



**UNIVERSITÀ
DI TORINO**

Corso di Laurea Magistrale in Informatica

Tecnologie del Linguaggio Naturale

Progetto:

Lara Dialogue System

Camoia Andrea

2025

di.unito.it

DIPARTIMENTO
DI INFORMATICA

Indice

1	Introduzione	2
1.1	Obiettivo	2
1.2	Tech Stack	2
1.3	Proprietà del dialogo	2
1.4	Le domande	2
1.4.1	Generazione	2
1.4.2	Tipologie di domande	3
1.5	Struttura della Conversazione	4
2	Dialog Manager	6
2.1	User Frame	6
2.2	Memoria	6
3	Natural Language Generation	7
3.1	Obiettivo del modulo	7
3.2	Strategia di generazione	7
3.3	Generazione della domanda	7
3.4	Generazione del risultato di una risposta	8
4	Natural Language Understanding	8
4.1	Gestione delle domande a risposta singola. [extract_wh_word_answer()] . .	8
4.2	Gestione delle domande a risposta binaria	9
4.3	Gestione delle domande a risposta multipla	9
4.4	Estrazione delle informazioni utente	10
5	Valutazione	10
5.1	Esempi ed Analisi di Dialoghi	11
5.2	Trindi TickList	13

1 Introduzione

1.1 Obiettivo

Il progetto prevede la realizzazione di un **Dialogue System** in grado di **interpretare il personaggio di Lara Croft**, interagendo con l'utente tramite una serie di domande inerenti la storia dell'archeologa. Lo scopo è valutare se l'utente potrebbe essere un valido candidato per diventare suo assistente. Il sistema deve essere coerente con la personalità della famosa eroina, formulando domande di diverso tipo e catturando efficientemente le risposte ricevute. Alla conclusione del "colloquio", il sistema fornirà un giudizio appropriato per decidere l'idoneità dell'utente nel ricoprire il ruolo di assistente.

1.2 Tech Stack

L'intero sistema è stato sviluppato utilizzando **Python** come linguaggio di programmazione.

Per l'elaborazione e analisi del testo, è stata impiegata la libreria **spaCy**, una delle più efficienti e largamente utilizzate nel campo del NLP. In particolare, è stato utilizzato il modello `en_core_web_trf`, una variante avanzata basata su Transformer, in grado di fornire analisi linguistiche profonde e accurate. Le principali operazioni effettuate tramite spaCy, che approfondiremo in seguito, includono *Part-of-Speech (PoS) recognition*, *Named Entity Recognition (NER)* e soprattutto *l'Analisi della struttura sintattica* del testo.

Il sistema opera interamente tramite **interfaccia a riga di comando**, costruito però per essere facilmente integrato con un'ipotetica interfaccia grafica più complessa.

1.3 Proprietà del dialogo

- **Turni:** Il dialogo segue una struttura a turni alternati tra utente e sistema, senza sovrapposizioni o interruzioni (*no barge-in*), rispettando il principio di alternanza.
- **Iniziativa:** L'iniziativa è interamente affidata al sistema, che avvia, guida e conclude la conversazione. Introduce il contesto, pone domande, valuta le risposte e presenta il risultato finale all'utente.
- **Lingua:** Il dialogo si svolge esclusivamente in lingua inglese.

1.4 Le domande

1.4.1 Generazione

Il sistema dispone di un dataset in formato JSON, composto da un totale di **80 domande**. La creazione di questo dataset è avvenuta attraverso un processo articolato in più fasi, con

l'obiettivo di ottenere domande tematicamente coerenti con la figura di Lara Croft, formulate in prima persona e strutturate secondo un formato standard.

La prima fase ha previsto uno scraping del Web Forum¹ ufficiale di Lara Croft, contenente importanti aneddoti e informazioni riguardanti la biografia e la storia del personaggio.

