

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

SCUOLA DI INGEGNERIA E ARCHITETTURA
Corso di Laurea Magistrale in Ingegneria e Scienze Informatiche

LA RIVOLUZIONE DEI CHATBOT:
SVILUPPO DI UN SISTEMA CONVERSAZIONALE
IN AZIENDA

Tesi di Laurea di:
PATRYK WOJTOWICZ

Relatore:
Prof. ANDREA OMICINI

ANNO ACCADEMICO 2016–2017
SESSIONE III

Indice

Introduzione	1
1 Chatbot	3
1.1 Storia	4
1.2 Motivazioni	5
1.3 Vantaggi	6
1.4 Funzionamento	8
1.5 Categorie di chatbot	9
1.6 Piattaforme di sviluppo	10
2 Requisiti	17
3 Analisi dei requisiti	23
3.1 Glossario	24
3.2 User Stories	25
3.3 Modello del dominio	26
4 Progetto	29
4.1 IBM Watson Conversation	32
4.2 Configurazione iniziale	34
4.3 Intents	35
4.3.1 Welcome	36
4.3.2 RicercaSoluzione	37
4.3.3 ConfermaSoluzione	38
4.3.4 SoluzioneNonFunzionante	39
4.4 Entity	40
4.4.1 Prodotto	43
4.4.2 Difetto	44

4.5	Dialog	45
4.5.1	Benvenuto	51
4.5.2	Login	52
4.5.3	Utente necessita assistenza	57
4.5.4	Ricerca della soluzione	63
4.5.5	In altri casi	68
4.6	Deploy	68
4.6.1	Application Backend	70
4.6.2	Application Frontend	73
4.6.3	Improve	74
4.7	Test	75
4.7.1	Key Performance Indicator	76
4.8	Evoluzioni	77
	Conclusion	79

Introduzione

Per un'azienda la comunicazione con i clienti è uno degli aspetti più importanti da valorizzare, tuttavia molte interazioni continuano ad avvenire tramite strategie di contatto che impattano negativamente sull'esperienza complessiva.

L'incremento continuo del numero delle persone che utilizzano app di messaggistica ha dato spazio a nuovi ambienti da poter sfruttare per cambiare i propri processi di business ed offrire una scelta alternativa in grado di rendere gli utenti più connessi a livello personale.

La digitalizzazione ha profondamente modificato le aziende poiché i metodi convenzionali utilizzati non erano in grado di garantire le aspettative dei clienti. Oggi i clienti si aspettano che le aziende siano in grado di rispondere immediatamente alle loro richieste con disponibilità 24/7, servizi personalizzati in tempo reale e senza errori [4].

L'impatto della digitalizzazione in questo periodo è inevitabile e molte aziende ne hanno approfittato. Uno studio di Sloan Management [19] del MIT del 2015 ha rilevato che le aziende che ottengono più del 50% dei loro ricavi dagli ecosistemi digitali, hanno una crescita del 32% superiore e dei margini di profitto del 27% più alti rispetto ai loro concorrenti.

Uno degli effetti diretti della digitalizzazione è la digital disruption, ovvero il cambiamento verificatosi quando nuove tecnologie digitali e modelli di business influenzano la proposta di valore di beni e servizi esistenti [16]. Questo processo aiuta un'organizzazione ad essere più attenta ai cambiamenti dei valori aziendali, a personalizzare i prodotti e le risposte con i clienti a livello personale e ad automatizzare tutti i processi in modo che i lavoratori possano concentrarsi su quello che è realmente importante.

Lo sviluppo di un sistema conversazionale è una delle scelte vincenti che un'azienda deve intraprendere per garantire performance elevate, riducendo tempi e costi.

Partendo da un bisogno reale di un'azienda è stata progettata una nuova modalità di interazione con i clienti cercando di trovare dei compromessi tra capacità tecnologica, standard ed aspettative di mercato. L'obiettivo principale è quello di ottenere un sistema da testare per ricavare feedback che serviranno in seguito per la realizzazione del prodotto definitivo. La realizzazione del prototipo è avvenuta cercando di sperimentare ed imparare nuove tecnologie per riuscire a capire e vivere il cambiamento evitando di subirlo.

Capitolo 1

Chatbot

Il bot, abbreviazione di robot, in terminologia informatica indica un programma o uno script utilizzato per l'automazione di compiti troppo gravosi o complessi per gli utenti umani [5]. Il tipo di bot più diffuso è il web crawler [9], un bot per navigare sul World Wide Web in grado di analizzare i contenuti di una rete in modo metodico e automatizzato, in genere per conto di un motore di ricerca per acquisire ed indicizzare una copia testuale di tutti i documenti visitati. Sui canali IRC ci sono bot dedicati allo spamming rivolto automaticamente a tutti gli utenti che entrano in chat, all'offerta di servizi di file sharing integrati nel sistema di chat oppure come gestori di canali. Nei Videogiochi ci sono bot che partecipano come avversari virtuali al posto di giocatori reali oppure come supporto a giocatori reali, come aimbot per aiutare a prendere la mira in giochi di tipo sparatutto. Su Wikipedia ci sono bot che svolgono compiti di riordino automatico delle pagine, compilazione dei collegamenti, correzioni dei reindirizzamenti ed altre funzionalità. L'integrazione di bot in ecosistemi di messaggistica con lo scopo di dialogare e assistere gli utenti individua una nuova tipologia denominata chatbot, un software in grado di simulare una conversazione tra uomo e macchina. Può essere utilizzato per automatizzare un servizio, sostituendo delle persone, e di conseguenza risparmiando denaro. Questa caratteristica è alla base dell'interessamento da parte di molte aziende, ed è per questo che i chatbot sono diventati lo strumento di marketing di una nuova generazione.

1.1 Storia

```

Welcome to

      EEEEE LL      IIII ZZZZZZ  AAAAA
      EE   LL      II   ZZ   AA  AA
      EEEEE LL      II   ZZZ  AAAAAA
      EE   LL      II   ZZ   AA  AA
      EEEEE LLLLL IIII ZZZZZZ  AA  AA

Eliza is a mock Rogerian psychotherapist.
The original program was described by Joseph Weizenbaum in 1966.
This implementation by Norbert Landsteiner 2005.

ELIZA: Is something troubling you ?
YOU:   Men are all alike.
ELIZA: What is the connection, do you suppose ?
YOU:   They're always bugging us about something or other.
ELIZA: Can you think of a specific example ?
YOU:   Well, my boyfriend made me come here.
ELIZA: Is it important to you that your boyfriend made you come here ?
YOU:   He says I'm depressed much of the time.
ELIZA: I am sorry to hear that you are depressed.
YOU:   It's true. I am unhappy.
ELIZA: Can you explain what made you unhappy ?
YOU:   █

```

I chatbot forniscono informazioni ed eseguono azioni, è importante che siano in grado di comprendere ciò che gli viene chiesto e che riescano a soddisfare rapidamente le richieste.

L'origine dell'applicazione di intelligenza artificiale legata ad una conversazione tra umani è nata negli anni cinquanta con il test di Turing di Alan Turing, matematico e crittografo britannico, considerato uno dei padri dell'informatica. Il test consisteva nel far conversare testualmente tre persone attraverso una macchina e ad un certo punto sostituire una di queste persone con una macchina che imitasse comportamento e risposte umane. Se le altre due persone non si accorgevano della sostituzione, il test indicava che la forma di intelligenza artificiale fosse in grado di pensare. Il primo esempio di chatbot risale al 1966, Joseph Weizenbaum progettò ELIZA, un software creato per cercare di ingannare gli utenti facendogli credere di conversare con un essere umano reale. L'algoritmo era molto semplice infatti l'input dell'utente veniva processato cercando di riconoscere parole o frasi. Di seguito l'output veniva dato da risposte precedentemente preparate, a seconda dell'input, allo scopo di far progredire una conversazione in modo

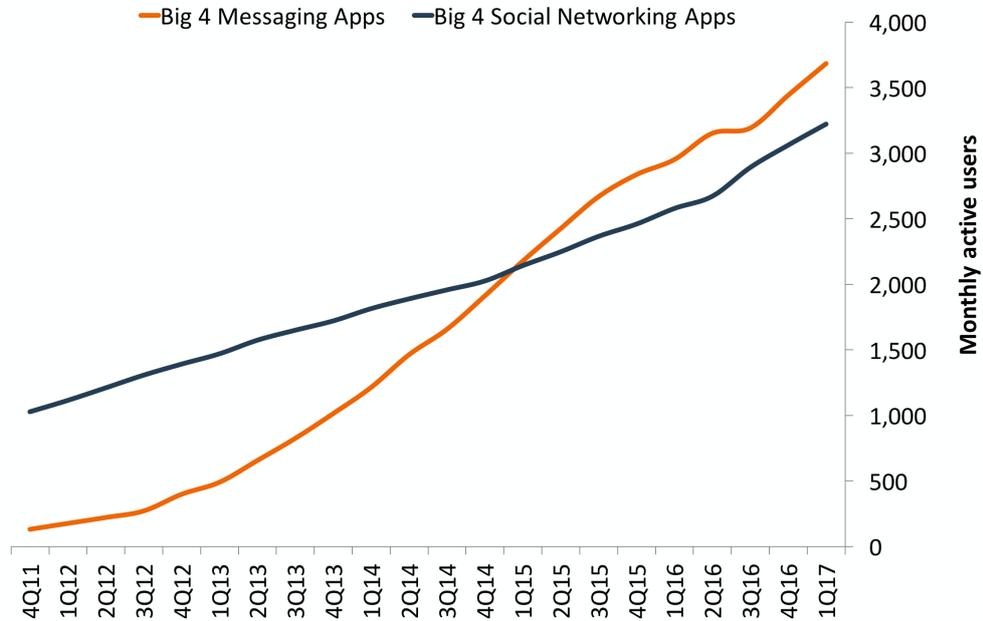
apparentemente significativo. Si riuscì così, per la prima volta, a creare l'illusione di un interlocutore umano con un software.

1.2 Motivazioni

Il motivo per il quale si sta vivendo un'esplosione nell'ambito dei chatbot è dovuto anche dal fatto che per la prima volta in assoluto il numero di persone che utilizzano le app di messaggistica è maggiore rispetto al numero di persone che usano i social network. Secondo Business Insider [3] un utente medio possiede circa 27 app sul proprio dispositivo mobile, ma ne utilizza regolarmente solo 5. La maggioranza dei consumatori ha sviluppato una forte riluttanza nel provare nuove applicazioni, come confermato anche dai dati forniti da Localytics [2] che mostra come dopo un solo utilizzo il 23% degli utenti disinstalli l'applicazione. Le ricerche di comScore indicano invece che il 16% del tempo speso su desktop è passato su dispositivi mobili. Questa crescita va di pari passo con l'aumento esponenziale di download di applicazioni mobili, ma sempre secondo comScore, il 50% del tempo speso su dispositivi mobili è impiegato nell'app preferita. Le app di messaggistica vanno controcorrente e viene stimato da comScore che vengano aperte 9 volte al giorno con un indice di apertura medio quattro volte maggiore rispetto a qualunque altra app installata sul dispositivo mobile. Per questo motivo le app di messaggistica sono diventate la piattaforma principale per i chatbot, come per esempio Facebook ha fatto con Facebook Messenger. Nell'Aprile del 2016 Facebook ha annunciato il supporto dei chatbot all'interno della piattaforma Messenger ed oggi l'app di messaggistica può essere vista come un negozio per migliaia di applicazioni integrate nelle nostre conversazioni quotidiane. I dati mostrano che da Aprile 2016 ad oggi sono più di 100.000 i bot che sono stati aggiunti e più di un miliardo gli utenti che ne fanno uso. Nell'espansione dei chatbot non bisogna dimenticare che i costi ed i tempi di sviluppo sono molto inferiori rispetto alle app. Al giorno d'oggi è possibile creare chatbot sofisticati sfruttando tool e framework offerti gratuitamente ed utilizzare le interfacce grafiche delle piattaforme di messaggistica per la distribuzione.

Messaging Apps Have Surpassed Social Networks

Global monthly active users for the top 4 messaging apps and social networks, In millions



Note: Big 4 messaging apps are WhatsApp, Messenger, WeChat, Viber.
Big 4 social networks are Facebook, Instagram, Twitter, LinkedIn
Source: Companies, Apptopia, TechCrunch, BI Intelligence estimates, 2017

BI INTELLIGENCE

1.3 Vantaggi

I chatbot possono essere sviluppati con l'obiettivo di aiutare gli utenti a realizzare attività oppure ottenere informazioni specifiche. I vantaggi principali che si possono individuare sono:

- un servizio più rapido per ridurre il tempo di attesa dei clienti e di conseguenza migliorare la user experience;
- la possibilità di essere disponibile 24 ore su 24 e tutti i giorni della settimana;
- facilità nell'utilizzo dovuta all'assenza di curva di apprendimento per utilizzarli.

- riduzione del costo per il customer service. Costruire e mantenere un chatbot è molto più economico che assumere un dipendente. Si può ipotizzare senza problemi che questo gap aumenti nel tempo con il progredire della tecnologia.
- la capacità di trovare le informazioni necessarie in una base di dati molto più veloce di una persona appositamente addestrata.

Tutti questi vantaggi sono individuabili nell'ambito del marketing aziendale, ma sono anche strumenti indispensabili per molti altri casi d'uso. L'importanza che i giganti della tecnologia come Google, Facebook, Microsoft, IBM e Amazon danno ai chatbot è un forte indicatore del fatto che questa tecnologia avrà un ruolo chiave in futuro. Non sempre però è necessario sviluppare chatbot per avere dei vantaggi, esistono infatti i chatbot conversazionali che sono focalizzati sull'avere una conversazione con l'utente ed il loro obiettivo è incentrato sull'intrattenimento.

Un chatbot, rispetto ad un essere umano, può essere sempre paziente, attento ad ogni singolo dettaglio e capace di analizzare l'interlocutore e valutare in un istante i suoi bisogni e desideri.

L'enorme potenziale è chiaramente visibile, ma ovviamente sono presenti anche degli svantaggi. Spesso succede che si possa cadere in una delle seguenti situazioni:

- Non sempre funziona e di conseguenza non è possibile raggiungere il proprio obiettivo;
- Ci vuole più tempo per raggiungere il risultato richiesto e quindi il chatbot perde di utilità.

Escluse le situazioni in cui un essere umano è in grado di distinguere se sta comunicando con un bot o no, bisogna decidere se è giusto o sbagliato che il chatbot si identifichi all'utente come intelligenza artificiale.

Un essere umano ha aspettative diverse in base all'interlocutore quindi è giusto che l'intelligenza artificiale si identifichi come tale senza cercare di impersonificare una persona reale. In un dialogo con un chatbot non è necessario investire emotivamente perché questo investimento non può essere ricambiato, inoltre il chatbot potrebbe rispondere male e quindi offrire un'esperienza negativa. Tuttavia l'imitazione del comportamento umano,

concetto diverso dall'impersonificazione, è giusta per poter offrire un servizio migliore. Anche nel caso in cui un utente parlasse con un bot, sostituito da un essere umano in quel momento, è opportuno che l'essere umano si identifichi come tale perché l'utente potrebbe scegliere di rivelare informazioni diverse in base all'interlocutore e nel caso in cui venga riconosciuto andrebbe ad impattare sull'esperienza complessiva del servizio. Inoltre da un chatbot, un utente, si aspetta risposte rapide e accurate ad un livello tale che un essere umano non potrebbe offrire. Wizard of Oz development è una pratica che viene sfruttata nell'ambito della ricerca sull'intelligenza artificiale per sostituire algoritmi di IA con essere umani per apprendere in prima persona come gli utenti utilizzino un determinato servizio.

1.4 Funzionamento

Il principio di funzionamento di un chatbot è semplice, dato un input di un'utente, restituisce una risposta. In pratica le cose però sono più complicate. Innanzitutto, il chatbot deve comprendere l'input. Esistono due tecniche principali per ottenere questo risultato: pattern matching e classificazione degli intent. Un approccio di tipo pattern matching richiede un elenco di possibili pattern di input. Il punto positivo di questo approccio è che gli schemi possono essere letti dagli umani e la fase di modellazione degli input risulta semplice. Questo approccio è molto limitato perché i pattern sono costruiti manualmente e quindi non sono flessibili nei vari casi d'uso. L'approccio basato su classificazione degli intent si basa su tecniche di apprendimento automatico. Come per ogni algoritmo di intelligenza artificiale è necessario un training set per addestrare un classificatore con il compito di estrarre gli intent dall'input di un utente. Il classificatore impara anche dalle conversazioni con l'utente in modo da aumentare, con il tempo, l'efficacia. Una volta che il chatbot capisce la richiesta dell'utente genera una risposta in base all'input ricevuto ed al contesto della conversazione. La risposta più semplice è di tipo statico, ovvero un elenco di risposte da utilizzare in base all'input. Questo tipo di risposte può essere un semplice testo oppure un modello con delle variabili calcolate sul momento. Un altro tipo di risposta è quella dinamica. Viene utilizzata una base di conoscenza per ottenere un elenco di potenziali risposte alle quali viene assegnato un

punteggio per scegliere la risposta migliore. Questo approccio è utilizzato per chatbot che come obiettivo primario hanno la risposta a domande. Infine è possibile utilizzare un terzo tipo di risposta, che si basa su tecniche di deep learning, per formare un modello generativo che, dato un input, genera la risposta. L'efficacia aumenta con l'aumentare degli esempi di conversazione che sono utilizzati, ma resta sempre la possibilità di risposte inaspettate. Qualsiasi sia la tipologia della risposta, l'input corrente non è sufficiente, ma serve definire e modellare il contesto nel quale il chatbot opera.

