

Il mercato online: le desuete FAQ e le nuove CHATBOT

Malagnino Marino José

— *Good Evening. Tell me your problems. Please terminate input with a period or a question mark.*

— *People get on my nerves sometimes.*

— *I am not sure I understand you fully.*

— *You should pay more attention.*

— *Suppose you should pay more attention.*

— *You're entitled to your own opinion.*

— *What makes you think I am entitled to my own opinion? [...]*

(conversazione fra Eliza e Parry, 1972)

Le chatbot si avviano in questi ultimi anni a sostituire le più note e statiche FAQ (*Frequently Asked Questions*) sui siti commerciali. Parliamo quindi di un preciso ambito di utilizzo, quello del web imbevuto di corpus testuali in costante aggiornamento, in cui ogni sito che integri una chatbot per interagire con la propria clientela è un dominio linguistico a sé. Benché personalità di spicco come Matt Mullenweg, il padre di WordPress, distingua una chatbot, dotata di motore semantico basata sull'intelligenza artificiale e su un sistema di autoapprendimento, da un bot costruito su una minuscola porzione di nozioni che non aumentano nel tempo in modo automatico ma devono essere sempre gestite da un essere umano, tendenzialmente non si riscontra online una grande cura a tenere distinti i termini, e anzi si sente spesso usare bot come semplice diminutivo di chatbot.

Da quando il 12 Aprile 2016 Facebook ha presentato all'evento F8 la nuova piattaforma di Messenger con la sua ultima novità, ovvero le chatbot per Messenger, il mondo degli sviluppatori e le aziende, piccole, medie o grandi che siano, hanno iniziato a darsi molto da fare per trovare nuove idee e giusti impieghi per questa rinnovata tecnologia. A giugno dello stesso anno si contavano già 11,000 bot per interagire con vari singoli user di Facebook. Parliamo di un mercato stimato in 900 milioni di persone. Salvo repliche, e profili *fake*, Facebook risulta quindi usato dall'11% della popolazione mondiale. Si è pertanto variamente sentito del 2016 come l'anno delle chatbot, e del 2017 come l'anno delle possibili conferme delle promesse.

Originariamente, sul web, le aziende usavano il proprio sito per parlare di sé. Spese le dovute pagine in presentazioni e offerte promosse, non era scontato continuare a ricevere dubbi, curiosità e solleciti da parte degli utenti. Ben presto fu chiaro che alcune domande ricorrevano sempre, e dopo un primo estenuante copia/incolla via email, molti siti hanno preferito fissare una pagina che collezionasse le domande più comuni, e le risposte più tipiche a queste.

Una buona pagina FAQ sui siti commerciali rappresenta quindi un punto di riferimento per i clienti dubbiosi o in cerca di informazioni, tuttavia il visitatore che potrebbe voler approfondire una questione non avrebbe modo di fare altre domande. Oppure l'organizzazione della FAQ potrebbe non essere immediata nei suoi vari rimandi, o essere priva di iper-link ad esempio, e non tutti gli utenti sono disposti a ripetere una ricerca. Se l'azienda prevede la possibilità di poter contattare un operatore telefonicamente o via email, l'utente potrebbe contattarlo, se ne ha voglia o se ha trovato subito il numero, anche se il rischio di abbandonare la pagina è ancora più alto e più facile dello sforzo di continuare a cercare una risposta. Per di più non tutte le aziende possono permettersi del personale per l'assistenza clienti ventiquattro ore al giorno e sette giorni su sette.

Utilizzando invece tutte le domande che sono state realmente poste, magari pervenute via e-mail, si possono creare degli schemi di domanda e risposta che realizzino l'automatismo che ha per fine l'acquisto di un certo prodotto o del suo ordine.

Gli attuali bot rispondono ad un unico dominio di domande con risposte in parte già scritte. I vari servizi online che offrono chatbot per aziende lo fanno forti del fatto che le domande ai vari servizi clienti (customer service) sono ripetitive e altamente prevedibili (o sarebbe meglio dire: totalmente vincolate al target dell'azienda). Con del training supervisionato da un operatore reale si possono poi migliorare le risposte automatiche rendendole sempre meno goffe e artificiose. Meno l'utente percepisce che sta dialogando con una macchina, più si sente appagato.

La storia delle chatbot comincia quando i computer non erano ancora, neanche lontanamente, imparentati con il mercato privato. Non esisteva Internet e il primo corpus digitale, l'«*Index Thomisticus*» di Padre Roberto Busa iniziato nel 1949 sarebbe stato portato a termine soltanto negli anni Ottanta. Correva gennaio 1966 e Joseph Weizenbaum rilasciò il

programma Eliza il cui compito era simulare una seduta psicologica rogersiana.