Una volta ottenuti i testi grezzi, questi sono stati sottoposti ed elaborati da alcuni LLM² al fine creare le domande vere e proprie, formulate in prima persona da Lara Croft e soprattutto formattate in una struttura dati unificata, compatibile e favorevole all'utilizzo da parte del Dialogue System. Il risultato di questo processo è un dataset strutturato come segue:

```
[
  ...
  {
    "id": [int],
    "type": [string],
    "question": [string],
    "answer": [string],
    "score": [int]
  }
  ...
]
```

Dove:

- **id**: identificativo univoco della domanda all'interno del dataset.
- **type**: tipologia della domanda (risposta singola, risposta multipla, vero/falso).
- **question**: testo della domanda, formulata in prima persona da Lara Croft.
- **answer**: risposta corretta o attesa, utilizzata per la valutazione dell'utente.
- **score**: punteggio associato alla domanda, che rappresenta sia il livello di difficoltà che il peso in termini di valutazione finale.

1.4.2 Tipologie di domande

Come anticipato, il sistema gestisce tre tipologie di domande, ognuna delle quali prevede un diverso meccanismo di gestione ed elaborazione della risposta:

- **Domande a risposta singola**: `[single_wh_answer]`: che appunto prevedono una singola risposta che può essere una data, una persona, un luogo o qualsiasi informazione. (*Where did I track the Hand of Rathmore after learning it had been taken?*)

¹[https://tombrainer.fandom.com/wiki/Lara_Croft_\(Original_Timeline\)](https://tombrainer.fandom.com/wiki/Lara_Croft_(Original_Timeline))

²ChatGPT 4 e Claude Sonnet 4

- **Domande a risposta multipla** [multiple_answer]: le cui risposte prevedono una lista di informazioni. (*Name the four meteorite artifacts I collected.*)
- **Domande vero o falso** [true_false]: le quali chiedono all'utente di confermare/smentire una data informazione. (*I was officially declared dead after my expedition in Egypt.*)

1.5 Struttura della Conversazione

Come anticipato la conversazione viene aperta dal sistema, che restituisce un messaggio introduttivo e chiedendo all'utente di introdursi (e salvando le informazioni ricevute, come il nome dell'utente) **Superata la fase introduttiva il sistema comincia a porre le domande** all'utente. Una volta ricevuta la risposta possono verificarsi **quattro situazioni** (gestite nel file `question_handlers.py`):

- **L'utente ha risposto correttamente**, e di conseguenza il sistema aggiorna il punteggio e restituisce una risposta consona.
- **La risposta dell'utente è sbagliata**, allora il sistema avverte l'utente dell'errore e procede con la prossima domanda (se esiste).
- **L'utente ha risposto in maniera ambigua**, ad esempio con un messaggio vuoto o che contiene molteplici risposte contrastanti. In questo caso il sistema chiede all'utente di rispondere nuovamente.
- **L'utente ha risposto parzialmente ad una domanda a risposta multipla**. Il sistema in questo caso chiede all'utente se riesce a completare la risposta, per poter valutare di conseguenza il risultato.

In ogni caso, **al termine del ciclo** delle domande, **il sistema elaborerà e restituirà il risultato del colloquio** all'utente.