1.5 Categorie di chatbot

Partendo dal presupposto che la funzione primaria dei chatbot sia quella di soddisfare delle richieste da parte di utenti è possibile suddividerli in categorie in base al contesto in cui operano e la funzione alla quale vengono destinati. Le tre categorie in base alla funzione sono:

- Commerce Bots permettono di visualizzare ed acquistare i prodotti rimanendo sempre sulla piattaforma di chat, possiamo parlare quindi di "commercio conversazionale". Shopsyng è stata la prima Azienda in campo fashion ad integrare un chatbot all'interno del proprio e-commerce.
- Customer Service Bots programmati per rispondere a richieste degli utenti nel minor tempo possibile. KLM è la prima compagnia aerea ad aver integrato il proprio chatbot nella piattaforma Facebook Messenger con lo scopo di gestire e ottimizzare il check-in del biglietto aereo.
- Content Bots permettono, in tempo reale, di recuperare e proporre informazioni su un determinato argomento, prodotto o servizio da un utente anche senza un'esplicita richiesta. Sono Chatbot in grado di selezionare contenuti in base a quello che potrebbe interessare all'utente, diventando a tutti gli effetti dei promotori.

Oltre alla divisione per funzione è possibile individuare altre categorie in base a delle caratteristiche particolari:

- Ottimizzatore è la tipologia di chatbot più ampia che viene utilizzata per facilitare e snellire determinate operazioni. In pratica cercano di

migliorare le modalità tradizionali di determinati processi su app o web.

- Proattivo, utilizzato per offrire informazioni in modo rapido.
- Social per creare una chat di gruppo per riunire un certo numero di persone in una community intorno ad un determinato argomento.
- Scudo, questa tipologia di chatbot viene utilizzata per aumentare la user experience degli utenti. A differenza della categoria degli ottimizzatori si cerca di mascherare il concetto di conversazione con un computer con un qualcosa di più simile ad un operatore reale.
- Chatty con lo scopo di allargare e solidificare la user base, intrattenendo conversazioni, solo per cercare di creare un legame con lo user.
- Super Bot per identificare gli assistenti personali intelligenti con il ruolo di supporto alle attività umane come Siri, Google Assistant o Alexa.
- One Trick Pony, permettono di trasformare immagini in diversi tipi di contenuti virali e divertenti.

1.6 Piattaforme di sviluppo

Le piattaforme per sviluppare chatbot si possono raggruppare in tre tipi:

- Piattaforma senza programmazione, orientate ai compiti da svolgere. Non è richiesta alcuna conoscenza di programmazione e quindi sono semplici e con bassa curva di apprendimento. Sono limitati allo sviluppo di chatbot semplici, ma con poco tempo è possibile ottenere un ottimo risultato. Lo svantaggio più grande è che hanno poca o nessuna capacità di elaborazione del linguaggio naturale e quindi risultano non adatti a chatbot complessi. Alcuni esempi sono: Chatfuel, ManyChat, Motion.ai e molti altri che differiscono in termini di maturità, usabilità della GUI e potenza di elaborazione del linguaggio naturale.

- Piattaforme orientate alla conversazione con l'obiettivo principale di avere una conversazione flessibile senza considerare uno scenario orientato alle attività. Queste piattaforme utilizzano in genere linguaggi come AIML (Artificial Intelligence Markup Language) per modellare le interazioni con l'utente ed estrarre informazioni dall'input. La scalabilità non è assicurata se i pattern sono costruiti manualmente e le capacità di estrazione delle informazioni sono limitate in confronto all'NLP. Uno degli esempi più noti è Pandorabots
- Piattaforme dei giganti della tecnologia che cercano di avere una bassa curva di apprendimento e, allo stesso tempo, un forte potere espressivo. Queste piattaforme raccolgono le best-practice del settore e sono molto più complesse. Alcuni esempi sono: DialogFlow, Watson, Wit.ai e LUIS.

I canali più usati sui quali distribuire chatbot sono:

- Facebook Messenger è un software di messaggistica di Facebook che ad Aprile 2017 ha raggiunto 1.2 miliardi di utenti. Le piattaforme principali per costruire chatbot basate su questo canale sono Chatfuel, primo partner iniziale che offre la possibilità di costruire chatbot senza scrivere codice, e ManyChat.
- Telegram è un servizio di messaggistica istantanea basato su cloud ed erogato senza fini di lucro dalla società Telegram LLC che da Giugno 2015 ha introdotto la piattaforma per sviluppare bot. Attualmente viene utilizzato da 100 milioni di utenti.
- Slack offre degli strumenti di collaborazione aziendale utilizzabile da tutti i dispositivi mobili e da web browser, attualmente conta più di 5 milioni di utenti attivi al giorno, con un tasso di crescita del 3-5% a settimana. Lo strumento di sviluppo leader per la creazione di chatbot, app e integrazioni personalizzate per le principali piattaforme di messaggistica è Botkit.

Ovviamente esistono altri canali di distribuzione SMS, Skype, email, ma anche siti web. Tutte queste piattaforme di distribuzione sono state un'ottima rampa di lancio per l'evoluzione dei chatbot, è però possibile adottare un approccio diverso utilizzando le Conversational Interfaces (CI) che, tramite

un miglioramento dell'interfaccia utente e senza utilizzare canali di comunicazione di terze parti, permettono un'esperienza utente migliore, minori costi di manutenzione e tempi di valutazione più rapidi.

Gli ambienti nativi come Web o app permettono di avere il pieno controllo ed una migliore personalizzazione della conversazione contro la convenienza di una maggiore rivelabilità e distribuzione dei canali di messaggistica. Inoltre lavorando con API di terze parti c'è il rischio di generare troppa dipendenza dalla piattaforma e mettere a repentaglio la sostenibilità futura. Per le aziende in settori altamente regolamentati il problema della privacy è fondamentale. Gli utenti appartengono al proprietario della piattaforma e quindi si perde il controllo dei dati scambiati in un canale di messaggistica e nascono problemi di privacy. Questo è uno dei motivi principali per cui molte banche si stanno allontanando da Facebook Messenger e stanno costruendo la propria piattaforma Chatbot come Erica di Bank of America. Erica è un chatbot nativo in beta che si basa sulle conversational interfaces nel tentativo di ottenere un maggiore coinvolgimento dei giovani. Lo sviluppo nativo di chatbot è un argomento molto recente e quindi è possibile che in un futuro non molto lontano vengano creati esempi di comunicazioni cross-channel per usufruire al meglio di entrambi i benefici.

Una tipologia molto frequente di chatbot nativo è quello dei Website Chatbot che permettono di interagire con gli utenti tramite browser. Si possono individuare tre principali categorie:

- Live chat è uno dei tipi più frequenti a causa delle somiglianze visuali della finestra di chat utilizzata come canale di comunicazione con un'app di messaggistica.
- Banner è una variante della Live chat in cui la chat è inserita in un banner che è collocato in posizioni diverse della pagina web. Viene utilizzata come finestra per offrire promozioni o come guida interattiva per l'onboarding di un prodotto
- Sito web completo è una tipologia ambiziosa che sostituisce la pagina web in un chatbot per iniziare una conversazione con gli utenti. Un esempio di questo approccio si può vedere su www.landbot.io.

Le interfacce di conversazione focalizzano meglio l'attenzione dell'utente, hanno migliori capacità di personalizzazione e quindi possono essere utilizzate per ottenere nuovi clienti oppure per rendere l'assistenza dell'utente

più interattiva sostituendo le FAQ di un sito con un chatbot.

Un altro caso d'uso interessante è quello di ottenere feedback dagli utenti tramite conversazione e non con il solito form da compilare che spesso risulta lungo e noioso. Con la corretta progettazione del questionario, è possibile ottenere dati più affidabili ed utilizzabili, poiché l'esperienza di polling è interattiva, il che consente di evitare l'affaticamento del sondaggio, l'eccesso di velocità ed altri tipi di comportamento negativi dell'utente. Su un sito Web le interazioni dell'utente sono generalmente rese anonime, il che contribuisce a ridurre eventuali frizioni sulla rivelazione dell'identità dell'utente registrato sul canale di messaggistica.

A differenza dei canali di messaggistica in cui gli utenti sono attivi su base giornaliera, in un sito Web il tasso è inferiore. Il click-through rate(CTR), ovvero la misura dell'efficacia di una campagna pubblicitaria on-line rapportando il numero di visite al numero di click, è del 40% tramite notifiche push per le app di messaggistica. Questo tasso è inferiore per i chatbot su siti web, ma la differenza è ammortizzata da strumenti di retargeting e dalle notifiche del browser. Le interazioni O2O(Offline to Online) sono più accessibili con un'app di messaggistica di uso comune rispetto ad un sito web.

Quello della customer care rappresenta il settore in cui si sta sperimentando maggiormente l'utilizzo di chatbot ed intelligenza artificiale. Numerose aziende usano la piattaforma Messenger di Facebook per comunicare e rispondere alle richieste dei propri clienti attraverso i bot dedicati. Il vantaggio principale per loro è non dover più disporre di un'apposita applicazione e di forza lavoro dedicata a svolgere questo compito. Marketing Informatico ha lanciato sul mercato web il progetto Bot Facebook. Come gli altri chatbot di questo settore, è un risponditore automatico, attivo 24/7, che ha il compito e lo scopo di trasformare i visitatori di una pagina in clienti. Si tratta, quindi, di una soluzione software per le aziende che vogliono offrire una customer care ed una customer experience sempre attive e di qualità.

Come già scritto precedentemente, i chatbot sono ottimi strumenti per incrementare il custom care, e infatti è il settore in cui si sta espandendo maggiormente, ma non è l'unico dove possono essere utilizzati. Nel settore dell'informazione diverse testate giornalistiche, come CNN o TechCrunch, utilizzano chatbot per aggiornare gli utenti, i quali possono personalizzare le notizie da ricevere in base ai propri interessi.

Un settore con molto successo è quello dell'apprendimento per approfondire

le proprie conoscenze ed esercitarsi. StudyWithRobot, utilizzabile su Telegram, consente agli utenti di compiere numerose attività tra cui test del QI, perfezionamento del proprio inglese, incremento delle proprie conoscenze riguardo a materie umanistiche e scientifiche.

Un altro ambito inerente all'apprendimento è quello della traduzione. Questi tipi di chatbot funzionano sia da classici traduttori sia da traduttori come PronunciationBot che aiuta a capire l'esatta pronuncia dei termini in diverse lingue. Non mancano chatbot per le pubbliche amministrazioni nel rapportarsi con la cittadinanza. Il comune di Solarino, in Sicilia, offre un servizio di aiuto al cittadino nella richiesta di documenti dell'amministrazione tramite chatbot. Potenzialmente un chatbot potrebbe essere utilizzabile per altri scopi come aiutare un cittadino nel processo di raccolta differenziata.

Un chatbot può offrire il proprio servizio verso l'esterno, e quindi verso un potenziale cliente, oppure verso l'interno per l'azienda stessa. L'agenzia eFM [8] durante il ChatBot Day del 3 dicembre 2016 a Napoli ha mostrato come sfruttare i chatbot nei processi di smart working e flexible workplace. Un'azienda con l'utilizzo di chatbot potrebbe vendere prodotti e servizi, semplificare i pagamenti, aumentare la conoscenza sui propri clienti ed anche offrire ai clienti un'esperienza personalizzata di acquisto. Infatti secondo una ricerca effettuata da Oracle [1], entro il 2020 l'80% delle aziende offrirà ai clienti interazioni con chatbot e realtà virtuale.

Tramite un chatbot è possibile fare acquisti online e quindi un altro settore di interesse è quello degli e-commerce. Esistono chatbot che monitorano le promozioni dei negozi online come Pricetrackbot oppure chatbot che monitorano le spedizioni come Trackbot oppure ancora chatbot che accompagnano l'utente in tutte le fasi dell'acquisto impersonificando l'immagine di personal shopping assistant. Esiste poi Bot Facebook con il quale qualunque venditore in possesso di una pagina sul social ha la possibilità di vendere i propri prodotti agli utenti senza appoggiarsi ad un e-commerce. L'unico requisito è l'utilizzo di Paypal che garantisce la sicurezza della transazione e la rapidità dell'acquisto. Altro settore interessato dall'espansione dei chatbot è ovviamente quello della mobilità e del turismo. Bot Magellano è il primo chatbot ideato ed ottimizzato per il settore turistico ed alberghiero con il quale è possibile collezionare indirizzi Facebook attraverso la creazione di un database e semplificare l'interazione di un cliente con la necessità di raggiungere un luogo di interesse come un hotel o un ristorante. Infine,

ma non meno importanti, abbiamo il settore della gastronomia con chatbot per ricevere informazioni su ricette e cibi come Food Makers presente su Facebook Messenger ed il settore della salute con iCliniq per ricevere consulti medici su Telegram oppure HealthTap su Facebook Messenger. Uno dei bot più utilizzato, nell'ambito salute, in Italia è FarmacieBot che fornisce l'elenco delle farmacie vicine attraverso la geolocalizzazione dell'utente.

Tutti i chatbot elencati appartengono ad ambiti diversi, ma non sono pensati per scopi negativi. Esiste infatti un altro ambito in cui si sono sviluppati i chatbot ed è quello del malicious software. Ci sono bot in grado di distribuire dei malware infettando i dispositivi e la rete a cui questi sono connessi creando delle botnets. Esistono anche chatbot, denominati spambot, che inviano contenuti promozionali, raccolti dalla rete, per poi generare traffico verso un determinato sito. Infine ci sono anche gli impersonator bot che possono essere utilizzati per attacchi DDoS, ma anche per azioni di propaganda come è successo in Messico [18] in cui si è cercato di influenzare l'opinione pubblica tramite i bot di Twitter.

Capitolo 2

Requisiti

L'azienda Cosmogas attualmente offre un servizio di segnalazione dei difetti relativi ai prodotti venduti con relativa assistenza tecnica.

Un difetto consiste in un problema o un guasto in grado di compromettere il corretto funzionamento del prodotto. Soltanto un centro di assistenza tecnica, ovvero il clienti rivenditore e/o installatore e non l'utilizzatore finale, può individuare un difetto e di conseguenza segnalarlo all'azienda per richiedere una soluzione.

La segnalazione avviene tramite chiamata telefonica, nella quale una figura interna all'azienda riceve la chiamata da parte di un centro di assistenza tecnica ed incomincia ad elaborare l'intervento.

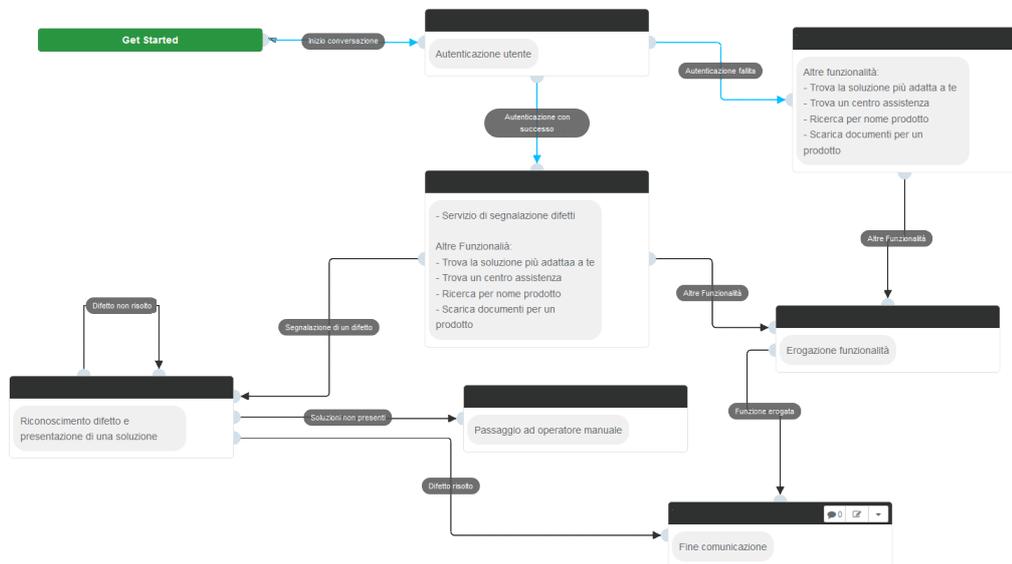
Un operatore raccoglie le informazioni importanti tramite una serie di domande mirate all'individuazione del difetto e del suo contesto. Le domande che vengono poste sono mirate ad individuare il prodotto (tramite richiesta di matricola o nome prodotto) e del problema (utilizzando anche l'esperienza degli operatori per non porre domande inutili). Quando l'operatore ha raccolto sufficienti informazioni, propone una o più soluzioni che possono essere applicate immediatamente oppure che richiedono più tempo, come ad esempio la sostituzione di un componente. In ogni caso l'intervento termina e l'operatore procede al salvataggio, su una base di dati, delle informazioni relative ad esso:

- dati relativi al centro di assistenza tecnico;
- codice e descrizione del difetto;
- data e descrizione della chiamata;

- soluzioni proposte;

Durante una giornata avvengono in media 50 interventi della durata di 10-15 minuti che vengono gestiti singolarmente. Quindi se un centro di assistenza cerca di chiamare mentre l'operatore è impegnato nella risoluzione di un intervento, questo viene messo in attesa oppure richiamato appena l'intervento in corso è stato risolto.