ELIZA deve il suo nome a Eliza Doolittle, la fioraia dall'eloquio incolto e dialettale protagonista della commedia *Pigmalione*, di Shaw, che grazie alla ripetizione delle forme corrette di pronuncia, apprese il modo di esprimersi delle classi più agiate.

```
180 PRINT "HI! I'M ELIZA. WHAT'S YOUR PROBLEM?"
190 REM
200 REM    ----USER INPUT SECTION----
210 REM
220 INPUT I$
```

Il programma è implementato mediante semplici tecniche di *pattern matching*, ovvero viene fatto il *parsing* delle frasi del paziente e vengono poi sostituite le parole chiave in frasi preconfezionate, ritenute verosimili se dette dalla bocca di uno psicologo. Nonostante la sua semplicità, ELIZA venne preso sul serio da molti dei suoi utenti, anche dopo la spiegazione del suo funzionamento da parte dello sviluppatore. Secondo lo stesso Weizenbaum infatti, il programma doveva semplicemente parodiare una seduta psicanalitica.

È utile allora spendere due parole circa la particolarità della psicologia rogersiana: la tecnica principale di questo tipo di seduta è la cosiddetta risposta riflettente, che è lo strumento con cui lo psicologo statunitense Carl Rogers (1902-1987) cercò di affinare la comprensione empatica col “cliente” (termine che lo psicologo preferiva a paziente). Anche nella sua applicazione più semplice, che consiste nel riformulare il contenuto appena espresso, questa modalità fa sentire al cliente che non è da solo e che qualcuno lo ascolta e che cerca di capirlo. Inoltre permette di mettere a fuoco meglio quanto sta dicendo; infatti è del tutto istintivo per il “cliente” cercare di capire se la riformulazione fatta dal terapeuta gli corrisponda oppure no; in questo modo ha la possibilità di esplorarsi. L'effetto Eliza, ovvero ritenere un computer assai più complesso di quanto non sia in realtà, non è assolutamente dovuto ad una reale complessità del codice programmato, ma è frutto soltanto della disperata ricerca umana (almeno nella contingenza di paziente/cliente) di voler sapere di essere ascoltati. Inoltre giova ricordare che la struttura della lingua inglese permette molto facilmente di trasformare una frase affermativa in una interrogativa

spostando la posizione del verbo ad inizio sintagma. Ecco quindi che l'eventuale sfogo "I'M AFRAID OF THE DARK", trovato il pronome personale che identifichi il cliente (*I'm*) e invertito questo (reso un "io" in un "tu") e spostatolo dalla sua posizione, può facilmente essere reso in una frase interrogativa "DID YOU COME TO ME BECAUSE YOU ARE AFRAID OF THE DARK?". Ovviamente tutto questo a patto che una porzione di testo fosse già stata preventivata. Eccone alcuni esempi:

```
1500 REM ----REPLIES----
1510 REM
1520 DATA "DON'T YOU BELIEVE THAT I CAN*"
1530 DATA "PERHAPS YOU WOULD LIKE TO BE ABLE TO*"
1540 DATA "YOU WANT ME TO BE ABLE TO*"
1550 DATA "PERHAPS YOU DON'T WANT TO*"
1560 DATA "DO YOU WANT TO BE ABLE TO*"
1570 DATA "WHAT MAKES YOU THINK I AM*"
    [...]
1790 DATA "DID YOU COME TO ME BECAUSE YOU ARE*"
```

Vediamo qui sopra riportate alcune delle frasi preconfezionate ritenute possibili in una situazione di sessione con uno psicologo. Le frasi vengono completate sostituendo all'asterisco il verbo usato dal cliente, o l'intero costruito sintagmatico. Questo è possibile finché la frase digitata dal cliente è costruita nel più semplice dei modi, se la frase contiene un errore o una forma non standard (o più semplicemente non preventivato dal programma) viene rilasciata una frase randomica (una frase scelta a caso fra una lista di frasi chiuse), totalmente cristallizzata, ma comunque ritenuta verosimile nel contesto di una seduta psicanalitica. Frasi ad esempio come

```
2390 DATA "HOW?"
2400 DATA "YOU SEEM QUITE POSITIVE."
2410 DATA "ARE YOU SURE?"
2420 DATA "I SEE."
2430 DATA "I UNDERSTAND."
```

Questo tipo di frasi vengono interpretate nella situazione di seduta rogersiana come un invito a continuare a parlare, a discorrere; quando il programma riesce invece a fare *matching* allora rilascia una delle frasi che

si completano con elementi del sintagma digitato dal cliente, come già mostrato nell'esempio precedente. Per inciso: la prima scrittura del programma fu in SLIP (*Symmetric List Processor*), un linguaggio ideato dallo stesso Weizenbaum e non più in uso, per semplicità ho riportato quindi una riscrittura del programma in linguaggio Basic di cui l'ultimo editing risale al 1979.

Bisogna ammettere che Eliza è ben congegnato per riflettere quello che in linguistica sarebbe stato poi individuato come *frame semantico* da Charles Fillmore nel 1982; per cui il significato di un dato termine può essere compreso solo nel suo contesto. Frame: scenari ricorrenti, tipici. I legami con le esperienze e credenze che costituiscono la visione del mondo condivisa dai parlanti in atto (il background) e quindi una condizione essenziale per la comprensione linguistica. Le capacità di Eliza, e quindi delle chatbot, sono state fin da subito sovrastimate non tenendo conto del *frame* in cui veniva inizialmente collocata. Quasi a mo' di sberleffo Parry, un chatbot del 1972 ideato dallo psichiatra Kenneth Colby, come contro risposta, simulava la conversazione di un paziente schizofrenico, e nello stesso anno fu tentato un dialogo tra lui e il suo più noto omologo (da cui la nota in esergo di queste pagine).