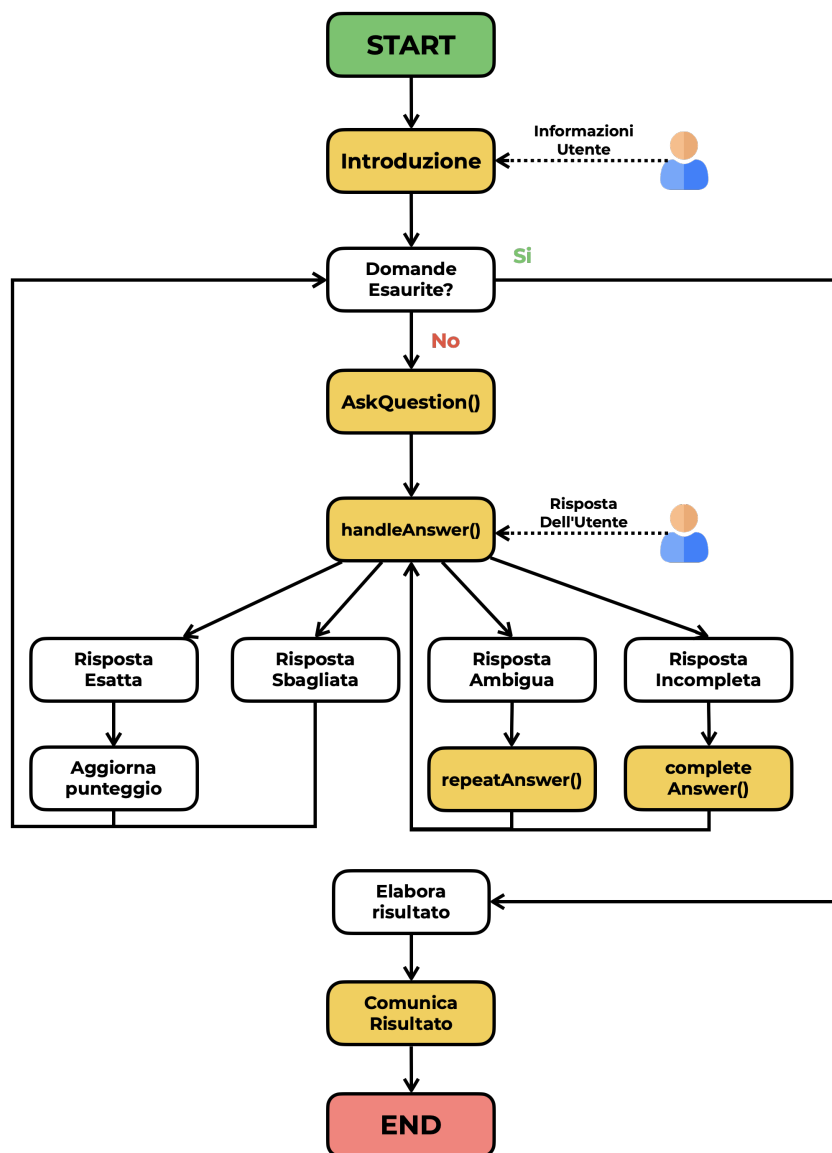


Figure 1.1: Schema della conversazione

2 Dialogue Manager

2.1 User Frame

Il sistema **mantiene e gestisce tutte le informazioni riguardo l'utente in un apposito frame**. Questa struttura permette quindi di accedere in modo rapido ed efficiente alle relative informazioni in qualsiasi momento. In particolare, viene salvato:

```

1 USER_FRAME = {
2     "name" : "Andrea",
3     "score" : 5,
4     "number_of_questions" : 3,
5     "adjective": ["brave", "smart"]
6 }
```

- **Il nome dell'utente**, fornito durante la fase di introduzione.
- **Gli aggettivi** che l'utente ha (eventualmente) inserito nella sua descrizione. Come vedremo queste informazioni sono molto utili per generare delle risposte molto realistiche e contestualizzate.
- **Numero di domande** da effettuare.
- **Punteggio** totalizzato dall'utente.

2.2 Memoria

Per quanto riguarda la memoria, **il sistema tiene traccia di tutte le domande poste fino a quel momento all'utente**, salvando anche la risposta di quest'ultimo ed il relativo risultato.

```

1 MEMORY = {
2     "questions_asked": [62], # list of the question IDs asked.
3     "question_results" : [{
4         "question_id": 62,
5         "question": "The Iris artifact was stored at Von Croy
6                     Industries.",
7         "answer": "Yes, it's true", # user answer to the question
8         "score": 2, # question score
9         "result": "correct" # question result
10    }],
11     "correct_answers" : 1
12 }
```

Ad ogni iterazione quindi, il sistema esegue un'operazione di **slot filling** per riempire il **frame relativo alla domanda** con le informazioni estratte dalla risposta utente. Una struttura di questo tipo risulta estremamente utile per gestire ad esempio le risposte ambigue o incomplete, aggiornando lo specifico frame fin quando non si raggiunge un risultato consono.