L'azienda ha estrapolato tutti i dati relativi agli interventi avvenuti in un file csv. Nell'ultimo periodo si è registrato un incremento significativo dei prodotti venduti proporzionale al numero di richieste di assistenza tecnica ricevute e quindi è nata l'esigenza di investire su questo servizio cercando di aumentarne efficacia ed efficienza.



L'azienda vuole un chatbot in grado di offrire il servizio di gestione delle segnalazioni dei difetti in maniera autonoma, rapida e senza limiti di richieste contemporanee. Il servizio deve essere immediato e disponibile 7 giorni su 7, 24 ore su 24, festivi compresi.

L'operatore dell'ufficio tecnico che attualmente gestisce gli interventi dovrebbe essere sgravato da questo servizio in modo da permettergli di utilizzare il proprio tempo lavorativo per altre attività.

Il servizio inoltre deve essere il più semplice possibile da utilizzare, con una curva di apprendimento bassa in modo tale da migliorare la user experience

globale.

Un utente centro di assistenza dovrebbe essere in grado quindi di utilizzare il servizio di segnalazione dopo un opportuna fase di autenticazione. Dopo essersi autenticato, l'utente può segnalare il difetto e il chatbot dovrebbe proporgli una soluzione e rimanere in attesa di un riscontro sull'effettiva risoluzione del problema. Se il difetto persiste allora deve continuare a proporre soluzioni valide per quel contesto. Solo quando il chatbot non ha più risposte valide da proporre allora dovrebbe passare l'intervento ad un operatore umano per non lasciare l'utente insoddisfatto. Se l'operatore è disponibile potrebbe aiutare subito l'utente tramite chat oppure tramite chiamata telefonica. In caso contrario, gli interventi sospesi devono essere salvati in modo tale che un operatore possa prenderli in carico manualmente.

Se invece il difetto viene risolto allora tutte le informazioni importanti dell'intervento devono essere salvate dal chatbot.

Tutti i dati memorizzati devono essere disponibili all'azienda per poter poter effettuare analisi.

L'azienda vuole valorizzare maggiormente il custom care, sia per i centri di assistenza tecnica e sia per gli utilizzatori finali dei prodotti, integrando all'interno del chatbot alcune funzionalità che offre attualmente sul proprio sito web:

- Una guida interattiva per visualizzare la soluzione adatta all'esigenza di ogni cliente rispondendo a poche domande. All'inizio è necessario selezionare la fonte di energia a disposizione e di seguito, tramite altre domande, vengono filtrati i prodotti in maniera coerente con quanto voluto dall'utente.
- Localizzazione su di una mappa dei centri assistenza tecnici più vicini sulla base della provincia e della tipologia di prodotto interessato. Le tipologie selezionabili sono riscaldamento e condizionamento. Oltre alla localizzazione viene visualizzata anche l'anagrafica base dei centri assistenza tecnici vicini.
- Ricerca e visualizzazione delle informazioni relative ad un determinato prodotto, inserendone il nome. Se il prodotto esiste allora viene visualizzata la scheda con tutte le caratteristiche tecniche del prodotto, le dimensioni, la descrizione, il capitolato, lo schema funzionale e

i componenti. Tutti questi documenti sono, oltre che visualizzabili, anche scaricabili in maniera gratuita.

Di seguito un'immagine esplicativa della funzionalità "Trova la soluzione più adatta a te".

Quale fonte energetica hai a disposizione?

- Gas/Metano**
 - Gas
 - Gas+Solare
 - Gas + Areotermico
 - Gas + Solare + Areotermico
 - Gas + Solare + Areotermico + Biomassa
- Solare termico**
 - Solare
 - Solare + Areotermico
 - Solare + Geotermico
 - Solare + Areotermico + Biomassa
- Areotermico**
 - Areotermico
- Geotermico**
 - Geotermico

Quale tipo di caldaia desideri?

- Scegli -
- Condensazione alta efficienza
- Tradizionale

Qual è la tua esigenza?

- Scegli -
- Riscaldamento
- Acqua calda
- Climatizzazione
- Riscaldamento + Acqua calda
- Riscaldamento + Climatizzazione
- Riscaldamento + Acqua Calda + Climatizzazione

AGUADENS
Scaldabagno murale a condensazione premiscelato ecologico
[SCOPRI](#)

AGUADENS
Scaldabagno murale a condensazione premiscelato ecologico
[SCOPRI](#)

AGUADENS
Scaldabagno murale a condensazione premiscelato ecologico
[SCOPRI](#)

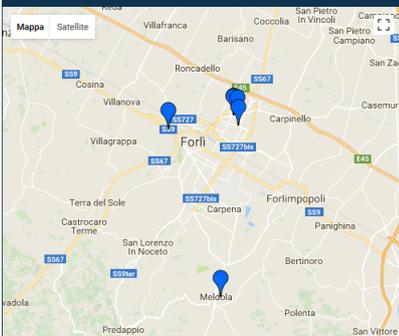
AGUADENS 180T, 210T, 280T
Produttori autonomi di acqua calda sanitaria a gas a condensazione modulare
[SCOPRI](#)

AGUADENS 180T, 210T, 280T
Produttori autonomi di acqua calda sanitaria a gas a condensazione modulare
[SCOPRI](#)

AGUADENS 180T, 210T, 280T
Produttori autonomi di acqua calda sanitaria a gas a condensazione modulare
[SCOPRI](#)

Di seguito un'immagine esplicativa della funzionalità "Trova un centro assistenza":

CENTRI ASSISTENZA AUTORIZZATI



C.A.B.E.C. SNC.DI MEZZANOTTE A. & C
 VIALE BOLOGNA, N. 88/A 47121 FORLÌ FC
 Tel. 0543 702150
 Fax. 0543 700163
 E-mail: c.a.b.e.c@libero.it

Cosmoclima di Maltoni Alfio e Cavalieri Michele snc
 Viale della Repubblica, 31/33 47014 Meldola FC
 Tel. 0543 494165
 Fax. 0543 494185
 Cell. 333 5410792
 E-mail: info@cosmoclima.it

FORLICLIMA SNC DI MORGAGNI & C.
 VIA CERVESE, 162/C 47122 FORLÌ FC
 Tel. 0543 722942
 Fax. 0543 758213
 Cell. 335 5615861
 E-mail: info@forliclima.com

Di seguito un'immagine esplicativa della funzionalità "Ricerca per nome prodotto" e "Scarica documenti per un prodotto".

MYDENS 115T CONDENSAZIONE ALTA POTENZA / MYDENS T / MYDENS 115T



Prestazioni eccezionali grazie al nuovo scambiatore al TITANIO

generatori di calore a condensazione modulari a basamento fino a 140 kW

I generatori di calore a condensazione Mydens 60T, 70T, 100T, 115T e 140T possono essere installati direttamente a pavimento. Sono potenti, di dimensioni contenute, leggeri e occupano poco spazio nelle centrali termiche. Mydens 60T e 70T sono dotati di uno scambiatore di calore C.R.V. in acciaio inox AISI 316 TI (Titanio). Mydens 100T, 115T e 140T sono dotati di 2 scambiatori di calore C.R.V. in acciaio inox AISI 316 TI (Titanio) che lavorano in cascata con potenze rispettivamente da 99 kW, 115 kW e 140 kW e in caso di malfunzionamento di uno scambiatore, l'altro funziona indipendentemente garantendo continuità di servizio. Gli scambiatori sono dotati ciascuno di centraline che li fanno dialogare fra di loro e li coordinano al fine di **garantire la massima efficienza, modulazione e un'usura uniforme**. Ogni scambiatore è dotato di tutti i dispositivi di controllo e sicurezza. Campo di modulazione fino a 1:10.

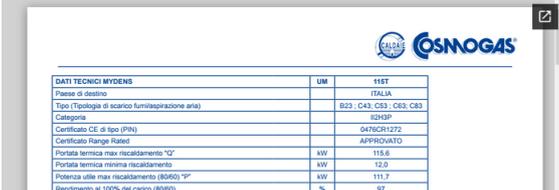
- Potenza - ▾

Caratteristiche tecniche DOWNLOAD PDF

Capitolato

Descrizione

Dimensioni



DATI TECNICI MYDENS	UM	115T
Paese di destino		ITALIA
Tipo (topologia di scarico fumisugrazione aria)		B23 - C43 - C53 - C63 - C83
Categoria		90/10P
Certificato CE di tipo (PIN)		0476CR1272
Certificato Range Rated		APPROVATO
Portata termica max riscaldamento °C	kW	115,8
Portata termica minima riscaldamento	kW	12,0
Potenza utile max riscaldamento (80/60) °C	kW	111,7
Rendimento al 100% del carico (80/60)	%	97

Capitolo 3

Analisi dei requisiti

L'affidabilità di un chatbot è inversamente proporzionale al quantitativo di cose che può fare. Per questo motivo si è scelto, in accordo con l'azienda, di considerare la prima parte relativa al sistema di segnalazione e tralasciare gli aspetti di integrazione delle funzionalità accessibili dal sito web.

Il primo passo essenziale è stato decidere cosa deve fare il chatbot sia per l'azienda e sia per gli utenti del servizio in modo tale da evitare di creare un sistema che nessuno vorrà utilizzare oppure un sistema di cui l'azienda non possa trarre vantaggio.

I principali obiettivi che si possono individuare dal punto di vista dell'azienda sono:

- Incrementare la qualità del customer service. Nel caso specifico offrendo un sistema di assistenza tecnica disponibile sempre e in grado di trovare una soluzione più velocemente di un operatore umano.
- Ridurre il costo del customer service, riducendo l'intervento umano e riallocando le risorse in altre attività.

Gli obiettivi dell'utente finale da soddisfare invece sono:

- Accesso facilitato alle informazioni.
- Supporto a assistenza tecnica immediata ed efficiente.

Quando gli obiettivi sono stati decisi è necessario analizzare l'estensione del dominio della conversazione tra utente e chatbot. In un dominio aperto l'utente può svolgere una conversazione senza necessariamente un obiettivo

ben definito. L'infinito numero di argomenti e il fatto che sia necessaria una certa conoscenza del mondo per creare risposte ragionevoli rendono questa scelta abbastanza problematica.

Si è quindi scelto di modellare il chatbot in un dominio chiuso. Lo spazio di possibili input e output è limitato, ma questo non risulta un problema per il nostro sistema perché esso cerca di raggiungere un obiettivo molto specifico.

Il sistema conversazionale deve essere soprattutto in grado di comprendere l'input dell'utente ed estrapolare da esso le informazioni importanti.

Per raggiungere l'obiettivo finale il chatbot deve essere modellato in modo tale che la conversazione sia a lunga durata, ponendo domande inerenti e tenendo traccia delle risposte ottenute. Bisogna tenere conto del fatto che più la conversazione è lunga e più l'automatizzazione risulta complessa, ma non è possibile pensare di modellare la conversazione in maniera breve con una risposta ad ogni richiesta ricevuta.

Infine è essenziale che il chatbot sia in grado di salvare le informazioni relative all'intervento avvenuto sia che sia stato risolto sia che sia in stato di attesa di essere risolto.

3.1 Glossario

In questo glossario vengono spiegati alcuni termini dei requisiti che potrebbero essere ambigui visto che l'interpretazione del linguaggio naturale è limitato alla propria conoscenza base.

- **Chatbot** è un sistema software in grado di estrapolare, da un input in linguaggio naturale, una richiesta e soddisfarla.
- **Difetto** è un guasto o un problema che deve essere segnalato per poter ottenere assistenza tecnica in merito al difetto.
- **Centro di assistenza tecnica** è l'utente che usufruisce del servizio di segnalazione e assistenza tecnica.
- **Segnalazione** è la descrizione del difetto da parte di un centro di assistenza tecnico all'azienda.

- **Intervento** è la risoluzione di una segnalazione da parte dell'operatore. Può essere in fase di risoluzione mentre l'operatore interagisce con il centro di assistenza tecnica oppure risolto se la risoluzione è stata proposta.
- **Soluzione** è un'azione in grado di risolvere un difetto. Viene generata dall'operatore umano dopo aver raccolto informazioni sul difetto segnalato dal centro di assistenza tecnica.
- **Operatore** è un utente umano che gestisce manualmente le segnalazioni da parte dei centri di assistenza, cercando una soluzione al difetto proposto.

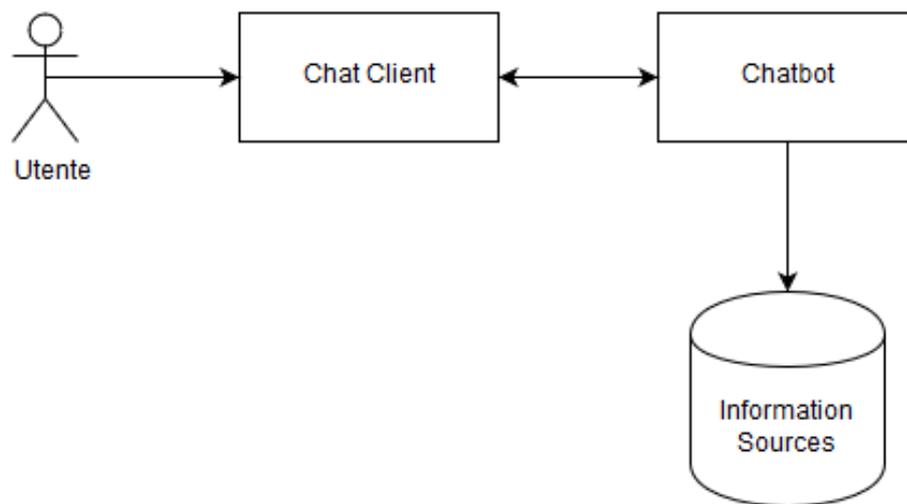
3.2 User Stories

Per cercare di definire cosa il sistema dovrà fare e per evitare di scrivere specifiche ambigue verranno proposte di seguito le user stories. Ogni user stories si focalizza su un aspetto importante che il sistema deve soddisfare per un determinato soggetto.

- Come utente azienda voglio visualizzare i dati relativi agli interventi così che possa fare analisi statistica.
- Come utente azienda voglio visualizzare gli interventi in sospeso così che possa cercare di risolverli.
- Come utente centro di assistenza tecnica quando ho bisogno di assistenza per un difetto di cui non conosco risoluzione voglio descrivere il difetto al chatbot così che esso possa offrirmi la soluzione oppure chiedermi altre informazioni per raggiungere la soluzione.
 - Verificare che l'utente possa accedere a tale servizio.
- Come utente centro di assistenza tecnica voglio essere ricontattato da un utente umano se il chatbot non è riuscito a trovare una soluzione oppure se l'ho richiesto esplicitamente così che riesca a risolvere il difetto
 - Verificare che l'utente possa accedere a tale servizio.

3.3 Modello del dominio

L'interazione avviene tramite un Chat cliente, ovvero un'interfaccia con la quale generare un input che viene elaborato dal chatbot e scatena un'azione con la quale viene generata una risposta in tempo reale con una sorgente di informazioni predisposte. La risposta viene mostrata sul Chat client che permette all'utente di continuare oppure terminare la conversazione iniziata.



Come risultato dell'analisi effettuata il sistema è composto da due componenti principali da progettare:

- **Chat Client**
 - **Struttura:** Il chat client è un'entità composta.
 - **Interazione:** Il chat client interagisce, come strumento di input, con il componente chatbot e come strumento di output, per mostrare le risposte elaborate.
 - **Comportamento:** Il chat client è composto da oggetti attivi che cambiano di stato sotto il controllo di un operatore umano. Permette anche la visualizzazione dell'output del componente chatbot.

- **Chatbot**

- **Struttura:** Il chatbot è un'entità composta proattiva.
- **Interazione:** Il chatbot riceve l'input dal Chat client e gli invia l'output.
- **Comportamento:** Il chatbot è un'entità in grado di mantenere una conversazione nella quale elaborare i messaggi in linguaggio naturale e generare dei messaggi coerente con l'input ricevuto. Quando la conversazione è terminata deve salvare i dati relativi all'intervento.