L'accettazione di una concezione meccanicistica del mondo implica tuttavia una o più concezioni meccanicistiche del costruito che chiamiamo mente; così l'epistemologo statunitense Thomas Kuhn nel suo saggio "*La struttura delle rivoluzioni scientifiche*" (1962), rifiuta la concezione tradizionale della scienza come accumulazione progressiva di nuove scoperte e afferma invece che in certi momenti (detti rivoluzionari) viene interrotto il rapporto di continuità con il passato e viene iniziato un nuovo corso, in modo *non* completamente razionale. Sarà lo stesso Weizenbaum a farsi autocritica nel suo libro "*Computer Power and Human Reason: From Judgment to Calculation*" (1976), in cui discute i limiti dei calcolatori affermando che le visioni antropomorfe dei computer sono delle ingiustificate riduzioni degli esseri umani. Torniamo coi piedi per terra.

Oggi, attraverso Internet e in pochi minuti, sul solo Facebook saranno generati più di 20 milioni di post, su Snapchat oltre 3 milioni di messaggi e su Instagram ben più di 250.000 foto verranno caricate, con a corollario molti tag su come trovare (e, implicitamente, su come interpretare le foto). Non è difficile immaginare di poter trovare dei pattern ricorsivi all'interno dei dialoghi quotidiani e delle azioni compiute sul web. Però più questo diventa pieno di dati, e quindi dispersivo, più si fa urgente trovare dei

metodi per riuscire a tenere un utente il più tempo possibile su una pagina per poterlo acquistare come cliente.

Le chatbot attuali, dette anche più semplicemente bot, rappresentano una evoluzione delle relazioni online. I bot di cui si tratta fra queste righe, è bene precisarlo quindi, non rappresentano un tentativo di creare una coscienza, né tanto meno di psicanalizzare il prossimo (al più solo di ricavare dati statistici buoni per vendite e pubblicità). Certo, sono servizi ancora in una fase iniziale del loro sviluppo e della loro integrazione nella messaggistica di supporto e nella loro scelta da parte dell'utente, ma è facile prevedere che saranno sempre più utilizzati dalle varie assistenze clienti (*customer service*).

Nel caso di Facebook Messenger, che ha dato una spinta alla moda nell'ultimo periodo, i bot assumono il ruolo di intermediari tra una azienda ed un cliente finale, fungendo da supporto automatizzato per informazioni su prodotti o nella ricerca dei contenuti, ad esempio. Attraverso lo spazio della messaggistica si riesce oggi a portare a termine anche acquisti, senza barcamenarsi in continui rimandi di pagine web. L'utente non è più bombardato da pubblicità invasive, è lui stesso a contattare i bot e a decidere il numero di volte in cui desidera essere aggiornato.

Attenzione però, tutti i bot sono limitati a un sottoinsieme di argomenti. Nel programma ELIZA, il dominio era una sessione di terapia, e la personalità del bot era quella di un terapeuta rogersiano. Dominio e personalità non necessariamente hanno bisogno di essere strettamente accoppiati, anche se un bot-commerce ha bisogno di poter prevedere ogni possibile domanda sui prodotti dell'azienda, e alcune preferiscono fare del proprio bot una mascotte stilizzandolo in questa o quella forma (antropizzare un bot è comunque un buon modo per renderlo simpatico ad una vasta fetta di utenza). Internet allora è visto come un enorme laboratorio, un cantiere aperto, in cui si sta secondo dopo secondo formando il più grande corpus testuale della storia. Grande si intende qui esclusivamente come volume, non come spessore culturale. Castronerie, errori e pensieri opinabili occorreranno comunque a dare forma, giustamente, a questo enorme corpus.

Per provare a comprendere come funzionano i chatbot, prendiamo come esempio un semplice codice per la lingua inglese scritto in Python (<https://github.com/lizadaly/brobot/>). Scegliamo come linguaggio Python perché i programmatori che lo usano e lavorano con la NLP (*Natural Language Processing*) hanno due grandi librerie di alto livello tra cui scegliere: TextBlob e spaCy.

La prima è una libreria Python per l'elaborazione di dati testuali, ovvero una semplice API per maneggiare l'elaborazione del linguaggio naturale (NLP) in compiti come l'analisi grammaticale, l'estrazione di un sostantivo in una frase, sentiment analysis, la classificazione, la traduzione, e altro ancora. SpaCy è un'altra libreria, maggiormente indirizzata al *deep learning* e che in queste poche pagine non vedremo.