Inoltre, la memoria permette di **ottenere un colloquio molto più realistico e contestualizzato in base ai risultati parziali dell'utente**. Ad esempio, permette di reagire ad una striscia di risposte negative da parte dell'utente (magari proponendo una domanda facile), oppure far notare alcuni particolari errori durante il resoconto finale.

3 Natural Language Generation

In questa sezione viene analizzato il modulo di *Natural Language Generation* (NLG), responsabile della generazione del testo che il sistema restituisce all'utente durante l'interazione.

3.1 Obiettivo del modulo

L'obiettivo principale dell'NLG è produrre risposte che siano:

- **Coerenti con il carattere e lo stile comunicativo di Lara Croft.**
- **Contestualizzate rispetto allo stato del dialogo**, ovvero adattate in base alla fase della conversazione, alle risposte dell'utente e alle informazioni già acquisite.

3.2 Strategia di generazione

Per garantire una tale generazione, è stato adottato un approccio basato principalmente su **template predefiniti**. Le frasi sono state progettate a mano in modo da riflettere lo stile di Lara Croft, e includono variabili che vengono **istanziate dinamicamente** in base alle informazioni presenti nel Frame Utente e nella Memoria del dialogo.

L'integrazione di queste componenti consente al sistema di generare risposte **coerenti, fluide e contestualmente appropriate**, e soprattutto permette di restituire frasi molto complesse ed articolate. Cosa che sarebbe stata impossibile utilizzando una generazione dinamica che costruisce il testo ogni volta.

3.3 Generazione della domanda

Uno dei contesti in cui la filosofia di generazione risulta più evitante è durante **la formulazione della domanda** da porre all'utente.

Infatti, invece di stampare direttamente il testo della domanda, il Dialogue System sviluppato è in grado di **formulare un'introduzione coerente con lo stato del colloquio** e sempre diversa, gestendo in maniera fine tutte le possibili situazioni:

- Introduzione alla prima domanda: ["First question, I won't be too evil"]

- Reazione quando l'utente sbaglia la prima domanda (o ancor peggio quando sbaglia le prime due): [f"Not the best way to begin, user_frame['name'}. Let's see if you can recover.",]

3.4 Generazione del risultato di una risposta

Lo stesso concetto di generazione è stato applicato anche al testo restituito dopo ogni risposta dell'utente. Ancora una volta il sistema è in grado di **formulare un testo coerente al risultato della domanda**. Mostriamo qualche esempio:

- **Risposta corretta ad una domanda difficile:** [f"Sharp work, {user_frame['name']}. You're starting to sound like someone I could rely on."]
- **Risposta sbagliata ad una domanda semplice:** [f"No, {user_frame['name']}. Let's try not to trip over the easy stones."]
- **Gesione di una risposta multipla incompleta:** [f"Ok {user_frame['name']} you have {user_correct_answers} correct answers out of {total_possible_answers}. Can you complete the rest of the questions? Otherwise i have to consider this question as incorrect."]
- **Reazione ad una risposta ambigua:** ["You're dancing around the question. I suggest you don't.",]

4 Natural Language Understanding

Il modulo di Natural Language Understanding è una delle componenti più importanti, poichè **la comprensione e l'estrazione delle informazioni rilevanti** dalle risposte dell'utente rappresentano un aspetto cruciale del sistema.

4.1 Gestione delle domande a risposta singola. [extract_wh_word_answer()]

Per quanto riguarda l'estrazione delle informazioni utili a partire dalla risposta utente, il **caso più interessante** riguarda la **gestione delle domande a risposta singola**.