Capitolo 4

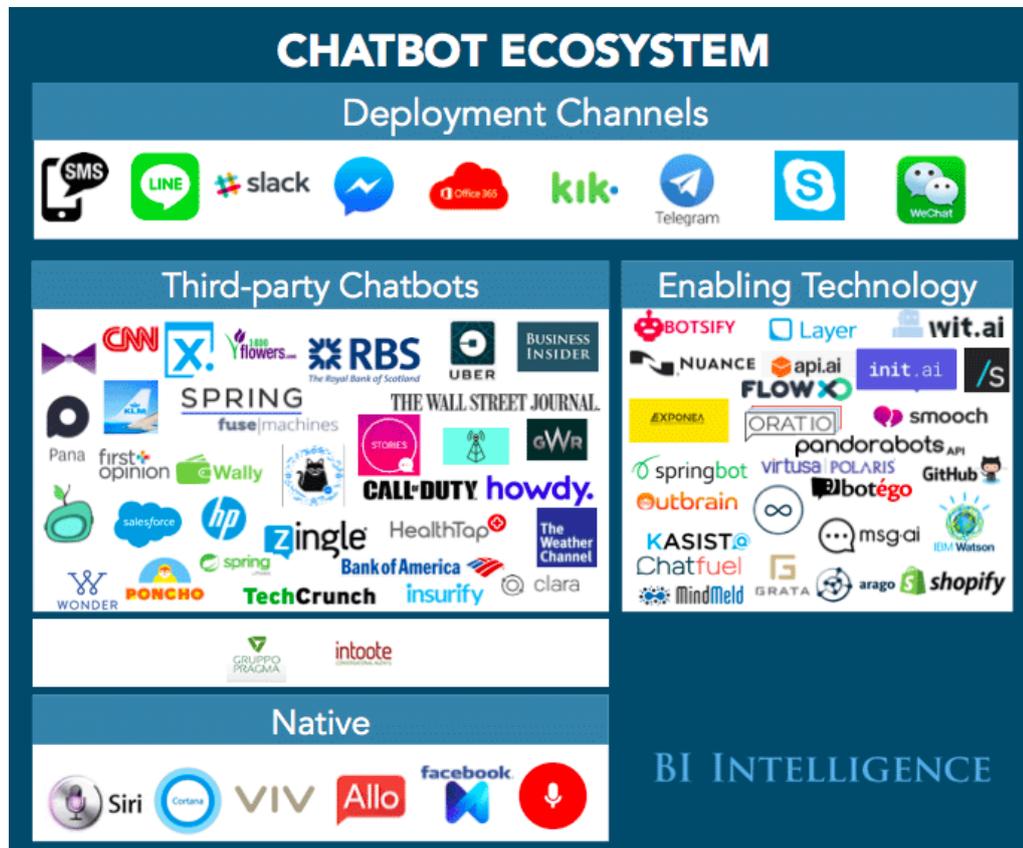
Progetto

La progettazione di un chatbot deve essere effettuata in modo tale da poter sviluppare tutte e tre le dimensioni principali dell'innovazione. Il chatbot deve essere desiderabile, ovvero deve essere uno strumento che un utente vorrebbe utilizzare. Non ha alcun senso progettare qualcosa che nessuno vorrebbe mai provare.

Il chatbot deve essere anche essenziale, è necessario cercare di ignorare aspetti non importanti o secondari rispetto agli obiettivi decisi in fase di analisi. Troppi dettagli inutili possono portare ad una perdita del focus ed al fallimento del progetto.

Infine bisogna utilizzare delle tecnologie in grado di soddisfare le specifiche e quindi il chatbot che si cerca di realizzare deve essere fattibile.

Questi tre criteri riguardanti il design, il business e la tecnologia devono essere alla base delle decisioni che vengono prese per realizzare un prodotto efficiente.



L'immagine rappresenta l'attuale ecosistema dei chatbot rappresentato da BI INTELLIGENCE [7]. Nella parte superiore possiamo osservare i canali di distribuzione che rappresentano i luoghi in cui i chatbot vivono e interagiscono con gli utenti come app di messaggistica, reti cloud, SMS e client di posta elettronica.

Il riquadro a sinistra mostra i principali chatbot di terze parti che un utente può utilizzare mentre alla sua destra possiamo osservare le aziende che offrono degli strumenti per creare chatbot tramite tecnologie di AI. Nella parte inferiore invece ci sono i bot costruiti dalla piattaforma o app in cui stanno operando.

Sempre più aziende importanti continuano ad annunciare il loro supporto per i chatbot all'interno della propria attività, come LinkedIn, Starbucks e eBay. I principali attori del settore chatbot, come Facebook, Google, Microsoft e Telegram stanno investendo nello sviluppo delle tecnologie per

chatbot.

Questo ecosistema è in continua espansione come possiamo vedere dalla seguente immagine di KeyReply [6] che offre una panoramica completa sulla situazione attuale ed offre aiuto a comprendere e scegliere le strategie di sviluppo di chatbot più appropriate incrociando le esigenze e le capacità del mercato.



L'asse orizzontale varia da funzione "Marketing" a funzione "Support". La prima funzione si riferisce alla capacità di un chatbot di valorizzare il marchio o il prodotto per i potenziali e attuali clienti mentre la seconda funzione si riferisce alla capacità di un bot di assistere gli attuali clienti con problemi e di risolverli.

L'asse verticale invece varia da "Managed", riferito alle società che danno in outsourcing lo sviluppo di bot, a "Self-serve", aziende che costruiscono i bot internamente con strumenti esistenti eventualmente.

Dall'interno verso l'esterno i cerchi concentrici rappresentano innanzitutto le piattaforme di messaggistica che consentono l'utilizzo dei chatbot trami-

te API o framework. Subito attorno possiamo vedere i marchi di aziende che hanno sperimentato e rilasciato bot e allontanandoci ancora di più dal nucleo i Providers, ovvero le aziende che possiedono le capacità di offrire ottime prestazioni per il proprio quadrante.

Infine abbiamo gli strumenti di supporto utilizzati nei cerchi interni per sviluppare chatbot.

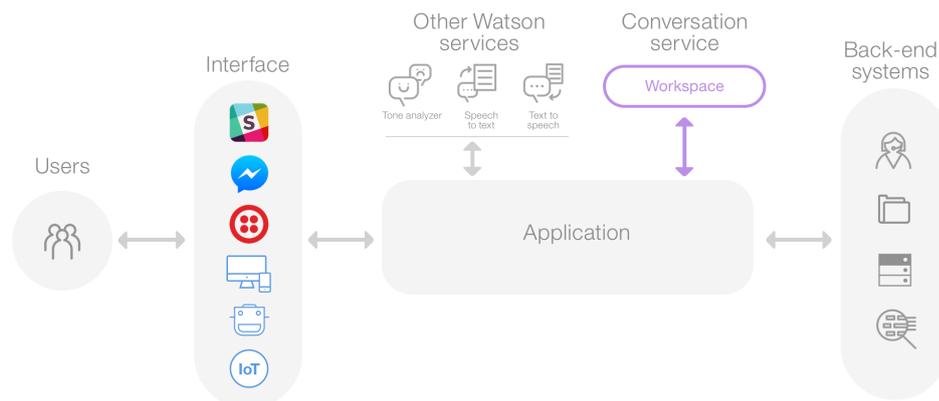
Dall'analisi dei requisiti è emerso che l'obiettivo principale è quello di offrire un servizio di assistenza e supporto al cliente e quindi, se viene preso come riferimento l'ecosistema appena presentato, è necessario focalizzarsi sul lato destro.

Chiaramente la scelta del quadrante risulta semplice poiché privilegiando il lato "Self-service" è possibile innanzitutto risparmiare denaro, ma soprattutto è possibile avere un controllo maggiore sullo strumento.

Ristretto il numero di opzioni disponibili in questo denso panorama, per la progettazione del sistema, si è giunti alla decisione di sperimentare i servizi di IBM Watson Conversation, in quanto le sue funzionalità soddisfano tutte le specifiche.

4.1 IBM Watson Conversation

Con il servizio IBM Watson Conversation è possibile creare un sistema in grado di comprendere l'input in linguaggio naturale che utilizza algoritmi di machine learning per rispondere ai clienti in modo da simulare una conversazione tra gli esseri umani.



Gli utenti interagiscono con l'applicazione attraverso un'interfaccia utente che deve essere implementata oppure tramite un sistema di messaggistica già esistente.

L'applicazione, da implementare, all'inizio si connette ad uno spazio di lavoro che contiene il flusso di dialogo e i dati di addestramento. Successivamente il servizio Watson interpreta l'input ricevuto dall'utente, dirige il flusso della conversazione e raccoglie le informazioni di cui ha bisogno.

L'applicazione offre la possibilità di interagire con sistemi di back-end in base all'intento dell'utente ed ad ulteriori informazioni. L'elaborazione del linguaggio naturale per il servizio Conversation avviene all'interno di uno spazio di lavoro, che può essere visto come un contenitore di tutte le risorse che definiscono il flusso di conversazione per un'applicazione.

Una singola istanza del servizio Conversation può contenere più spazi di lavoro. Uno spazio di lavoro contiene i seguenti tipi di risorse:

- **Intents** oppure **intenti** rappresentano lo scopo dell'input di un utente. Si possono definire intenti per ogni tipo di richiesta utente che si vuole supportare. Per addestrare lo spazio di lavoro a riconoscere gli intenti è opportuno fornire una quantità elevata di esempi.
- **Entities** oppure **entità** rappresentano dei termini rilevanti per fornire un contesto specifico ad un intento. Per addestrare lo spazio di lavoro a riconoscere le entità è necessario elencare i valori possibili per ogni entità ed i sinonimi che gli utenti potrebbero immettere.
- **Dialogo** è un flusso di conversazione ramificato che definisce come l'applicazione risponde quando riconosce gli intenti e le entità che sono state precedentemente definite.

Il sistema si addestra automaticamente ogni volta che viene aggiunta una delle risorse, ma è possibile effettuare anche un addestramento manuale simulando delle conversazioni e modificando le risorse che il sistema riconosce. Il numero di spazi di lavoro che si possono creare in una singola istanza del servizio dipende dal piano di servizio Conversation. Equivale a 5 spazi lavoro per il piano Lite e a 20 spazi lavoro per il piano Standard/Premium. Il servizio Watson offre il supporto alla lingua italiana in maniera completa.

4.2 Configurazione iniziale

Per iniziare è essersi registrati al servizio Conversation con un account IBM Cloud. La registrazione è gratuita ed è si può effettuare a questo link <https://console.bluemix.net/catalog/services/conversation/> scegliendo il piano che si preferisce.

Per la progettazione del prototipo di chatbot è stato scelto il piano gratuito Lite con delle limitazioni sul numero di spazi di lavoro, intenti, entità e chiamate API. Queste limitazioni non sono un problema perché, essendo un prototipo, il numero di risorse sono sufficienti per lo sviluppo.

In ogni momento è possibile aggiornare il piano di lavoro alla versione Standard a pagamento eliminando le limitazioni sulle chiamate API e aumentando in maniera elevata i limiti sulle altre risorse oppure aggiornare alla versione Premium per le aziende con ulteriori migliorie.

Una volta che è stato effettuato l'accesso al servizio è necessario creare uno spazio di lavoro aggiungendo nome, descrizione e lingua con la quale il servizio funzionerà. Il servizio non può funzionare multilingua quindi è importante scegliere il valore migliore. Nello sviluppo del prototipo, la clientela è interamente italiana e quindi la scelta della lingua è stata piuttosto semplice ed immediata.



Create a workspace

Workspaces enable you to maintain separate intents, user examples, entities, and dialogs for each use or application.

Name
CosmoBot

Description
Prototype of a chatbot for technical assistance to customers

Language
Italian 



4.3 Intents

Gli Intents [11] sono scopi o obiettivi espressi nell'input di un cliente e possono essere suddivisi in due categorie:

- Casual Intents ovvero il tentativo di condurre una conversazione senza un vero e proprio obiettivo. Vengono utilizzate principalmente per iniziare o terminare la conversazione oppure come congiunzione tra gli Intents di tipo Business.
- Business Intents si mappano direttamente con gli obiettivi del chatbot. Idealmente il focus dovrebbe essere maggiore su questo tipo perché sono molto differenti in base allo scenario e sono necessari per massimizzare la user experience. Non va però trascurata l'altra categoria in quanto permettono di rendere la conversazione più simile ad una vera conversazione

Il servizio IBM Watson Conversation, mediante l'elaborazione del linguaggio naturale, riconosce l'intento espresso nell'input di un cliente e di conseguenza sceglie il flusso di dialogo corretto per proporre una risposta.

Per pianificare gli intents del prototipo bisogna considerare cosa potrebbero voler fare gli utenti e cosa deve essere in grado di gestire il chatbot. Scegliere correttamente gli intents è il primo passo per fornire una risposta valida e per iniziare a definire i flussi di dialogo.

All'inizio sono stati raccolti e raggruppati in categorie il maggior numero possibile di domande, comandi o altri possibili input dei clienti in modo tale da avere una visione migliore. Infatti è molto importante ricordarsi che la stessa richiesta può essere espressa in maniera molto diversa ed è altrettanto importante che il sistema riesca a riconoscerla.

Per ogni raggruppamento è stato creato un intent che a sua volta è stato popolato con gli esempi di input con i quali il sistema IBM si addestra.

Il numero massimo di intents ed esempi che si possono creare sono 25 e dipendono dal piano di servizio Conversation scelto. Per la progettazione di questo prototipo è stato deciso che sono più che sufficienti.

Di seguito verranno mostrati nel dettaglio gli intent creati.

4.3.1 Welcome

Intent name
#Welcome

Description
Start of the conversation

Add user examples
Add user examples to this intent

[Add example](#)

User examples (5) ▼

Buonasera

Buongiorno

Ciao

Salve

Weila

4.3.2 RicercaSoluzione

Intent name
#RicercaSoluzione

Description
The user try to ask for help

Add user examples
Add user examples to this intent

[Add example](#)

User examples (10) ▼

aiutami

aiuto

Come devo fare per avere assistenza

come posso fare per avere assistenza

Devo risolvere un problema

Ho bisogno di aiuto

Ho un problema

Mi puoi aiutare?

Mi serve una mano

vorrei segnalare un problema

4.3.3 ConfermaSoluzione

Intent name
#ConfermaSoluzione|

Description
The user has solved his problem

Add user examples
Add user examples to this intent

[Add example](#)

User examples (10) ▼

certo

funziona ora

grazie

Grazie funziona!

Non ho più il problema

ok

perfetto

si

Sì, ora funziona

sono a posto così

4.3.4 SoluzioneNonFunzionante

Intent name
#SoluzioneNonFunzionante

Description
The user has NOT solved his problem

Add user examples
Add user examples to this intent

[Add example](#)

User examples (10) ▼

altre soluzioni

dammi un'altra soluzione

ho ancora il problema

Il problema persiste

mi serve ancora aiuto

no

non funziona

non ho risolto

non va

Non va bene

4.4 Entity

Ogni volta che si progettano gli intents, anche le entità devono essere identificate ed etichettate di conseguenza.

Una entità rappresenta un termine o un oggetto nell'input dell'utente che fornisce chiarimenti o specifici contesti per un particolare intento. Se gli intents possono essere visti come i verbi, ovvero qualcosa che un utente vuole fare, le entità possono essere viste come i nomi, permettendo di rappresentare più azioni specifiche in un unico intents.

Le entità rappresentano una classe di oggetto o di un tipo di dati rilevante per uno scopo dell'utente. Riconoscendo le entità che sono indicate nell'input dell'utente, il servizio Conversation può scegliere le azioni specifiche da intraprendere.

Il servizio Conversation fornisce una serie di entità di sistema da utilizzare per qualsiasi applicazione. L'abilitazione di un'entità di sistema consente di popolare rapidamente lo spazio di lavoro con i dati di addestramento comuni a molti casi di utilizzo. Le entità di sistema disponibili sono:

- @sys-currency rileva i valori monetari espressi in un'espressione con un simbolo di valuta o termini specifici della valuta e restituisce un valore numerico.
- @sys-date restituisce una stringa nel formato "aaaa-mm-gg" a partire da una rappresentazione di data come anche i giorni della settimana.
- @sys-time restituisce una stringa nel formato "hh:mm:ss" a partire da una rappresentazione di orario.
- @sys-location estrae i nomi dei luoghi.
- @sys-number rileva i numeri scritti in numero o in parole e restituisce un valore numerico.
- @sys-percentage rileva le percentuali indicate in un'espressione con il simbolo della percentuale o scritte usando le parole e restituisce un valore numerico.
- @sys-person permette di estrarre i nomi.

Per lo sviluppo del prototipo non è stato necessario abilitare le entità di sistema.

Il numero di entità, valori di entità e sinonimi che si possono creare dipende dal piano di servizio Conversation scelto ed equivale a 25 entità per spazio lavoro e 100.000 valori e sinonimi di entità per spazio di lavoro.