Cominciamo quindi ad entrare nel vivo del funzionamento di un basilare bot, partendo dal più tipico esempio di inizio dialogo (le righe anticipate dal carattere cancelletto sono dei commenti al codice)

```
# PAROLE CON CUI RISPONDERE IN CASO DI SALUTO
GREETING_KEYWORDS = ("hello", "hi", "greetings", "sup", "what's
up",)
GREETING_RESPONSES = ["sup bro", "hey", "*nods*", "hey you get
my snap?"]

def check_for_greeting(sentence):
# SE UNA DELLE PAROLE INSERITE DALL'UTENTE È
# NELLA LISTA IL CHATBOT RISPONDE CON UNA
# PAROLA A CASO DALL'ARRAY DI OUTPUT
    for word in sentence.words:
        if word.lower() in GREETING_KEYWORDS:
            return random.choice(GREETING_RESPONSES)
```

Questo è la più semplice implementazione possibile di un chatbot: esso cioè cerca l'espressione dell'utente per una o più note parole chiave e restituisce una delle diverse possibili risposte. Venuta meno la parola chiave, quel "hello" inviato per primo dal bot, la conversazione non prende mai luogo. Una azienda online può allora non considerare vantaggioso aspettare che sia l'utente a richiedere di avviare una chat. Controllando, ad

esempio, che il visitatore sia sulla pagina di un prodotto specifico per più di 30 secondi e interpretando questo dato come la ricerca di ulteriori informazioni circa il prodotto in vendita sulla pagina, ovvero una dimostrazione di interesse dell'utente, potrebbe aprirsi la chat con un *trigger* e presentare una finestra di dialogo in cui sarà possibile inserire un vasto numero di contenuti, dal testo alle immagini. Il trigger possiamo descriverlo qui, sommariamente, come un timer associato alla pagina, si attiva appena l'utente entra dentro questa (si può però modificare ed inserire ad esempio variazioni se l'utente muove il mouse verso un link). Il contenuto di questa prima finestra, ovvero il saluto automatico del bot, deve cercare di catturare l'attenzione del visitatore. Pertanto dovrà essere, per stile e lessico, ben congegnato al target aziendale e a quello che si aspetta sia l'utente medio.

Il ciclo principale di un chatbot dovrebbe eseguire le seguenti operazioni:

- a)** Fare qualche iniziale pre-elaborazione del testo dell'utente, in modo da poter agganciare dei controlli per ingresso testuali non sicuri quali frasi non ben formate o frasi **contenti** parole offensive.
- b)** Chiedere a TextBlob di analizzare l'input per noi, ovvero tokenizzarlo e farne la part-of-speech-tagging.
- c)** Eseguire una serie di routine progettata per estrarre le informazioni dalle espressioni dell'utente in modo strutturato.
- d)** Comporre una risposta che meglio corrisponda alla dichiarazione inviata dall'utente.
- e)** Eseguire una post-elaborazione per garantire nel miglior modo possibile che il nostro bot non possa “comportarsi male”, ovvero rispondere con frasi volgari ad esempio.

Le scarse righe di codice che adesso vedremo **affibbiano** una part-of-speech-tagging ai singoli token, ovvero associano ciascuna parola in un corpus ad una parte del discorso. Più precisamente, più che una parola (una entità difficile da stabilire, come variamente sostenuto dal linguista Edward Sapir) qui si parla di token, ovvero si assegna una parte di discorso ad ogni blocco di caratteri testuali divisi, a sinistra e destra, da spazi delimitatori (tanto per fare un esempio la locuzione latina “ad hoc”, ampiamente usata anche nel vocabolario italiano, sarebbe trattata dalla part-of-speech-tagging come due parole indipendenti).

```
def respond(sentence):
```

```
# ANALIZZA FRASE IN ENTRATA DALL'UTENTE E TROVA I  
# TERMINI CANDIDATI PER FORMARE UNA RISPOSTA
```

```

cleaned = preprocess_text(sentence)
parsed = TextBlob(cleaned)
# ESTRARRE LA PIÙ RILEVANTE PARTE DELLA PART-OF-SPEECH
pronoun, noun, adjective, verb =
find_candidate_parts_of_speech(parsed)

# SE ABBIAMO DETTO QUALCOSA CIRCA IL BOT COSTRUIRE
# LA RISPOSTA SCARTANDO GLI ALTRI CANDIDATI

resp = check_for_comment_about_bot(pronoun, noun, adjective)

[...]