Per estrarre il contenuto infatti, è stata adottata **una combinazione di due strategie**. La tecnica primaria consiste nell'**analizzare le dipendenze della struttura sintattica della risposta**, ricercando dei pattern coerenti alla tipologia di domanda e alla WH-Word a cui fa riferimento (*what, who, when, where, etc.*) In particolare:

- Si effettua un'operazione di Named Entity Recognition per cercare di estrapolare la risposta a domande del tipo "*when*", "*where*", "*who*".

- A partire dalla radice di ciascuna frase, si estrapolano le dipendenze coerenti alla tipologia di domanda. Ad esempio per il tipo "who", si ricercano dipendenze "nsubj", "agent", "nsubjpass", a cui molto probabilmente è associato la risposta cercata.

```
WH_MAPPING = {
    "what": ["attr", "dobj", "obj", "pobj", "acomp"],
    "who": ["nsubj", "agent", "nsubjpass"],
    "when": ["npadvmod", "pobj", "obl"],
    "where": ["pobj", "obl", "prep"],
    "which": ["attr", "dobj"],
    "how": ["advmod", "acomp"]
}

ENTITY_MAPPING = {
    "when": ["DATE", "TIME"],
    "where": ["GPE", "LOC", "FAC"],
    "who": ["PERSON", "ORG"]
}
```

Listing 1: Mapping tra la tipologia di WH Word e le rispettive entità/dipendenze ricercate

Se questa prima operazione non riesce ad estrarre alcuna informazione, si passa ad una **tecnica di riserva, più semplice**, che va a ricercare la risposta all'interno del testo **tramite espressioni regolari** e ricerca di parole.

4.2 Gestione delle domande a risposta binaria

Per le **domande a risposta binaria (vero/falso)**, la strategia più efficiente per estrarre la risposta dell'utente consiste nell'utilizzo di **espressioni regolari** e **matching di parole chiave** all'interno del testo. In particolare, il sistema ricerca la presenza di specifiche keyword indicative:

- **Risposte affermative**, come "true", "correct", "right", ecc.
- **Risposte negative**, come "false", "no", "not true", ecc.

È inoltre previsto un meccanismo per **rilevare risposte ambigue**, ovvero contenenti sia termini positivi che negativi. In questi casi, il sistema richiede all'utente un chiarimento, al fine di evitare interpretazioni errate.

4.3 Gestione delle domande a risposta multipla

Infine, per quanto riguarda le domande a risposta multipla, l'informazione viene estratta confrontando le possibili risposte corrette con il contenuto testuale fornito dall'utente. Il sistema

verifica la presenza di ciascuna risposta attesa all'interno del testo, determinando il numero di risposte corrette individuate.

- La risposta viene considerata **corretta** se l'utente fornisce almeno i **due terzi delle possibili risposte**.
- Se il numero di risposte corrette è insufficiente, il sistema **richiede esplicitamente** all'utente di completare l'elenco. In assenza di una risposta completa, la risposta sarà considerata come errata.

4.4 Estrazione delle informazioni utente

Un'altra applicazione utile di questo modulo, riguarda l'estrazione del nome e di alcuni elementi a partire dalla descrizione iniziale fornita dall'utente durante l'introduzione al colloquio.

Per quanto riguarda **l'estrazione del nome**, è stata creata un'apposita funzione che, **tramite l'analisi delle dipendenze sintattiche**, vada ad intercettare strutture del tipo "I am Andrea", "My name is Andrea" ed estrapolare di conseguenza il nome dell'utente. Si è deciso di implementare una soluzione di questo tipo, e non utilizzare il Named Entity Recognition, per risolvere situazioni ambigue in cui l'utente nomina nella descrizione sia Lara che sé stesso. Quest'ultima operazione però viene utilizzata come strategia secondaria nel caso la prima non riuscisse ad estrapolare correttamente il nome.