La modalità "Fuzzy" offre delle funzionalità per migliorare le capacità del chatbot di riconoscere i termini dell'input utente con una sintassi simile all'entità, ma senza richiedere una corrispondenza esatta. In particolare cerca di riconoscere la forma della radice dei valori di entità che hanno varie forme grammaticali oppure tenta di associare l'input utente all'entità corrispondente adeguata nonostante la presenza di errori di ortografia o lievi differenze sintattiche. Questa modalità offre anche la possibilità di suggerire automaticamente dei sinonimi, basati sulla sotto-stringa, presenti nelle entità definite dall'utente ed associando un punteggio di affidabilità inferiore rispetto alla corrispondenza esatta dell'entità.

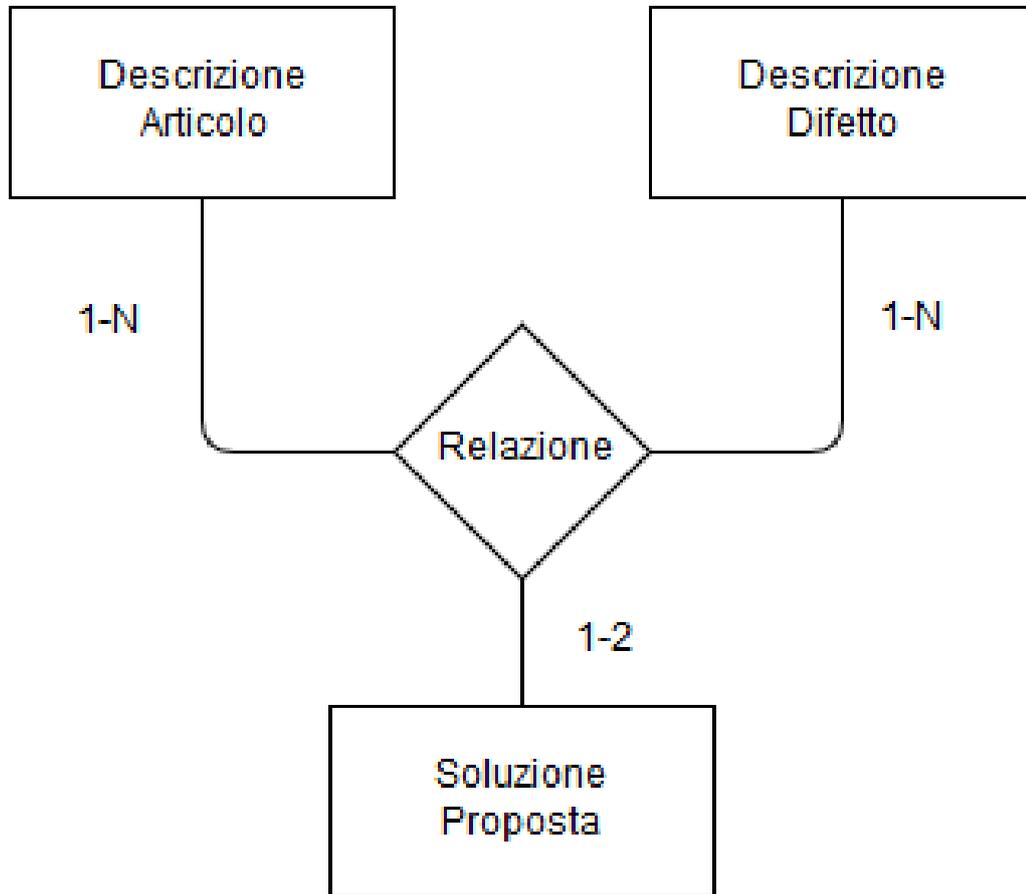
La modalità "Fuzzy" è attualmente in beta per la lingua italiana ed è stato pertanto scelto di non abilitarla per lo sviluppo del prototipo. Nessun particolare vincolo vieta di attivare questa modalità in futuro quando sarà terminata la fase di beta.

Nella fase di definizione delle entità come primo passo si sceglie il nome dell'entity. Di seguito si aggiungono i valori e relativi sinonimi o modelli per aiutare il chatbot ad imparare e capire i dettagli importanti che gli utenti possono esprimere.

Per definire le entità del sistema si è partiti dai dati estratti dall'azienda che rappresentano tutti gli interventi registrati. Il file in questione presenta 58 colonne e 1305 righe, quindi si è resa opportuna una prima fase di bonifica per eliminare i dati non importanti. Per lo sviluppo del prototipo gli unici dati interessanti alla fine della pulizia sono:

- Descrizione articolo, ovvero il prodotto che presenta un difetto da segnalare per avere assistenza.
- Descrizione difetto, il problema per il quale si vuole ricevere assistenza.
- Prima soluzione proposta.
- Seconda soluzione proposta.

I 1305 record così ripuliti possono essere rappresentati con il seguente modello ER:



Avendo un numero elevato di entità è possibile importarle tramite un file CSV anziché definirle una ad una nello strumento Conversation. Il formato è "`<entity>, <value>, <synonyms>`" dove `<entity>` è il nome di un'entità, `<value>` è il valore dell'entità e `<synonyms>` è un elenco di sinonimi separati da virgole per questo valore.

I dati sono stati successivamente importati in un database di supporto al servizio per recuperare le soluzioni relative al prodotto e al difetto riconosciuto. Il Database management system utilizzato è la versione 10.1.30 di MariaDB che viene utilizzata anche dall'applicazione con la quale si può interagire con il chatbot.

Ogni riga della tabella "solutions" contiene un prodotto, un difetto, una soluzione ed un indice di priorità. Per lo stesso prodotto e difetto possono esserci più soluzioni e di conseguenza l'indice di priorità viene utilizzato per ordinarle dalla più utile a quella meno utile in base alle soluzioni passate ed alle interazioni con il chatbot future.

4.4.1 Prodotto

Partendo dai dati bonificati, sono stati estratti i dati relativi alla "Descrizione Articolo" e formattati in un file csv da importare nel sistema Conversation.

Dopo che il sistema termina l'apprendimento sui dati caricati sarà in grado di riconoscere i 245 tipi diversi di entità "Prodotti". Per rendere il sistema più efficiente sarebbe necessaria una fase di controllo sui dati caricati con aggiunta di ulteriori sinonimi.

Entity name
@Prodotto

Value name
Enter value

Synonyms

Synonyms ▼ Enter synonym +

Add value

Entity values (245) ▼ **Type**

<input type="checkbox"/> AGUA SPLIT AC2 CON TELE...	Synonyms	AGUA SPLIT AC2 CON TELECOM.
<input type="checkbox"/> AGUA SPLIT AS2 CON TELE...	Synonyms	AGUA SPLIT AS2 CON TELECOM.
<input type="checkbox"/> AGUADENS 16-0000	Synonyms	AGUADENS 16-0000
<input type="checkbox"/> AGUADENS 22-0000	Synonyms	AGUADENS 22-0000
<input type="checkbox"/> AGUADENS 27-0000	Synonyms	AGUADENS 27-0000
<input type="checkbox"/> AGUADENS 37-0000	Synonyms	AGUADENS 37-0000
<input type="checkbox"/> AGUADENS 37-0224 TSP	Synonyms	AGUADENS 37-0224 TSP
<input type="checkbox"/> AXIA 20 -253 MEX	Synonyms	AXIA 20 -253 MEX
<input type="checkbox"/> AXIA 25 -275 SC	Synonyms	AXIA 25 -275 SC
<input type="checkbox"/> AXIA 29	Synonyms	AXIA 29

4.4.2 Difetto

Partendo dai dati bonificati, sono stati estratti i dati relativi alla "Descrizione Difetto" e formattati in un file csv da importare nel sistema Conversation. Dopo che il sistema termina l'apprendimento sui dati caricati sarà in grado di riconoscere i 179 tipi diversi di entità "Difetto".

Per questa entità in particolare, una fase di aggiornamento manuale, si rende obbligatoria in quanto il difetto potrebbe essere descritto in maniera molta

differenti e quindi i sinonimi importati andrebbero analizzati ed aggiornati singolarmente.

Entity name
@Difetto

Value name
Enter value

Synonyms ▼ **Synonyms**
Enter synonym +

Add value

Entity values (179) ▼ **Type**

		Synonyms	
<input type="checkbox"/>	60504187 CAVO VALVOLA ...	Synonyms	60504187 CAVO VALVOLA DEVCAVO INVERTITO
<input type="checkbox"/>	60505022 CAND ACC SX C...	Synonyms	60505022 CAND ACC SX CONDCERAMICA ROTTA
<input type="checkbox"/>	60505022 CAND ACC SX C...	Synonyms	60505022 CAND ACC SX CONDPIEGATA
<input type="checkbox"/>	60505023 CAND ACC DX C...	Synonyms	60505023 CAND ACC DX CONDPIEGATA
<input type="checkbox"/>	60505024 CAND IONIZ CO...	Synonyms	60505024 CAND IONIZ COND PIEGATA
<input type="checkbox"/>	60507034 SCHED PWM AX...	Synonyms	60507034 SCHED PWM AXIA-ESALTA IL FUSIBILE
<input type="checkbox"/>	60510021 ACCENDITORE ...	Synonyms	60510021 ACCENDITORE CONDIN CORTOCIRCUITO
<input type="checkbox"/>	60510021 ACCENDITORE ...	Synonyms	60510021 ACCENDITORE CONDNON SCOCCANO LE SCINTILLE
<input type="checkbox"/>	61201029 CENTRLINA AXI...	Synonyms	61201029 CENTRLINA AXIA ENON ALIMENTA LA SCHED PWM
<input type="checkbox"/>	61201029 CENTRLINA AXI...	Synonyms	61201029 CENTRLINA AXIA ENON ALIMENTA LA V.GAS
<input type="checkbox"/>	61201029 CENTRLINA AXI...	Synonyms	61201029 CENTRLINA AXIA ERESTA IN BLOCCO
<input type="checkbox"/>	61201029 CENTRLINA AXI...	Synonyms	61201029 CENTRLINA AXIA ERESTA IN VENTILAZIONE

4.5 Dialog

Definendo gli intents e le entities è stato definito cosa il nostro sistema deve catturare e comprendere. La fase successiva è quella di definire il flusso del dialogo.

Un albero di conversazione è uno strumento che permette di tracciare i percorsi che gli utenti percorrono in una conversazione. Sono equivalenti ai

diagrammi di flusso della programmazione, mappando ogni fase del percorso dell'utente dal messaggio di benvenuto all'output finale. Quando un albero diventa complesso, questi flussi possono diventare documenti ingombranti, ma sono estremamente utili per comunicare ad altri esattamente come funzionerà il chatbot, molto prima che ci sia un prototipo.

Modellare [10] un dialogo vuol dire utilizzare gli intent e le entità che vengono identificati nell'input dell'utente, oltre al contesto ottenuto dall'applicazione, per interagire con l'utente e infine fornire una risposta utile.

Il dialogo, con il sistema Conversation, viene rappresentato graficamente come una struttura ad albero di cui ogni nodo contiene, come minimo, una condizione ed una risposta.

Le condizioni specificano le informazioni che devono essere presenti nell'input utente affinché questo nodo nel dialogo venga utilizzato. Le condizioni all'interno di un nodo vengono valutate in ordine quindi è importante assicurarsi che l'ordine sia corretto.

If bot recognizes:

Enter an intent, entity or context variable...

Filter by:

- # intents
- @ entities
- \$ context variables

Common conditions:

- welcome Triggers when a conversation is started by the system
- anything_else Triggers when the user input does not match

Le informazioni potrebbero essere:

- un valore di variabile di contesto. Il dialogo è privo di stato, il che significa che non conserva le informazioni tra uno scambio di messaggi e l'altro quindi lo strumento IBM Watson offre la possibilità di salvare le informazioni importanti in variabili di contesto. La condizione viene specificata con questa la sintassi \$variable_name:value oppure \$variable_name == 'value'. Inoltre permette di specificare il contesto della

conversazione, infatti l'input ricevuto non è sufficiente per fornire una risposta corretta all'utente.

- un' entità utilizzando la sintassi @entity_name oppure un valore di entità utilizzando la sintassi @entity_name:value
- un intento specifico utilizzando la sintassi #intent_name

Inoltre ci sono condizioni speciali per eseguire funzioni di dialogo comuni:

- anything_else da utilizzare quando l'input utente non corrisponde a nessun altro nodo di dialogo. Viene utilizzata per catturare tutte le situazioni non gestite esplicitamente.
- conversation_start viene valutata come true durante il primo turno di dialogo a prescindere che la richiesta iniziale dall'applicazione contenga o meno l'input utente. Un nodo con la condizione conversation_start può essere utilizzato per inizializzare le variabili di contesto o per eseguire altre attività all'inizio del dialogo.
- false viene sempre valutata come false e di conseguenza non è possibile entrare nel nodo. Viene utilizzata soprattutto in fase di sviluppo per evitare che un utente acceda ad un nodo ancora non terminato.
- true viene sempre valutata come true. Viene utilizzata alla fine di un elenco di nodi o risposte per recuperare eventuali risposte che non corrispondono a nessuna delle condizioni precedenti
- irrelevant viene valutata come true se l'input dell'utente viene considerato irrilevante dal servizio di conversazione.
- welcome viene valutata come true durante il primo turno di dialogo, solo se la richiesta iniziale dall'applicazione non contiene alcun input utente

Then respond with:

	If bot recognizes	Respond with		
1	Enter an intent, entity or context vari	Enter a response...	⚙️	🗑️
2	Enter an intent, entity or context vari	Enter a response...	⚙️	🗑️

⊕ Add response

La risposta invece è l'espressione utilizzata dal servizio per rispondere all'utente e può essere definita come:

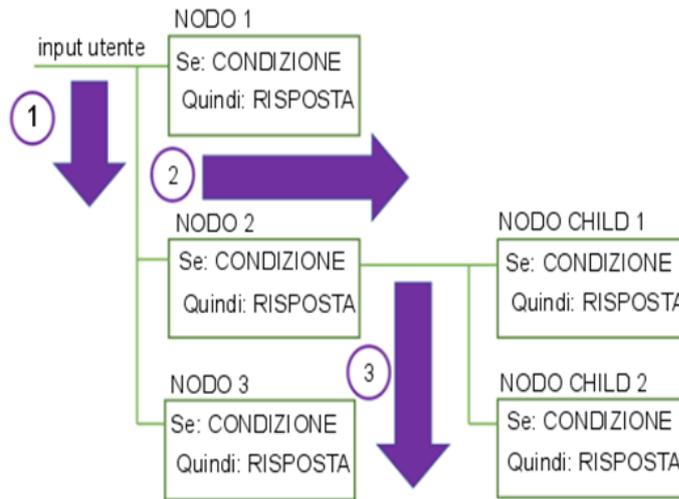
- una risposta di testo semplice, immettendo direttamente il testo che il servizio visualizzerà all'utente. Per evitare risposte sempre uguali che potrebbero influenzare la user experience della conversazione è possibile inserire delle variazioni della risposta in modo tale da proporre risposte diverse, scelte in maniera sequenziale o casuale, allo stesso nodo.
- una risposta multipla condizionata che offre la possibilità di avere risposte diverse, attivate da diverse condizioni, ma sempre nello stesso nodo. Infatti per attivare il nodo è necessario che venga validata la sua condizione e solo dopo avviene l'elaborazione delle condizioni per generare la risposta. Questo tipo di risposta permette di risolvere più scenari con un unico nodo.
- una risposta complessa utilizzando l'editor JSON per modificare la proprietà "output":. Per implementare un comportamento più complesso è possibile includere le seguenti proprietà:
 - values è un array di stringhe JSON che contiene più versioni del testo di output che può essere restituito da questo nodo di dialogo. L'ordine in cui vengono restituiti i valori nell'array dipende dall'attributo selection_policy.
 - selection_policy indica il modo in cui le risposte vengono recuperate e può essere in maniera sequenziale oppure casuale
 - append per specificare se aggiungere un valore a un array o sovrascrivere i valori nell'array con uno o più nuovi valori.

Dopo che è stata generata la risposta è necessario definire le operazioni successive per proseguire lo scorrimento dell'albero.

La prima opzione è quella di attendere l'input da parte dell'utente e finché questo non arriva il sistema rimarrà in attesa. L'altra opzione è quella di passare ad un altro nodo per ignorare l'attesa dell'input utente e spostare la conversazione direttamente al nodo figlio oppure ad un nodo completamente diverso. Oltre a scegliere il nodo di destinazione è necessario specificare se l'azione è destinata alla risposta o alla condizione del nodo di dialogo selezionato.

Vuol dire che se l'istruzione è destinata alla parte della risposta del nodo di dialogo selezionato, viene eseguita immediatamente. Ovvero, il sistema non valuta la parte della condizione del nodo di dialogo selezionato ed esegue immediatamente la parte della risposta. La specifica della risposta è utile per concatenare insieme più nodi di dialogo. La parte della risposta del nodo di dialogo selezionato viene elaborata come se la condizione di questo nodo di dialogo fosse true. Se il nodo di dialogo selezionato ha un'altra azione *Passa a*, anche tale azione viene eseguita immediatamente.