```

La conversione utente-bot non include in generale pronomi come “loro” o “essi”. Ricordiamo che ci stiamo rivolgendo ad una macchina per poter portare a termine un acquisto non per parlare dei nostri problemi familiari. Può darsi quindi che la frase digitata dall'utente non contenga nessuno dei pronomi che ci aspettiamo

```

# SE NON ABBIAMO OTTENUTO NULLA LANCIARE
# UNA RISPOSTA A CASO FRA QUELLE PREVENTIVATE

if not resp:
    resp = random.choice(NONE_RESPONSES)

logger.info("Returning phrase '%s'", resp)
# CONTROLLARE DI NON AVERE UN OUTPUT OFFENSIVO
filter_response(resp)

return resp

```

Possiamo immaginare una frase come una tupla di co-occorrenze di elementi di parti del discorso in modo da poterlo poi ricomporre

```

def find_candidate_parts_of_speech(parsed):

```

```
# DATO UN INGRESSO ANALIZZATO RESTITUIRE UNA  
# TUPLA DI PRONOME, NOME, AGGETTIVO E VERBO
```

```
pronoun = None  
noun = None  
adjective = None  
verb = None  
for sent in parsed.sentences:  
    pronoun = find_pronoun(sent)  
    noun = find_noun(sent)  
    adjective = find_adjective(sent)  
    verb = find_verb(sent)  
    logger.info("Pronoun=%s, noun=%s, adjective=%s, verb=%s",  
pronoun, noun, adjective, verb)  
    return pronoun, noun, adjective, verb
```

```
def find_pronoun(sent):
```

```
# RESTITUISCE _NONE_ SE NESSUN CANDIDATO DEL  
# PRONOME È STATO TROVATO NELLA FRASE
```

```
    pronoun = None
```

```
    for word, part_of_speech in sent.pos_tags:
```

```
# DISAMBIGUARE I PRONOMI
```

```
        if part_of_speech == 'PRP' and word.lower() == 'you':
```

```
            pronoun = 'I'
```

```
        elif part_of_speech == 'PRP' and word == 'I':
```

```
# SE L'UTENTE SI È PALESATO ALLORA VICEVERSA
```

```
            pronoun = 'You'
```

```
    return pronoun
```

Superati un'intera serie di controlli dovrebbe approssimarsi la risposta

```
def check_for_comment_about_bot(pronoun, noun, adjective):
```

```
# CONTROLLARE SE L'INPUT DELL'UTENTE ERA O MENO  
# CIRCA IL BOT STESSO
```

```
    resp = None
```

```
    if pronoun == 'I' and (noun or adjective):
```

```

    if noun:
        if random.choice((True, False)):
            resp =
random.choice(SELF_VERBS_WITH_NOUN_CAPS_PLURAL).format
(**{'noun': noun.pluralize().capitalize()})
        else:
            resp =
random.choice(SELF_VERBS_WITH_NOUN_LOWER).format(**{'noun': noun})
        else:
            resp =
random.choice(SELF_VERBS_WITH_ADJECTIVE).format(**{'adjective': adjective})
    return resp

```

Come per il modello Eliza, già visto, si possono prevedere delle frasi preconfezionate che aspettano solo l'integrazione della variabile digitata dall'utente. In questo caso però, la nuova tecnologia della *parserizzazione* permette di individuare esattamente la parte del discorso che ci interessa e copiare solo quella, mentre in Eliza ci si limitava (sulla scorta di previste frasi semplici) alla individuazione del soggetto, alla sua sostituzione e allo spostamento in blocco di quanto digitato dall'utente in modo che, secondo la sintassi inglese, una frase affermativa diventasse una interrogativa. Oggi invece possiamo prevedere risposte che attendono un sostantivo diretto, ad esempio

```

# MODELLO PER RISPOSTE CHE INCLUDONO UN
# SOSTANTIVO DIRETTO
SELF_VERBS_WITH_NOUN_CAPS_PLURAL = [
    "My last startup totally crushed the {noun} vertical",
    "Were you aware I was a serial entrepreneur in the {noun} sector?",
    "My startup is Uber for {noun}",
    "I really consider myself an expert on {noun}",
]

```

Il codice integra anche altri array per frasi randomiche con per variabili

aggettivi o nomi-sostantivi scritti interamente in minuscolo, sono strutturalmente **uguale** pertanto è in questo momento è superfluo mostrarle. Andando avanti, è bene fare in modo di rendere il bot immune dagli attacchi di perdigiorno o dalla possibile e impreveduta coprolalia dell'utente frustrato perché non ha ottenuto la risposta che desiderava.

È utile inserire pertanto una lista di insulti con cui schermare le risposte del bot. È appurato che davanti ad un chatbot in molti inizino a sfidarlo, si scatena uno strano meccanismo per il quale dare una voce e un aspetto a un bot fa sì che le persone vogliano metterlo alla prova nei modi più offensivi e inappropriati. Non si tratta di sottostimabile goliardia, un bot il cui *training* di addestramento contempli anche risposte offensive inizierà a dare risposte offensive con conseguente danno di immagine alla azienda che ha deciso di affidarsi all'assistenza clienti automatica.

```
def filter_response(resp):  
    #CONTROLLARE QUELLO CHE HA SCRITTO L'UTENTE  
    tokenized = resp.split(' ')  
    for word in tokenized:  
        if '@' in word or '#' in word or '!' in word:  
            raise UnacceptableUtteranceException()  
        for s in FILTER_WORDS:  
            if word.lower().startswith(s):  
                raise UnacceptableUtteranceException()
```

Esistono online diverse librerie, in vari formati (*.txt, *.xml, *.csv), composte esclusivamente da parole, ripetute in varie forme ortografiche, ritenute mediamente offensive o inappropriate ad una immagine pubblica (*balls, bastard, bitch, biatch bloody, blowjob, blow job, bollock, bollok, boner, etc*). Bisogna anche prevedere che l'utente più smaliziato possa usare delle storpiature delle succitate parole, ad esempio inserendo una chiocciola al posto di una lettera *a*, conscio che in questo modo possa eludere per qualche momento i filtri. È bene quindi che la lista non sia chiusa ma aperta ad accogliere ogni nuova tendenza alla cafonaggine. Ne va del decoro del aziendale.