A partire dalla descrizione iniziale, un'altra informazione estratta sono gli **aggettivi tramite i quali l'utente si descrive**. Questi risultano molto utili al sistema per **generare risposte altamente realistiche ed inerenti a al contesto!** Vediamo un esempio:

Lara: *Good evening! I'm Lara Croft. You're not here for a tour of my manor — this is an interview, and I don't have time for games. Now, tell me: who are you, and why do you think you can keep up with me?*

User: *Hi Lara, my name is Andrea. I am very brave and intelligent, ideal characteristics for your future assistant.*

Lara: *Brave... That's what they all say, Andrea. Time to prove it. Let's start.*

Lara: *First question, I won't be too evil.*

5 Valutazione

In quest'ultima sezione viene presentata una **valutazione complessiva** del *Dialogue System* implementato. Innanzitutto verranno esaminati alcuni esempi di dialoghi generati dal sistema, successivamente, verrà sottoposto alla *Trindi Tick List*, uno strumento per la valutazione formale di un Sistema di Dialogo, utile a verificare la copertura dei principali componenti teorici e funzionali di un dialogo naturale.

5.1 Esempi ed Analisi di Dialoghi

Dialogo Uno

Lara: Good evening!. I'm Lara Croft. You're not here for a tour of my manor - this is an interview, and I don't have time for games. Now, tell me: who are you, and why do you think you can keep up with me?

User: Hi Lara i am Andrea. The most intelligent and brave assistant that you will have.

Lara: brave... That's what they all say, Andrea. Time to prove it Andrea. Let's start. First question, I won't be too evil: The light of Horus successfully defeated Set.

User: Yes it's true

Lara: Wrong answer, Andrea. Maybe you are not as brave as you think. Missing the first question, Andrea? That's a bad sign-and I don't tolerate weakness. During which years did I make many important archaeological discoveries including finding the Ark of the Covenant?

User: From 1991 to 1996 if i remember well

Lara: Perfect Andrea! The answer is correct! Name the four meteorite artifacts I collected.

User: Ora Dagger and Element 115 for sure

Lara: Good Andrea you have given almost all the correct answers. But i need more to consider this question as correct.

User: mhhh, Eye of Isis and Infada Stone

Lara: Exactly. Seems you're not just lucky-you're capable. Good to know. Ok, we finished. Let me think for a moment

Lara: I'm giving you the chance, Andrea. Now earn it - out there, not here.

Da questo primo esempio possiamo notare diverse caratteristiche e comportamenti interessanti del Dialogue System:

- Nel momento in cui l'utente sbaglia la prima domanda, il sistema risponde in maniera acuta e coerente, riprendendo anche le informazioni date dall'utente nell'introduzione (l'essere "brave").
- Come anticipato, nella gestione di una domanda a risposta multipla, il sistema dopo aver ricevuto una prima risposta, chiede all'utente di completarla, in modo tale da considerarla corretta.

Dialogo Due

Lara: Good morning!. I'm Lara Croft. I've survived tombs, traps, and betrayal. Now I'm looking for someone with fewer skeletons - in closets or otherwise. Let's begin simply: what's your name, and why should I trust you?

User: Hi Lara.

Lara: I deal with ancient secrets, not anonymous applicants. What's your Name?

User: Sorry, my name is Andrea

Lara: Welcome Andrea, let's get started. First question, I won't be too evil: The light of Horus successfully defeated Set.

User: that's false

Lara: Great Andrea! That's the answer I was looking for. Keep this up. I found a backpack from a dead explorer during my Cambodia expedition, correct?

User: also this one it's false

Lara: Incorrect, Andrea, but this isn't one you'd find in a textbook. Where was the Zapadnaya Litsa port located where I infiltrated to find the Spear?

User: I think it is in Latvia

Lara: It's incorrect, Andrea, though I appreciate the effort. That was a tough one. Ok, we finished. Let me think for a moment

Lara: This line of work doesn't allow for hesitation or half-answers. You're not ready, Andrea.

