro modulo con un numero elevato di intent/entity che può confondere il sistema.

TEMPO:

Struttura (slide 7)

La comunicazione è strutturata turni, che consistono in un “botta e risposta” e i relativi ack di conferma tra l’utente e il bot.

Implementazione Dialog Manager (slide 8)

La conversazione è composta da componenti base chiamati Dialog. Dialog semplici sono composti da un solo turno, mentre altri più complessi multiturno. Mano a mano che la conversazione va avanti questi dialoghi vengono aggiunti allo stack scambiandosi informazioni tra loro. Una volta che la conversazione è finita si ritorna al dialogo di root.

Fine

Quindi questi sono bene o male tutti gli aspetti inerenti la progettazione di un assistente conversazionale, trattati con un livello di astrazione abbastanza alto senza scendere troppo nel dettaglio.

Per concludere, volevo citarvi uno studio effettuato da Gartner (una delle più importati aziende di consulenza nel mondo IT), il quale afferma che a fine 2020 più del 50% delle aziende medio/grandi offriranno i propri servizi anche attraverso di Chatbot. Quindi chi sceglierà questo indirizzo di studi (che varia da un minimo di 3 anni a salire in base al vostro percorso), molto probabilmente avrà a che fare anche con queste tecnologie in futuro.

Grazie per l’attenzione.