Questo esempio fin qui visto è ovviamente un modello giocattolo, tuttavia ci permette di poter pensare di comprendere cosa si cela al di sotto del pannello di controllo dato dai vari servizi offerti dalla Wit.ai, gestito da

Facebook, o dalla piccola startup italiana Awhy. Wit.ai funziona esclusivamente se si è registrati con Facebook, limitazione urticante e che vincola a dover dare i propri dati personali alla nota azienda della Silicon Valley.

Prendiamo in esame, a puro titolo di esempio, l'interfaccia offerta da Awhy. Il sito, dopo una necessaria iscrizione, permette di testare il bot in modo gratuito per qualche giorno. Appena *loggati* ci viene segnalato un frammento di codice HTML da inserire nella pagina web della nostra azienda

```
<script type="text/javascript"> awhy_api_url = '//manage.why.it';  
why_contents_base_url = '//widget.why.it/chat'; why_widget_id =  
'309'; instance_id = '369'; instance_name =  
'ay_m_malagnino_gmail_com'; </script> <script  
src='//widget.why.it/chat/whypa.js' type='text/javascript'></script>
```

Questo iniziale punto di partenza ci ricorda, se ce ne fosse bisogno, che i bot di oggi sono molto pragmaticamente rivolti alle aziende su internet. Niente voli pindarici di fantasia sul linguaggio, solo uno strumento per gestire con minore spesa i **constumer sevice** e rendere il servizio di assistenza attivo 24h. La chatbot è un software che comunica con gli utenti tramite servizi di messaggistica istantanea, quali quelle presenti su Facebook, Skype o altre piattaforme di comunicazione, non è un programma indipendente.

Awhy, con la sua interfaccia, ci aiuta a comprendere molto velocemente quali sono i pregi e i limiti di un chatbotter. Vediamo come:

Nella GESTIONE DEI DATI troviamo quelle che sono propriamente le FAQ tipiche di una azienda online, ad esempio domande come “*Come si fa ad ordinare un prodotto?*” oppure “*Dove posso controllare lo stato del mio ordine e trovare il mio tracking number?*” oppure “*Non ho ricevuto il mio ordine. Come posso fare un reclamo?*” e via, per altri venti domande considerate base. Le risposte sono vuote e **spessa** all'iscritto al servizio compilarle. Awhy suggerisce di non essere troppo laconici. Ad esempio alla domanda: “*Quanto costa la spedizione?*” non è gradevole un secco e piccato “10€”, tanto per dirne una, quanto piuttosto una risposta più

cordiale che contenga un frammento della domanda “*La spedizione costa 10€*”. Awby consiglia anche di inserire domande e risposte che siano generiche e non inserire casi d’uso troppo specifici; ad un “*Spedite anche in Germania?*” e consigliabile un “*Spedite anche all'estero?*” e al più sarà la risposta automatica già scritta a precisare che si spedisce anche o solo in Germania, per dire. Si possono ovviamente inserire proprie domande scritte di pugno dall'utente. Questo è quanto, per il resto l'utente si deve incaricare di supervisionare l'uso del chatbot per poterlo far funzionare meglio. Nella sezione REPORTISTICA si trovano report dettagliati sul funzionamento e l’efficienza della chat attraverso un semplice grafico statistico, quindi nella sezione AUTOAPPRENDIMENTO si offre un processo che aiuta l'amministratore a gestire tutte le domande che arrivano al sistema in modo da smistarle nella base di conoscenza, e secondo la corretta priorità. Perché le FAQ possano entrare nel database, dovranno raggiungere l’indice di frequenza stabilito, almeno tre identiche richieste ad esempio. Ogni 30 giorni, tutte le domande non gestite verranno però cancellate. Le altre sezioni del pannello di controllo di Awby, sono indirizzate esclusivamente al lato grafico o a questioni inerenti il timer del trigger per far avviare la chat, che va a questo punto letta anche come una finestra pubblicitaria popup interattiva che invita l'utente a risolvere i suoi eventuali dubbi.