In questo secondo esempio evidenziamo la gestione iniziale del dialogo, in cui il sistema si rende conto che l'utente non ha fornito il proprio nome, e glielo richiede rispondendo a tono.

Dialogo Tre

Lara: Good morning!. Lara Croft speaking. I'm in need of a capable assistant - someone who won't run when things get... complicated. So, before we dive in: who are you, and what makes you think you're the right person for this job?

User: My name is Andrea. I went through many adventures and became a very reliable person.

Lara: A reliable candidate? Well Andrea, this interview will put that to the test. First question, I won't be too evil: Who was the Fiamma Nera gang boss that used the Dagger of Xian's power?

User: Easy, it was Marco Bartoli

Lara: Good Andrea! The answer is correct! Let's continue... I found a backpack from a dead explorer during my Cambodia expedition, correct?

User: yes, it's true

Lara: Great Andrea! That's the answer I was looking for. Keep this up. I was raised in a world of privilege and aristocracy, surrounded by servants and high society, correct?

User: absolutely, i know your story

Lara: Good Andrea! The answer is correct! Ok, we finished. Let me think for a moment

Lara: Well done, Andrea. Now get ready. There's a shattered monastery in the Carpathians waiting for us

5.2 Trindi TickList

1. **L'interpretazione degli enunciati è sensibile al contesto?** Sì, come discusso precedentemente, il sistema è in grado di intercettare diverse situazioni del colloquio e reagire di conseguenza ad una risposta dell'utente. (Risposta ad una domanda difficile, 2 domande di fila sbagliate, etc.)
2. **Il sistema è in grado di gestire le risposte alle domande che forniscono più informazioni di quelle richieste?** Sì, il sistema riesce sempre ad estrapolare la risposta, se esiste, da un testo più articolato.
3. **Il sistema è in grado di gestire le risposte alle domande che danno informazioni diverse da quelle effettivamente richieste?** No, il sistema considera come "errate" risposte di questo tipo.

4. **Il sistema è in grado di gestire le risposte alle domande che forniscono meno meno informazioni di quelle effettivamente richieste?** Sì, questo comportamento si verifica nella gestione delle domande a risposta multipla. In tutti gli altri casi una risposta incompleta è considerata errata.
5. **Il sistema è in grado di gestire risposte ambigue?** Sì, il sistema intercetta risposte contraddittorie o senza informazioni, chiedendo all'utente di rispondere nuovamente.
6. **Il sistema è in grado di gestire informazioni specificate in logica negativa?** No, il sistema richiede sempre una risposta precisa e diretta.
7. **Il sistema è in grado di gestire una domanda senza risposta?** Sì, il sistema riesce ad intercettare una risposta vuota, considerandola ambigua.
8. **Il sistema è in grado di gestire informazioni rumorose?** In generale no, ma se la risposta è chiara ed esplicita, il sistema riesce ad intercettarla anche in un testo rumoroso.
9. **Il sistema è in grado di gestire un dialogo secondario di "aiuto" avviato dall'utente?** No, non sono previsti aiuti.
10. **Il sistema è in grado di gestire un dialogo secondario "non di aiuto" avviato dall'utente?** No, l'utente non può prendere iniziativa per iniziare un sotto-dialogo.
11. **Il sistema pone solo domande follow-up appropriate?** Sì, il sistema mantiene sempre una coerenza ed una linea durante tutto il colloquio. L'introduzione di ciascuna domanda è coerente con il risultato di quella precedente. Inoltre per alcune tipologie di domande il sistema chiede un chiarimento/completamento della risposta.
12. **Il sistema è in grado di gestire informazioni incoerenti?** Sì, l'incoerenza viene rilevata soprattutto nelle domande Vero/Falso, intercettando situazioni in cui l'utente esprime entrambi i risultati.