Invece se l'istruzione è destinata alla parte della condizione del nodo di dialogo selezionato, il servizio verifica prima se la condizione del nodo di destinazione viene valutata come true. Se la condizione viene valutata come true, il sistema elabora immediatamente questo nodo aggiornando il contesto con il contesto del nodo di dialogo e l'output con l'output del nodo di dialogo. Se la condizione non viene valutata come true, il sistema continua il processo di valutazione di una condizione del nodo di pari livello successivo del nodo di dialogo di destinazione e così via, finché non trova un nodo di dialogo con una condizione valutata come true. Se il sistema elabora tutti i nodi di pari livello e nessuna condizione viene valutata come true, verrà utilizzata la strategia di fallback di base e il dialogo valuterà anche i nodi al livello superiore. La specifica della condizione è utile per concatenare le condizioni dei nodi di dialogo.



Nella progettazione dei nodi è importante ricordarsi che l'albero viene elaborato dall'alto verso il basso ed è quindi molto importante stabilire i rami da includere e dove posizionarli. Mentre scende lungo la struttura ad albero, se il servizio trova una condizione che viene soddisfatta, attiverà tale nodo. Si sposta quindi da sinistra a destra sul nodo attivato per controllare l'input utente e verificare la presenza di eventuali condizioni del nodo figlio. Mentre controlla i nodi figlio, si sposta di nuovo dall'alto verso il basso. Il servizio continua il suo percorso attraverso la struttura ad albero del dialogo dall'alto verso il basso, da sinistra a destra, quindi di nuovo dall'alto verso il basso e da sinistra a destra finché non raggiunge l'ultimo nodo nel ramo che sta seguendo.

Per modellare l'albero in modo da raggiungere l'obiettivo del prototipo si è deciso di dividere la conversazioni in più fasi, ciascuna con uno specifico obiettivo.

Innanzitutto si è cercato di gestire le aspettative stabilendo una solida esperienza utente sin dall'inizio tramite un messaggio di benvenuto che spiegasse in maniera chiara cosa il chatbot può fare. In questa maniera vengono subito chiarite le capacità e le limitazioni che in caso non siano chiare potrebbero portare l'utente a rimanere deluso e scontento. Il messaggio iniziale inoltre

permette all'utente di capire che si trova davanti ad un chatbot e non ad un essere umano.

I flussi di conversazione sono stati progettati in modo da ridurre l'attrito e permettere di raggiungere l'obiettivo in pochi passaggi. Tenendo conto di una potenziale distribuzione su mobile si è tenuto conto anche dello spazio occupato, scegliendo risposte concise e non troppo lunghe.

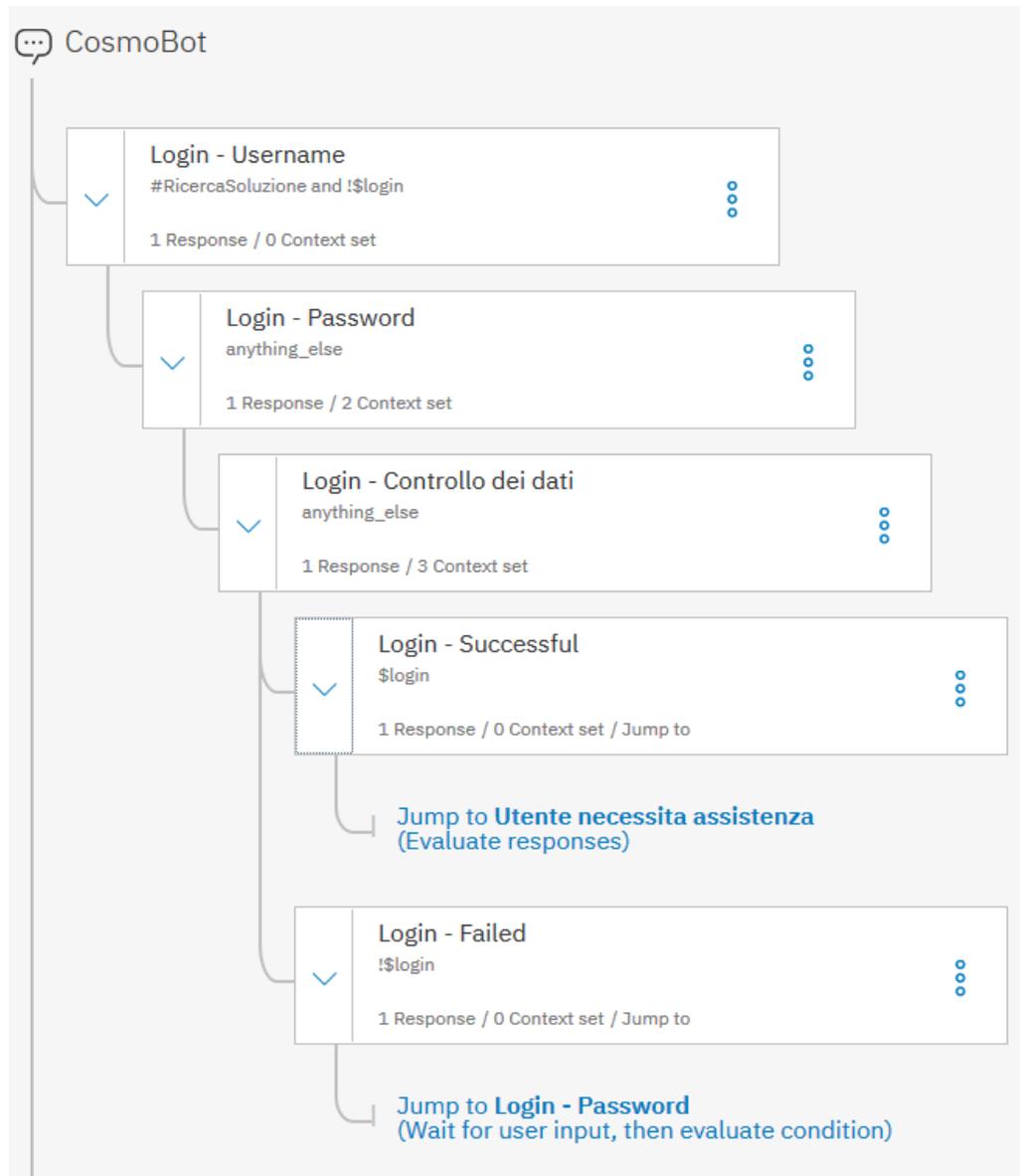
Altro passo fondamentale è stata la scelta di soluzioni per gestire l'imprevedibilità dell'input degli utenti tramite fallback con operatore umano e dei contatori per non permettere l'utente di rimanere bloccato in un ramo dell'albero conversazionale.

4.5.1 Benvenuto

The screenshot shows a configuration interface for a chatbot node titled "Benvenuto". At the top right, there is a "Customize" button with a gear icon and a close button with an "X" icon. Below the title, there is a section "If bot recognizes:" with two input fields: "welcome|" and "#Welcome", separated by "or". Each field has a minus sign icon to its left and a plus sign icon to its right. Below this is a section "Then respond with:" with a vertical ellipsis icon to its right. A list of responses is shown, with the first one being "1. Salve! Sono CosmoBot ,il tuo assistente per segnalare e risolvere ogni difetto. Come posso aiutarti?". To the right of this text is a minus sign icon. Below the list is a text input field with the placeholder "Add a variation to this response". At the bottom, there is a section "And finally" with a dropdown menu currently set to "Wait for user input".

Se il sistema riconosce la condizione speciale welcome o l'intent Welcome, entra dentro questo nodo. Il chatbot si presenta e aiuta l'utente, spiegandogli il servizio che può offrire.

4.5.2 Login



Il primo nodo di dialogo è quello relativo alla fase di login. Come da requisiti, per accedere al servizio di assistenza, l'utente deve inserire le proprie credenziali.

Login - Username ⚙️ Customize ✕

If bot recognizes:

#RicercaSoluzione ⊖ and ▾ !\$login ⊖ ⊕

Then respond with: ⋮

```
1 {
2   "output": {
3     "text": {
4       "values": [
5         "Per accedere al servizio devi fornirmi i tuoi dati di accesso. Inserisci il
        tuo username"
6       ],
7       "selection_policy": "sequential"
8     }
9   }
10 }
```

And finally

Wait for user input ▾

Per accedere al nodo il sistema deve riconoscere l'intent `#RicercaSoluzione`, con il quale l'utente esprime il bisogno di assistenza, e la variabile di contesto `$login` che inizialmente è impostata a `false`. All'interno del nodo il sistema chiede all'utente di inserire lo username con cui accedere al servizio ed attende la risposta

Login - Password

 Customize

If bot recognizes:

anything_else  

Then respond with:



```
1 {
2   "context": {
3     "login_password": "",
4     "login_username": "<? input.text ?>"
5   },
6   "output": {
7     "text": {
8       "values": [
9         "Inserisci la tua password"
10      ],
11     "selection_policy": "sequential"
12   }
13 }
14 }
```

And finally

Wait for user input 

Il nodo "Login - Username" ha un unico nodo figlio che viene attivato tramite la condizione speciale `anything_else` quando l'utente invia un messaggio. Dentro al nodo viene salvata la variabile di contesto `login_username` con l'input dell'utente e il nodo termina chiedendo all'utente di fornire la password.

Login - Controllo dei dati Customize ✕

If bot recognizes:

anything_else ⊖ ⊕

Then respond with: ⋮

```
1 {
2   "context": {
3     "request_login": true,
4     "login_password": "<? input.text ?>",
5     "login_username": "$login_username"
6   },
7   "output": {
8     "text": {
9       "values": [
10        "Confermi i dati inseriti?"
11      ],
12      "selection_policy": "sequential"
13    }
14  }
15 }
```

And finally

Wait for user input ▼

Il nodo "Login - Controllo dei dati" come il precedente si attiva con la condizione speciale. All'interno viene settata la variabile di contesto "login_password" con i dati inseriti dall'utente e la variabile di contesto "request_login" con il valore true. Quest'ultima in particolare serve per identificare il nodo in cui è necessario collegarsi ad una fonte di dati esterna per validare, in questo caso, l'accesso ad un utente.

I dati relativi alla password vengono trasformati con una funzione di hash e confrontati con i dati degli utenti abilitati ad accedere al servizio. Se i dati sono corretti allora deve essere settata la variabile di contesto \$login che sarà successivamente utilizzata come condizione per accedere ad uno degli ultimi due nodi. La variabile di contesto \$login deve necessariamente essere

settata dall'applicazione esterna proprio perché il servizio Conversation non ha un collegamento diretto al database.

Login - Successful ⚙️ Customize ✕

If bot recognizes:

\$login ⊖ ⊕

Then respond with: ⋮

```
1 {
2   "output": {
3     "text": {
4       "values": [
5         "Login effettuato con successo!"
6       ],
7       "selection_policy": "sequential"
8     }
9   }
10 }
```

And finally

Jump to... ▾ "Utente necessita assistenza" (Response) ⊖

Se l'utente ha effettuato correttamente il login allora verrà reindirizzato al nodo di conversazione "Utente necessita assistenza" con modalità "Respond", ovvero non verrà valutata la condizione del nodo.

Login - Failed ⚙️ Customize ✕

If bot recognizes:

!\$login ⊖ ⊕

Then respond with: ⋮

```
1 {
2   "output": {
3     "text": {
4       "values": [
5         "Dati non corretti!"
6       ],
7       "selection_policy": "sequential"
8     }
9   }
10 }
```

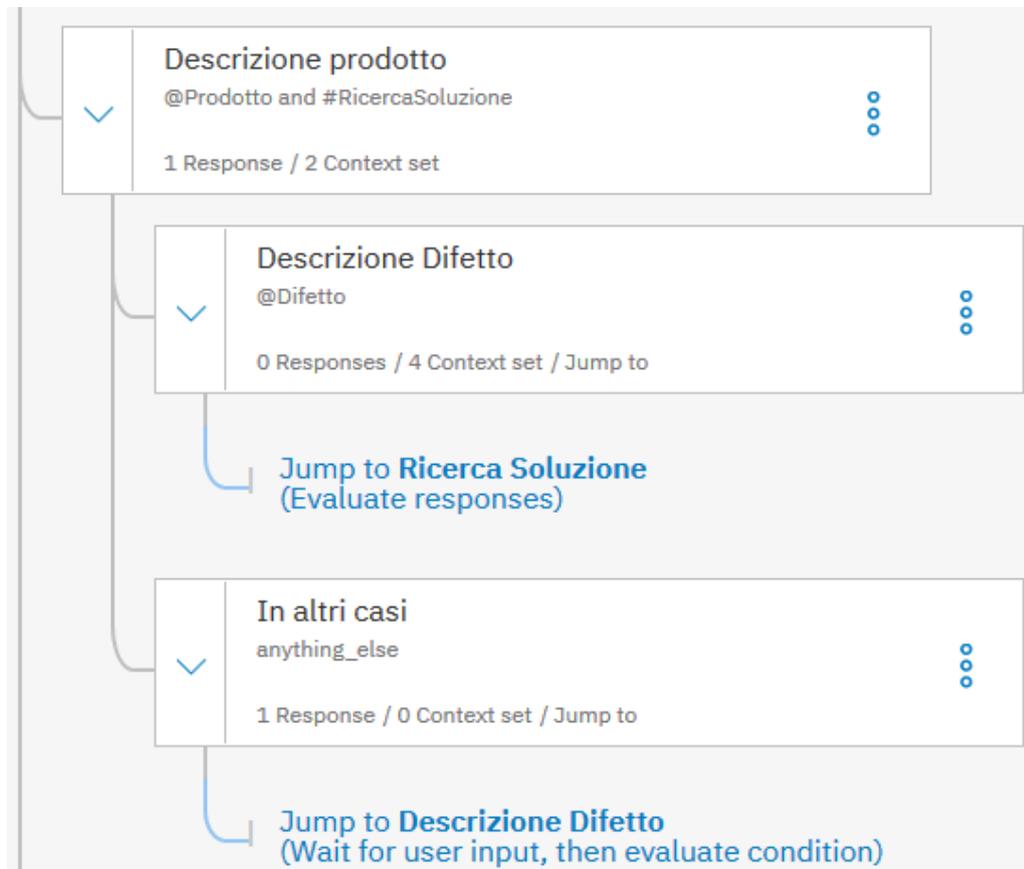
And finally

Jump to... ▾ "Login - Password" (User input) ⊖

Nel caso in cui i dati inseriti dall'utente non trovino una corrispondenza con i dati degli utenti abilitati allora si ritorna all'inizio di questo nodo conversazionale in attesa che l'utente reinserisca entrambi i dati.

4.5.3 Utente necessita assistenza

Questo nodo conversazionale è il terzo a partire dall'alto, ma è il secondo che si incontra seguendo l'ordine della conversazione. La modellazione di questa fetta di dialogo si è svolta con un iniziale idea che poi è stata sostituita da tecniche più efficaci e semplici.



Nella prima modellazione Il primo nodo che si incontra è quello che viene attivato quando il sistema riconosce l'intento dell'utente di cercare una soluzione e riconosce l'entità "Prodotto" all'interno dell'input.

Descrizione prodotto

 Customize

If bot recognizes:

@Prodotto  and  #RicercaSoluzione 

Then respond with:



```
1 {
2   "context": {
3     "prodotto": "@Prodotto.literal",
4     "prodotto_name": "@Prodotto"
5   },
6   "output": {
7     "text": {
8       "values": [
9         "Prodotto riconosciuto: $prodotto . Qual'è il problema con questo prodotto?"
10      ],
11     "selection_policy": "sequential"
12   }
13 }
14 }
```

And finally

Wait for user input 

Quando ciò avviene vengono salvate le variabili di contesto relative al prodotto, e si prosegue al nodo successivo in attesa che l'utente risponda con il problema che si è presentato con quel prodotto.

Descrizione Difetto ⚙️ Customize ✕

If bot recognizes:

@Difetto ⊖ ⊕

Then respond with: ⋮

```
1 {
2   "context": {
3     "difetto": "@Difetto.literal",
4     "prodotto": "$prodotto",
5     "difetto_name": "@Difetto",
6     "prodotto_name": "$prodotto_name"
7   },
8   "output": {}
9 }
```

And finally

Jump to: ["Ricerca Soluzione" \(Default\)](#) ↻

Se il problema viene riconosciuto allora vengono salvate le variabili di contesto relative al problema e si passa al nodo conversazionale "Ricerca Soluzione".

In altri casi  [Customize](#) 

If bot recognizes:

anything_else  

Then respond with: 

1. Devi essere più preciso. Spiegami in maniera diversa il problema che si presenta 

Add a variation to this response

And finally

Jump to...  "Descrizione Difetto" (User input) 

Altrimenti si torna al nodo "Descrizione Difetto" in attesa che l'utente descriva il problema in maniera differente.