Non tutto il lavoro può essere gestito da un bot, alcune domande altamente specifiche o eccezionali possono essere risolte solo da un centralinista reale, se è presente, oppure si può preventivare un messaggio automatico che invita a scrivere una email all'azienda che provvederà a ricontattare il cliente. Quello che è importa è premunirsi alle possibile storture nell'uso di questa tecnologia. Difatti la chatbot causa con le sue risposte emozioni negative quando non riesce a **processare** bene la frase dell'utente, rabbia o frustrazione, sentimenti o atteggiamenti facilmente identificabili nel linguaggio utilizzato dagli utenti, soprattutto quando iniziano a scrivere insulti, o frasi di sfogo come “*non servi a niente*”. Awby, con lodevole intelligenza, consiglia di preventivare queste situazioni e preparare un set di risposte automatiche come “*Mi dispiace, non posso aiutarti*” oppure “*Non ho capito, potresti spiegarti meglio?*” oppure per i casi più rozzi qualcosa come “*Non è una cosa carina da dire*”. Ricordando l'effetto Eliza, un utente adirato potrebbe sentirsi spiazzato dalla cordialità mostrata dal bot. Frasi ancora più utili per risolvere eventuali blocchi di dialogo sono “*Comprendo la tua frustrazione, prova a formulare diversamente la tua domanda*”, è verosimile che cambiando lessico e invitando l'utente a

scrivere più stringato e preciso l'algoritmo possa finalmente **processare** la frase inviata dall'utente. È utile però inserire un vincolo di errori possibili, non ci si può certo aspettare che la pazienza dell'utente sia infinita, e preparare frasi come *“Mi dispiace che la tua esperienza sia frustrante, per avere maggiore supporto contattaci alla mail xxx@mail.com e un operatore sarà pronto a gestire la tua richiesta”*. Quello che è importante è tenere a mente che la maggior parte delle chat è pensata per assistere il cliente nel completare un'azione specifica, come modificare una prenotazione di viaggio ad esempio.

Benché bisogna sforzarsi di rispettare la privacy dell'utente se il valore delle risposte di una chatbot dipende da parametri specifici quali il nome o il luogo in cui si trova l'utente con il quale sta interagendo, non c'è alcuna valida spiegazione al fatto che queste informazioni vengano perse dalla chatbot nel corso della conversazione stessa. L'importante sarà semmai segnalare, a norme di legge, che alcuni dati della conversazione virtuale saranno salvati per scopi commerciali strettamente interessati al solo utente, senza per questo farlo sentire usato come campione per analisi statistiche.

Gli sviluppi crescenti nel campo dell'Intelligenza Artificiale e della messaggistica istantanea potrebbero essere in grado di sostituire interamente gli operatori di call center determinando così la fine dell'era delle lunghissime attese telefoniche gestite da personale non sempre adeguatamente qualificato.

Concluderei questa breve introduzione ai bot con due singolari casi della più recente cronaca in cui, le risposte automatiche dei bot sono in qualche modo entrate nel dominio della vita quotidiana. Nel marzo del 2016 la Microsoft ha rilasciato “Tay” un chatbot basato sull'intelligenza artificiale, *targettizzato* per la generazione Millennials, dall'aspetto, ancora una volta, di una giovane donna. Pare ci sia un stereotipo sessista che ricorra negli assistenti vocali o nei bot, da Eliza (1966) a Alice (1995), Siri (2010), Alexa (2015) o Cortana (2015) quando il bot prevede di interagire con il pubblico si tende a darle una personalità femminile. Tay per i Millennials non ha fatto eccezione in questo. In meno di 24h dopo Tay è stata “spenta” dopo accuse di razzismo, sessismo e negazione dell'Olocausto. Microsoft è sempre stata piuttosto vaga nello spiegare i meccanismi e gli algoritmi che permettono a Tay di funzionare e, di conseguenza, non è nemmeno

possibile capire nel dettaglio cosa è andato storto. La Microsoft era ben consapevole di questo rischio: Cortana, il loro assistente virtuale, è spesso soggetto a molestie sessuali, e perciò programmato per gestirle, e nessuno può davvero dire di essere stupito se un chatbot in uso sui social media, dalla forma di una ragazza e nelle mani di adolescenti, avrebbe portato ad un simile risultato. Tuttavia per Tay era previsto che imparasse dalle infinite interazioni presenti su Twitter, senza alcuno schermo o tutela. Il problema reale con il quale la Microsoft si è scontrata è relativo al fatto che la maggior parte delle interazioni sui *social media* sono sessiste, razziste e via cantando. Per di più queste interazioni, nelle mani di un bot fatto per ripetere tutto ciò che apprende non potevano che arrivare presto a galla nelle risposte automatiche. Proprio per questo molti assistenti virtuali sono in grado di riconoscere una serie di insulti e di rispondere in modo appropriato, bisogna fare in modo che gli algoritmi siano sostenuti da una lista di parole becere per poterne fare un filtro di risposta. Secondo il Washington Post gran parte della colpa è stata di alcuni *troll* di un gruppo di 4chan. I troll sono, su internet, le persone che fanno cose il cui scopo è irritare gli altri, divertendosi; 4chan è invece una delle più note idiote chat in cui gli utenti possono scrivere di tutto godendo dell'anonimato.