Questa soluzione utilizza molti nodi ed inoltre non permette di specificare prima il difetto e poi il prodotto. Si potrebbe risolvere facilmente aggiungendo altri nodi, ma avendo scelto di utilizzare un piano con limitazioni si è scelto di adottare lo strumento degli slot per raccogliere informazioni in un unico nodo. I dettagli che l'utente fornisce all'inizio vengono salvati e il servizio chiede solo i dettagli che non lo sono.

Si possono utilizzare gli slot per ottenere le informazioni necessarie prima di poter rispondere con precisione all'utente aggiungendo condizioni di risposta che tengano conto delle informazioni sulla posizione fornite.

L'utilizzo di slot produce un flusso di dialogo più naturale tra l'utente e il servizio ed è più facile da gestire che cercare di raccogliere le informazioni utilizzando molti nodi separati.

Utente necessita assistenza  Customize 

If bot recognizes:

#RicercaSoluzione  

Then check for:  Manage handlers

	Check for	Save it as	If not present, ask	Type		
1	<u>@Prodotto</u>	<u>\$Prodotto</u>	<u>Per poterti aiutare ho</u>	Required		
2	<u>@Difetto</u>	<u>\$Difetto</u>	<u>Inserisci il problema</u>	Required		

 Add slot

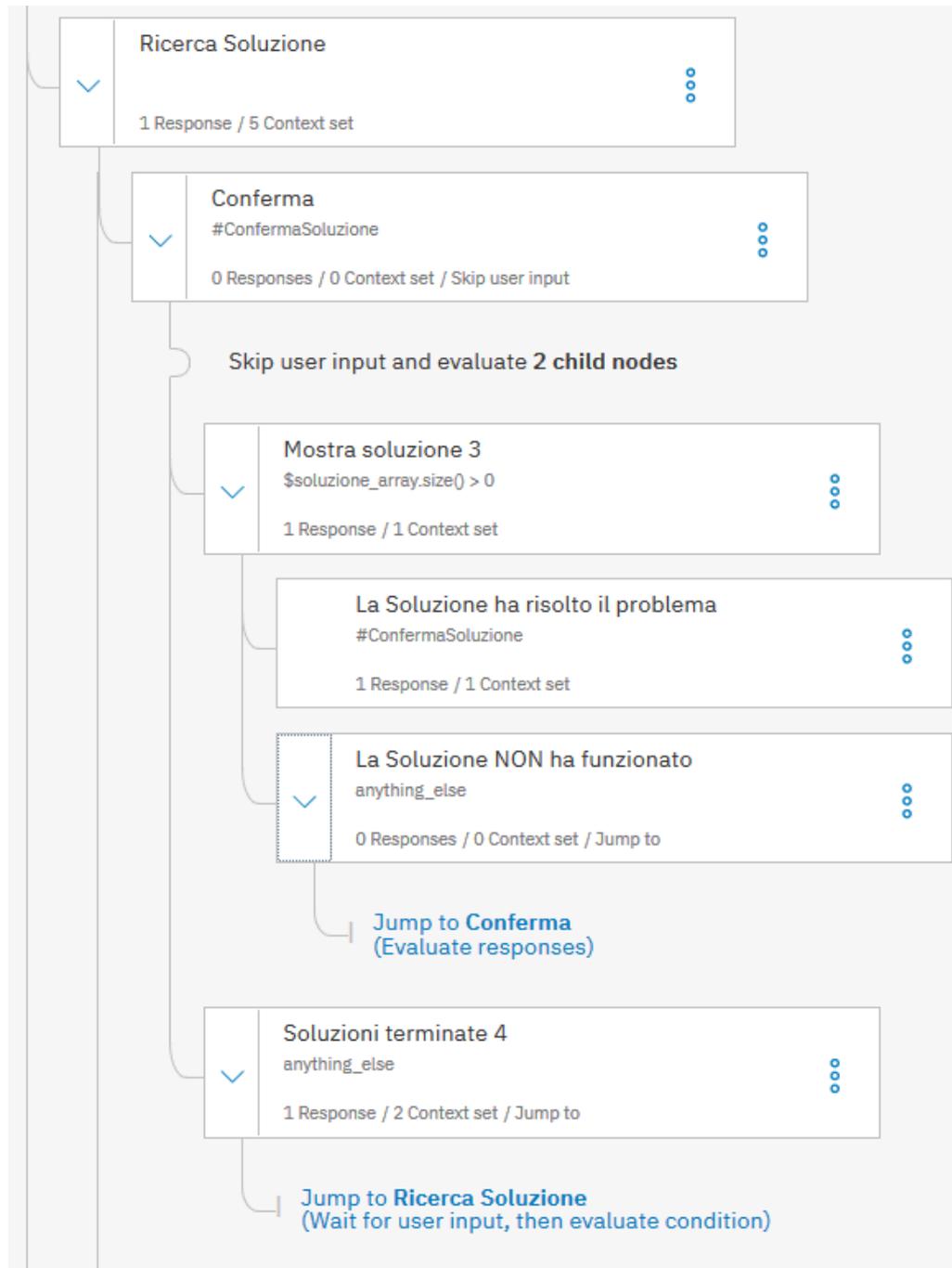
Then respond with: 

1. Sto esaminando le informazioni e a breve ti darò una soluzione 

Add a variation to this response

Tutti i nodi vengono raggruppati in un unico nodo che come condizione ha il riconoscimento dell'intento relativo alla ricerca della soluzione. L'utente rimane all'interno del nodo finché il sistema non ha raccolto le informazioni necessarie per trovare delle soluzioni. Il primo slot cerca di recuperare le informazioni riguardanti l'entità Prodotto mentre il secondo slot quelle del Difetto. Entrambi gli slot sono "required" e quindi è possibile passare al nodo "Ricerca Soluzione" solo quando entrambi sono stati identificati.

4.5.4 Ricerca della soluzione



Il secondo nodo conversazionale ha il compito di trovare e dare le soluzioni alle richieste di assistenza degli utenti.

Ricerca Soluzione
Customize ✕

If bot recognizes:

Enter an intent, entity or context variable... ⊖

Then respond with: ⋮

```

1 {
2   "context": {
3     "difetto": "@Difetto.literal",
4     "prodotto": "$prodotto",
5     "difetto_name": "@Difetto",
6     "prodotto_name": "$prodotto_name",
7     "search_soluzioni": true
8   },
9   "output": {
10    "text": {
11      "values": [
12        "Sto cercando una soluzione. Conferma di avere un problema $difetto con
13        $prodotto"
14      ],
15      "selection_policy": "sequential"
16    }
17  }
```

And finally

Wait for user input ▾

Per cercare le soluzioni è necessario utilizzare i dati all'interno del database e per far questo, nel nodo padre, viene impostato a true la variabile di contesto "search_soluzioni". L'applicazione deve essere strutturata in modo tale che, quando questo flag è impostato, le soluzioni devono essere cercate in base ai parametri impostati precedentemente e inserite all'interno di un'altra variabile di contesto "\$soluzione_array".

Mostra soluzione

[Customize](#)

If bot recognizes:

`$soluzione_array.size() > 0`

Then respond with:



```
1 {
2   "context": {
3     "show_soluzione": true
4   },
5   "output": {
6     "text": {
7       "values": [
8         "Soluzione trovata:"
9       ],
10      "selection_policy": "sequential"
11    }
12  }
13 }
```

And finally

Wait for user input

Se l'array delle soluzioni non è vuoto allora viene presentata una soluzione all'utente e viene rimosso l'elemento dall'array delle soluzioni.

La Soluzione ha risolto il problema

 Customize



If bot recognizes:

#ConfermaSoluzione  

Then respond with:



```
1 {
2   "context": {
3     "show_soluzione": false
4   },
5   "output": {
6     "text": {
7       "values": [
8         "Ottimo! Sono contento di sapere che il tuo problema è risolto. Cordiali saluti
9         e buona giornata"
10      ],
11     "selection_policy": "sequential"
12   }
13 }
```

And finally

Wait for user input 

Questa soluzione potrebbe essere accettata e quindi la conversazione esce dal nodo relativo alla ricerca della soluzione oppure potrebbe non essere accettata perché il problema persiste. Nel secondo caso allora il sistema propone una nuovo elemento dell'array delle soluzioni. Questo ciclo continua finché la variabile di contesto delle soluzioni non è vuota.

Soluzioni terminate Customize ×

If bot recognizes:

anything_else − +

Then respond with: ⋮

```
1 {
2   "context": {
3     "show_soluzione": false,
4     "operator_requested": true
5   },
6   "output": {
7     "text": {
8       "values": [
9         "Non ho più soluzioni disponibili. Verrai contattato da un operatore"
10      ],
11     "selection_policy": "sequential"
12   }
13 }
14 }
```

Nel nodo "Soluzioni terminate" viene settata una nuova variabile di contesto "operator_requested". Questa variabile serve per attivare il meccanismo di fallback per gestire situazioni in cui tutte le soluzioni proposte non hanno risolto il problema. Il meccanismo in questione disattiva il sistema conversazionale permettendo il subentro di un operatore manuale.

L'operatore a questo punto si presenta all'utente in quanto essere umano e può decidere di continuare a comunicare tramite chat oppure chiedere il contatto telefonico dell'utente per comunicare a voce. La cosa molto importante in questo passaggio è che l'utente venga informato di questo cambio tra chatbot e operatore umano perché un utente se parla con un bot ha aspettative diverse dal parlare con un essere umano e quindi non si vogliono generare incomprensioni e scontenti.

4.5.5 In altri casi

In altri casi  

If bot recognizes:

anything_else ⊖ ⊕

Then respond with: 

1. Non ho capito. Puoi riformulare la frase? ⊖

2. Puoi ripetere usando altre parole? Non ho capito. ⊖

3. Non ho capito cosa mi hai detto. ⊖

Add a variation to this response

Variations are **sequential**. [Set to random](#) ⓘ

And finally

Wait for user input ▼

Questo è il nodo di fallback di default che cattura tutte le richieste non gestibili tramite la condizione speciale "anything_else".

4.6 Deploy

Un chatbot per offrire un servizio in maniera efficiente deve essere integrato in maniera corretta. Se non è visibile non è utilizzato e quindi è inutile. Esistono diverse piattaforme di messaggistica istantanea pronte ad accogliere un chatbot, ma ogni scenario è differente e quindi è necessario scegliere una strategia sulla quale potenzialmente avremo migliori performance.

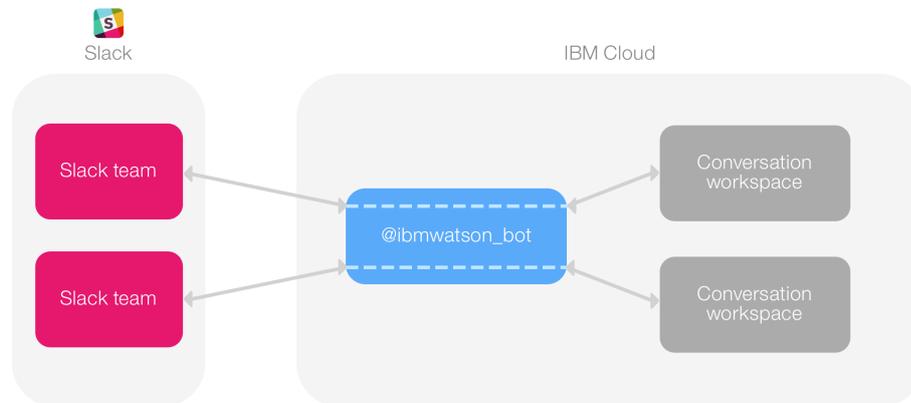
Ad oggi Messenger è la piattaforma più consigliata in quanto, trovandosi su Facebook, offre un pool di utenti significativo. In questo caso d'uso però il numero di utenti non è un fattore cruciale poiché il servizio è orientato al customer service e non alla ricerca e all'ottenimento di nuovi utenti.

Un approccio differente consiste di sviluppare o adottare una piattaforma proprietarie con possibilità di integrare le informazioni ricevute dal Chatbot in un CRM. Gli utilizzatori del servizio sono utenti che attualmente sono abituati ad accedere al sito dell'azienda e di conseguenza è giusto non cambiare le loro abitudini per evitare malcontento. Inoltre utilizzare un'app di messaggistica rende implicito l'obbligo di scaricare l'app in questione e quindi ad una potenziale diminuzione degli utenti. L'ultimo motivo per il quale questo approccio risulta ottimale per lo scenario è l'assenza di re-engagement visto che sarà l'utente a ricontattare il chatbot se ne avrà nuovamente bisogno.

Con IBM Watson è possibili distribuire l'applicazione connettendo lo spazio di lavoro all'interfaccia che verrà utilizzata dai clienti in diversi modi. [cit LINK]. Nella schermata Deploy dello strumento Conversation è possibile visualizzare i dati relativi al workspace e le credenziali per accedervi.

L'esecuzione di test in Slack risulta essere una soluzione molto rapida e semplice, ma con delle limitazioni.

- Non è possibile utilizzare questo strumento per pubblicare un'applicazione che venga usata da altri team.
- Se si utilizza questo metodo per distribuire più di uno spazio di lavoro allo stesso team, tutti gli spazi di lavoro risponderanno al nome utente @ibmwatson_bot.
- Bisogna avere le autorizzazioni per installare le applicazioni in un team Slack.
- A causa delle limitazioni di Cloud Functions questo strumento è disponibile solo per la regione Stati Uniti Sud di IBM Cloud.



Un'altra opzione è quella di creare un'applicazione client utilizzando l'SDK Watson per sviluppare il front-end dell'applicazione. Questa soluzione in particolare è quella che è stata scelta per sviluppare il prototipo poiché permette una personalizzazione elevata e la possibilità di integrazione con i servizi di terze parti o con i sistemi aziendali.

L'ultimo metodo è quello di distribuire il chatbot sui canali di social media e di messaggistica tramite il plug-in middleware Botkit. Una volta stabilite le integrazioni, i clienti possono avere conversazioni indipendenti e simultanee attraverso diversi canali. Come scritto precedentemente non è richiesta l'integrazione sulle piattaforme di messaggistica e quindi per lo sviluppo del prototipo, quest'ultimo metodo, non verrà preso in considerazione. Definita correttamente la conversazione tramite gli strumenti di IBM Watson sarà comunque possibile sfruttare questa integrazione senza nessun problema.

4.6.1 Application Backend

L'applicazione che gestisce il servizio Conversation di IBM Watson è stata sviluppata utilizzando Node JS.

Quando un utente si collega al server innanzitutto avviene la connessione al servizio IBM tramite le apposite API e la connessione al database di supporto.

```
conversation = watson.conversation({
  username: '<username>',
  password: '<password>',
  version_date: '2017-05-26',
  version: 'v1',
});
```

Se le connessioni vengono stabilite con successo allora l'utente può iniziare la conversazione. Quando un messaggio viene inviato tramite le API è necessario passare come input, sia il workspace id che fa riferimento allo spazio di lavoro su cui è presente il sistema di Conversation progettato, sia il contesto della conversazione. Quest'ultimo è essenziale per poter continuare un dialogo lungo l'albero conversazionale, e va reimpostato ogni volta che si riceve una risposta dal servizio Conversation.

```
function send_message_to_chatbot(msg, context) {

  conversation.message({
    workspace_id: workspace_id,
    input: {'text': msg},
    context: context != null ? context.context : null,
  }, function (err, response) {
    if (err) {
      console.log('error:', err);
    } else {
      conversation_context = response;
      parse_response(response);
    }
  });
}
```

Ricevuta una risposta ci sono tre importanti casi da gestire:

- la richiesta di assistenza da un operatore umano quando non ci sono più soluzioni disponibili da offrire all'utente. Viene catturata dalla variabile di contesto `operator_requested` e disattiva il servizio Conversation. Da questo momento in poi il chatbot non serve, ma lo

scambio di messaggi avviene tra utente e operatore manuale tramite le loro interfacce conversazionali.

- la richiesta di login quando la variabile di contesto `request_login` è impostata. In questo caso i dati inseriti vengono passati alla funzione `check_login` che imposta un'altra variabile di contesto `login` in base al fatto che i dati siano validi o no.
- la ricerca di soluzioni se è stata impostata la variabile di contesto `search_soluzioni`. Vengono recuperate le soluzioni in base al prodotto e al difetto che l'utente ha specificato e vengono inserite nella variabile di contesto `soluzione_array`. Il flag `search_soluzioni` viene disabilitato perché le soluzioni verranno nuovamente cercate solo se cambieranno i parametri nella richiesta utente.

```
function parse_response(response) {  
  
    if (response.context.operator_requested) {  
        operator_requested = true;  
  
        var message = dettaglio_conversazione;  
        var output = JSON.stringify(message, null, 2);  
        io2.emit("message_by_operator", output);  
    }  
  
    if (response.context.request_login) {  
        response.context.login = check_login(response);  
    }  
  
    if (response.context.search_soluzioni) {  
        response.context.soluzione_array = get_solutions(response);  
        response.context.search_soluzioni = false;  
    }  
  
    send_message_to_client(response);  
}
```

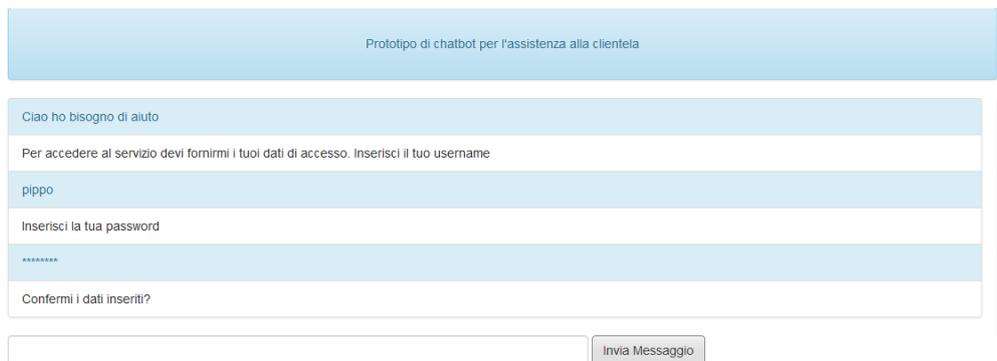
Se il dialogo non si trova ancora in nessuno dei tre casi allora la risposta da parte del servizio Conversation viene inviata al client dell'utente e visualiz-

zata.

4.6.2 Application Frontend

Le interfacce per comunicare con il servizio sono state sviluppate in HTML5 utilizzando Bootstrap per la parte CSS e JQuery per la parte dinamica della pagina. In totale le interfacce sono una per gli utenti ed una per l'operatore umano.

Attualmente le interfacce per testare il prototipo sono uguali, ma non c'è alcun vincolo che vieti di renderle diverse. Anzi, per l'interfaccia dell'operatore, sarebbe più opportuno fare un sistema che permetta di visualizzare tutte le richieste in sospeso ed eventualmente gestire più richieste contemporaneamente.



The image shows a web-based chatbot interface. At the top, a light blue header contains the text "Prototipo di chatbot per l'assistenza alla clientela". Below this is a chat window with a light blue background. The chat window contains the following text: "Ciao ho bisogno di aiuto", "Per accedere al servizio devi fornirmi i tuoi dati di accesso. Inserisci il tuo username", "pippo", "Inserisci la tua password", "*****", and "Confermi i dati inseriti?". Below the chat window is a text input field and a button labeled "Invia Messaggio".

Nella progettazione dell'interfaccia si è preso la decisione importante di non utilizzare bottoni con risposte preimpostate perché non sarebbe possibile coprire tutto il dominio di difetti da mostrare e comunque l'obiettivo è di simulare una conversazione umana e quindi incompatibile con l'uso di un pulsante. Inoltre l'utente in questa maniera è più libero potendo scrivere quello che preferisce e non deve forzatamente seguire un percorso già stabilito.

4.6.3 Improve

Una parte essenziale, dopo aver progettato e testato il dialogo, è quella di analizzare le conversazioni avvenute.

Tramite lo strumento "Panoramica" nel pannello Improve è possibile visualizzare un riepilogo delle interazioni tra gli utenti e chatbot. Si possono visualizzare la quantità di traffico per un determinato periodo di tempo, nonché gli intenti e le entità che sono stati riconosciuti più spesso nelle conversazioni degli utenti.

Questo strumento è molto importante per controllare il funzionamento del chatbot, la reale utilizzabilità ed ottenere statistiche di diversa natura.



Nella pagina principale è possibile vedere un riassunto del numero di conversazioni, degli intenti riconosciuti e anche delle entità. La parte superiore della pagina include alcuni comodi controlli per filtrare i valori, aggiornare le statistiche immediatamente oppure per variare il periodo di controllo. Per il controllo del periodo temporale in base alla scelta i punti dati sul grafico si adattano ad un periodo di misurazione appropriato offrendo una visualizzazione comoda ed utile.

Il primo grafico nella pagina mostra il numero delle interazioni, di qualunque tipo, avvenute con il chatbot. Queste conversazioni possono essere esaminate dettagliatamente ed addirittura modificate in modo tale che il sistema Conversation venga trainato con i valori nuovi.

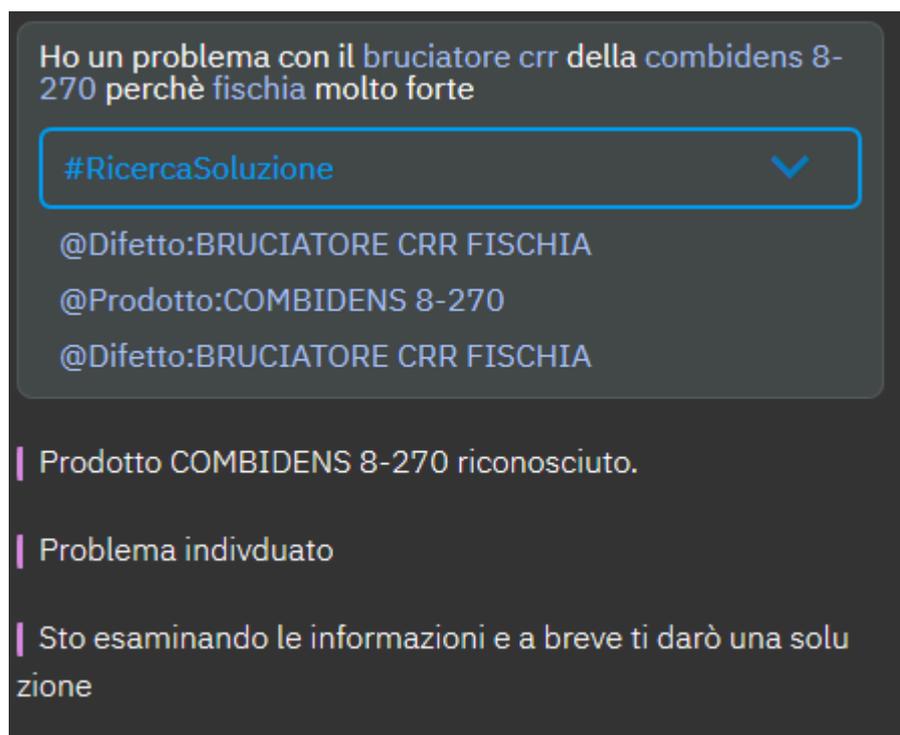
Nella parte inferiore invece si visualizzano la statistica degli intenti e delle

entità che sono state riconosciute più spesso durante il periodo di tempo specificato. Gli intenti principali sono mostrati in un elenco semplice mentre le entità principali sono mostrate in un grafico a barre. In base al piano scelto la durata prevista per la conservazione dei messaggi di chat varia da 7 giorni per il piano gratuito a 90 giorni per il piano Premium.

Quindi, anche se il chatbot è stato progettato al meglio possibile, sarà sempre necessario una sorta di affiancamento umano per poterne garantire performance elevate.

4.7 Test

Per capire se il dialogo modellato funziona sono stati eseguiti diversi test utilizzando lo strumento di chat di Watson con il quale è possibile controllare la risposta per vedere se il dialogo ha interpretato correttamente l'input e scelto la risposta corretta.



La finestra di chat indica quali intenti ed entità sono stati riconosciuti nell'input inoltre nel riquadro dell'editor di dialogo, è evidenziato il nodo attualmente attivo. Dal menù a tendina è possibile, se il riconoscimento è fallito, modificare l'intento riconosciuto per fare in modo che, il sistema Conversation, venga addestrato correttamente. inoltre è possibile visualizzare anche tutte le variabili di contesto impostate ed eventualmente impostarne delle altre o modificarne di già esistenti per fare dei test.

Il controllo principale è stato quello di valutare dopo ogni input inserito, che il nodo attivato fosse quello corretto e che il dialogo seguisse l'albero conversazionale prestabilito.

Purtroppo il sistema Conversation non prevede la progettazione di uno strumento per eseguire test automatici dopo eventuali modifiche e quindi tutte le prove sono state effettuate manualmente.

4.7.1 Key Performance Indicator

Il sistema è un prototipo e non un prodotto finito pronto ad essere utilizzabile dagli utenti, ma è giusto definire un sistema di misurazione delle performance del chatbot.

I KPI o Key Performance Indicator sono degli indici collegati con gli obiettivi di Business ed è possibile definirli in base ad alcuni parametri:

- Aumento dei ricavi dovuti alla riduzione dei costi in quanto non è più necessario avere un team di assistenza clienti. Inoltre la soddisfazione dei clienti si traduce automaticamente in un aumento dei guadagni perché saranno più propensi a continuare a compere i servizi o prodotti.
- Assenza dell'intervento umano, correlato con l'aumento dei ricavi in quanto se il chatbot è in grado di operare da solo, non sarà necessario l'intervento di un operatore, il cliente sarà subito soddisfatto ed i ricavi aumenteranno.
- Una scala di valutazione da utilizzare al termine della conversazione per sapere se all'utente è piaciuta la modalità di interazione, ma soprattutto se ha ricevuto effettiva assistenza.
- Tasso di attivazione ovvero il numero di corrette interazioni tra chatbot ed utente.

- Percentuale di errori per capire se un chatbot è stato programmato efficientemente. Una risposta come "non ho capito" non è un errore, ma è importante che l'utente capisca che non può chiedere qualsiasi cosa al chatbot perché è in grado di rispondere solo a precisi input.
- Tasso di conversione. La percentuale di utenti che ritornano ad utilizzare il chatbot è sicuramente un dato di cui bisogna tenere conto perché chi riprende un dialogo, molto probabilmente è rimasto soddisfatto dell'interazione precedente. A seconda del tipo di chatbot il rate di ritorno è diverso e va tenuto in considerazione.
- Intelligenza artificiale per valutare se il chatbot è in grado di risolvere la maggior parte delle interazioni e se è capace di apprendere
- Velocità di apprendimento.

4.8 Evoluzioni

Il sistema è stato progettato in modo che sia scalabile e quindi espandibile con altre funzionalità. Le potenzialità sono enormi ed alcune delle evoluzioni che il sistema potrebbe subire sono:

- Modellazione più efficiente dei nodi conversazionali. In questo momento un utente se non conosce il nome del prodotto non riuscirà ad ottenere assistenza. Si potrebbe cercare di modellare la conversazione per aiutare l'utente a trovare il prodotto utilizzando informazioni come ad esempio la matricola prodotto o il serial number.
- Più funzionalità partendo dall'integrazione con i servizi attualmente presenti sul sito. Quest'evoluzione è semplice perché basta aggiungere dei nodi all'albero conversazionale senza creare problemi ai nodi già esistenti.
- Profilazione dell'utente. Tramite l'interazione con un chatbot è possibile profilare l'utente, ovvero etichettarlo in base alle risposte che ha dato. Una volta profilata la persona, si ha la possibilità con un chatbot di inviare messaggi personalizzati e programmati solo a un determinato target di utenti.

- Integrazione del servizio Speech to Text che permette di trascrivere l'audio in testo in modo tale da comunicare con il chatbot tramite voce, senza dover scrivere nulla.

Conclusioni

I chatbot sono attualmente un trend, ma non è detto che lo siano anche nei prossimi anni.

Nello scenario attuale coesistono fattori che potrebbero far pensare che il futuro dei chatbot sia già segnato. L'evento più eclatante è la chiusura da parte di Facebook del progetto M, l'assistente artificiale con enormi potenzialità, che doveva aprire le porte a nuove opportunità ed eliminare tutti i problemi e svantaggi che affliggono i sistemi di conversazione.

Tuttavia con il progresso tecnologico, aumenteranno i settori che potranno beneficiare dei vantaggi dovuti dall'automazione dei processi aziendali. I chatbot hanno quindi la potenzialità di diventare gli elementi essenziali per standardizzare i canali di comunicazione delle aziende, rendendoli gli unici in grado di rispondere alle necessità dei clienti, ma l'entusiasmo degli utenti per questa nuova tecnologia sta scemando [15]. Attualmente sono la migliore soluzione per rispondere alle richieste esigenti dei clienti e finché non nascerà un nuovo sistema in grado di superare i chatbot si avrà tempo di cercare soluzioni per renderli più affidabili e robusti. Si prevede quindi un periodo ricco di novità che potranno essere sfruttate per poter migliorare il prototipo progettato, rendendolo uno strumento ancora più solido ed utile. Forse non si sta vivendo ancora una vera e propria svolta a livello globale, ma in questo caso di studio è stata una vera rivoluzione. Il prototipo oltre ad aver suscitato molto entusiasmo nel suo utilizzo, ha permesso all'azienda di espandere i propri confini ed aprire la possibilità a nuovi scenari prima impossibili.

La progettazione di questo prototipo non è sufficiente per avere un prodotto finito e subito utilizzabile, ma è in grado di garantire una reale soddisfazione delle esigenze espresse dagli utenti. Su tutte le interazioni avvenute con il chatbot, internamente nell'azienda, circa l' 80% di queste ha riconosciuto l'intenzione dell'utente e ha proposto una soluzione coerente. Le interazioni

restanti sono state inoltrate ad un operatore manuale che, in uno scenario reale, avrebbe provveduto a processare. Il numero di interventi avvenuti con successo ovviamente tende ad aumentare grazie all'apprendimento del sistema Watson e al sistema di gestione delle soluzioni implementato insieme al chatbot. Inoltre in un futuro non troppo lontano, estendendo il sistema, si potrebbe riuscire a coprire tutto il dominio delle richieste degli utenti.

I chatbot, a prescindere dalle difficoltà, sono strumenti utili in molte situazioni e continueranno ad evolversi nel tempo, trasformandosi in assistenti vocali o in qualcos'altro di inaspettato, ma ci vorrà ancora molto tempo affinché gli utenti possano fare affidamento al 100% senza rischiare di rimanere delusi ed insoddisfatti. Attualmente si sta cercando di evolvere i chatbot con meccanismi di interazione circolare per offrire un'esperienza di conversazione il più naturale possibile e cercare di instaurare una connessione in grado di aumentare il coinvolgimento e l'empatia tra esseri umani ed intelligenze artificiali.

Ringraziamenti

Grazie ai miei genitori che mi hanno dato la possibilità di studiare e laurearmi.

Grazie a Silvia per aver intrapreso questo cammino con me e per avermi aiutato a concluderlo insieme.

Grazie ai miei compagni di corso per tutte le risate e i momenti divertenti vissuti.

Grazie a Stack Overflow per aver risolto i miei problemi quando sembrava non esserci più speranza.

Bibliografia

- [1] 2020: realtà virtuale e chatbot domineranno le interazioni con i brand. <https://www.oracle.com/it/corporate/pressrelease/oracle-report-can-virtual-experiences-replace-reality-vr-chatbots-for-cx-20161206.html>.
- [2] 23 <http://info.localytics.com/blog/23-of-users-abandon-an-app-after-one-use>.
- [3] The app explosion is over. <http://www.businessinsider.com/average-number-of-apps-vs-time-spent-2016-5>.
- [4] Benefits of accelerating digitization in businesses. <https://blog.qiscus.com/benefits-accelerating-digitization-businesses/>.
- [5] Bot. <https://it.wikipedia.org/wiki/Bot/>.
- [6] The chatbot landscape, 2017 edition. <https://blog.keyreply.com/the-chatbot-landscape-2017-edition-ff2e3d2a0bdb?gi=8a208bd08401>.
- [7] Chatbots explained: Why businesses should be paying attention to the chatbot revolution. <http://www.businessinsider.com/chatbots-explained-why-businesses-should-be-paying-attention-to-the-chatbot-revolution-2016-7?IR=T>.
- [8] Come organizzare una giornata formativa? l'esempio di chatbot day. <https://www.insidemarketing.it/eventi/giornata-formativa-chatbot-day/?IM=int-link>.
- [9] Crawler. <https://it.wikipedia.org/wiki/Crawler>.
- [10] Creazione di un dialogo. <https://console.bluemix.net/docs/services/conversation/dialog-build.html>.

-
- [11] Definizione di intenti. <https://console.bluemix.net/docs/services/conversation/intents.h>
- [12] Drop it like it's bot: Brands have cooled on chatbots. <https://digiday.com/marketing/brand-bot-backlash-begun/>.
- [13] Facebook's perfect, impossible chatbot. <https://www.technologyreview.com/s/604117/facebooks-perfect-impossible-chatbot>.
- [14] How messenger and "m" are shifting gears. <https://www.theinformation.com/how-messenger-and-m-are-shifting-gears>.
- [15] Inside chatbots' year of growing pains: 'we're at an inflection point'. <https://marketingland.com/inside-chatbots-year-growing-pains-210182>.
- [16] Is your business ready to face digital disruption? <https://blog.qiscus.com/business-face-digital-disruption/>.
- [17] One of facebook messenger's first retail partners has ditched it as a notification tool. <https://www.recode.net/2017/3/1/14779978/everlane-facebook-messenger-stop-customer-service-notifications>.
- [18] Pro-government twitter bots try to hush mexican activists. <https://www.wired.com/2015/08/pro-government-twitter-bots-try-hush-mexican-activists>.
- [19] Thriving in an increasingly digital ecosystem. <https://sloanreview.mit.edu/article/thriving-in-an-increasingly-digital-ecosystem/>.