Ad ottobre dello stesso anno è venuto alla ribalta il caso, abbastanza gonfiato, di un chatbot dedicato ad una persona defunta. La 29enne Eugenia Kuyda, una startrupper russa, ha programmato un chatbot di un amico scomparso, di nome Roman. Si sono presto sperticate le riflessioni della stampa sul fatto di come il futuro fosse improvvisamente diventato più vicino, traendo forzatamente una analogia fra la storia di Kuyda e una puntata della nota serie televisiva Black Mirror, in cui nell'episodio "*Be right back*", la protagonista Martha rivive il rapporto col compagno Ash, scomparso il giorno dopo il trasloco in una nuova casa di campagna, grazie a una chat intelligente prima – arricchita dalla replica della voce – e, poi, a un clone sintetico. Nulla di più distante dalla realtà. Kuyda ha nutrito infatti il chatbot con le migliaia di messaggi scambiati con il suo amico attraverso Telegram. Parliamo di appena “ottomila righe” pare più che sufficienti a scatenare il trambusto della stampa, più che nel mondo reale. Ci sarebbero più righe di testo scritte da William Shakespeare o da Karl Marx, ed è facile immaginare come l'invenzione di un chatbot nutrito dai loro testi non sarebbe percepito dalla comunità molto più che un gioco di citazioni. Ecco che ci troviamo quindi di fronte ad un ennesimo e nuovo caso di effetto Eliza: dimenticando il contesto l'effetto diventa magico. Il chatbot digitalizzato è programmato per parlare volentieri della sua vita e

del suo lavoro ma, curiosità a parte, nel complesso la chiacchierata è frustrante, forse solo gli amici più stretti possono cogliere, di tanto in tanto, qualche chicca dimenticata. Il grande **interessete** suscitato però dal fallimento del chatbot intelligente Tay della Microsoft, dall'omaggio all'amico defunto in @Roman (di cui è ancora possibile scaricare qualche versione online), fino al pompaggio del mercato dei bot con Facebook e Messangers; mostrano con ogni evidenza che nel bene e nel male i bot diventeranno parte sempre più integrante della nostra vita. Bisogna però spendere due righe ancora e riportare a due concetti chiave stabili dal linguista americano Noam Chomsky: competenza ed esecuzione. La competenza è la capacità di comprendere il linguaggio, nel caso in specie diremmo che è la capacità di un algoritmo di un bot di computare adeguatamente un discorso. L'esecuzione è la capacità di formare nuove frasi. Se nel già citato Edward Sapir è la parola l'unità minima del linguaggio in Chomsky lo è la frase, e quest'ultima ha una **caratteristica** particolare: è sempre diversa. Per l'americano noi comprendiamo una struttura sottostante la frase, cioè la relazione fra le posizioni dei singoli argomenti di questa, mentre un bot può allo stato attualmente soltanto emulare la struttura superficiale. E la emula sino a sclerotizzarla, rendendola fissa. Un bot è quindi quanto più di distante dal linguaggio umano perché questo non si basa (almeno non si basa esclusivamente) su inferenze statistiche. Il linguaggio genera nuove frasi ed è aperto a nuove parole, la cui ontologia è rimandata al mondo esterno certo non al corpus testuale preso in esame che, per quanto vasto potrà essere, sarà sempre chiuso.

Sitografia consultata

<https://www.theatlantic.com/technology/archive/2014/06/when-parry-met-eliza-a-ridiculous-chatbot-conversation-from-1972/372428/>

<https://it.semrush.com/blog/chatbot-cosa-sono/>

<https://futurism.com/images/the-history-of-chatbots-infographic/>

<http://www.awhy.it/chatbot/>

<http://www.neuroscienze.net/?p=324>

<http://www.softfobia.com/news/mobile/13/cosa-sono-i-chatbot-come-funzionano-intelligenza-artificiale-marketing#sthash.xZvZQQEA.dpuf>

<http://www.federica.unina.it/sociologia/metodologia-e-tecnica-della-ricerca-sociale/il-concetto-di-paradigma-nelle-scienze-sociali/>

<https://chatbotsmagazine.com/a-brief-history-of-bots-9c45fc9b8901>

<https://apps.worldwritable.com/tutorials/chatbot/>

<https://github.com/lizadaly/brobot/>

https://www.unigre.it/sito/PUG_HG_03O820150936/web/LaGregoriana/46_38_busa_it.pdf

<https://chatbotslife.com/ultimate-guide-to-leveraging-nlp-machine-learning-for-you-chatbot-531ff2dd870c>

<https://textblob.readthedocs.io/en/dev/>

<http://www.ilpost.it/2016/03/25/tay-microsoft-ai-razzista/>

<https://www.wired.it/attualita/tech/2016/10/10/ho-chattato-col-bot-di-un-morto/>

<https://spacy.io/>

<http://corpora.dslo.unibo.it/People/Nissim/FdL-0910/lecture181109.pdf>

<http://www.awhy.it/awhy-blog/pagina-faq-customer-care/>

<https://users.dimi.uniud.it/~angelo.montanari/Udine2014-2.pdf>

<https://everything2.com/title/ELIZA+source+code+in+BASIC>

http://www.sabrinamarazzi.it/1/il_metodo_rogersiano_469046.html