**MODULO DI INTELLIGENZA ARTIFICIALE**

PROGETTO KEY-KNOWNLEDGE

Sommario

1. **INTRODUZIONE1**
2. **RUOLO DELL’IA1**
3. **DATASET2**
4. **FORMULAZIONE DEL PROBLEMA4**
5. **PSEUDOCODICE DELL’ALGORITMO**5
6. **MOTIVAZIONI**6
7. **PEAS**6
8. **INTRODUZIONE**

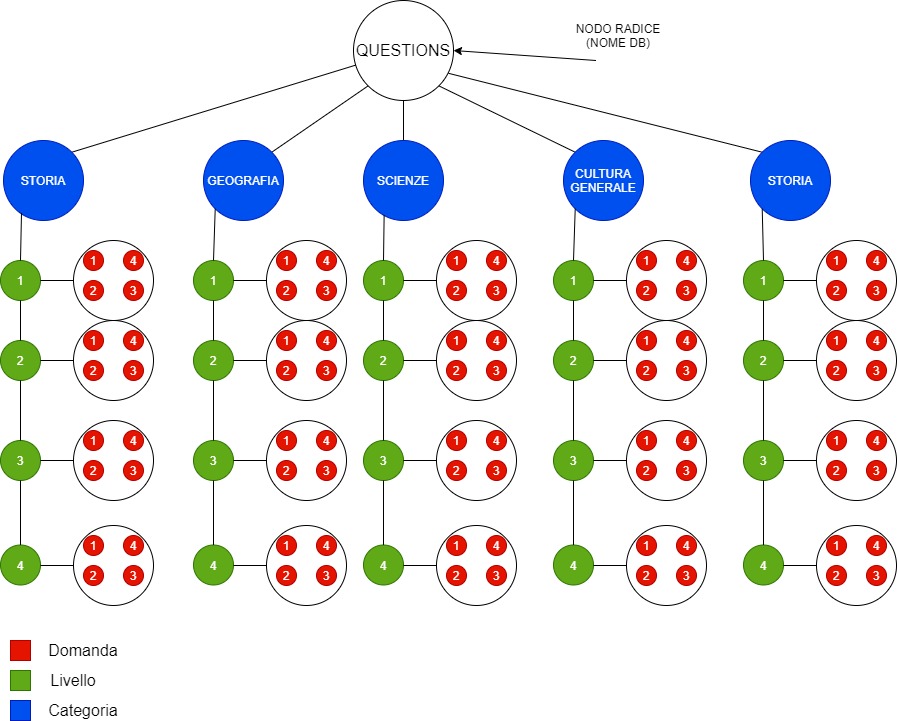
**Il progetto consiste nello sviluppo di un’applicazione Mobile-Android per giocare a un quiz. Il vincitore sarà il giocatore che risponde al maggior numero di domande, generate da un agente intelligente che ha la particolarità di evolversi e adattarsi all’utente. Le domande sono suddivise in categorie (STORIA, GEOGRAFIA, SCIENZE, ARTE, CULTURA GENERALE). Una singola partita è sempre 1v1.**

1. **RUOLO DELL’IA**

**L’obiettivo dell’intelligenza sarà quello di mettere in difficoltà l’utente proponendogli domande sempre più difficili. Ogni volta che l’utente risponde a una domanda, il suo risultato viene usato dall’intelligenza per capire su quali argomenti è più o meno bravo l’utente.**

**ES: Se un utente risponde in maniera corretta a diverse domande di storia sarà meno probabile che l’intelligenza gli proporrà ancora domande di storia.**

1. **DATASET**



1. **FORMULAZIONE DEL PROBLEMA**

* **STATO INIZIALE**: L’utente non ha ancora risposto a nessuna domanda
* **STATI**: Risposta corrente + insieme di risposte date precedentemente.
* **AZIONE:** Fornire una domanda all’utente, di una certa categoria e un certo livello, in base agli stati precedenti.
* **MODELLO DI TRANSIZIONE:** Restituisce la risposta corrente data dall’utente.
* **TEST OBIETTIVO:** L’insieme delle risposte date è pari a 10.

1. **PSEUDOCODICE DELL’ALGORITMO:**
2. **MOTIVAZIONI:**
3. **PEAS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tipo di agente | Misura di prestazione | ambiente | Attuatori | Sensori |
|  |  |  |  |  |
| Selezionatore di domande | Domande di una difficoltà che va in base alle conoscenze dell’utente. | Quiz | Schermo del dispositivo Android | Schermo del dispositivo Android |
|  | Domande di categorie diverse, scelte in base a una probabilità. La probabilità cambia in base alle conoscenze dell’utente. |  |
|  |  |  |  |  |

**INDICATORI DI PERFORMANCE**

**Completezza:** L’albero di ricerca ha uno spazio degli stati con profondità finita. L’algoritmo utilizzato è un algoritmo di ricerca in profondità, ma limitato: Il massimo numero di nodi esplorabili è 10, perché seleziona un massimo di 10 domande. Dunque, posso sostenere che l’algoritmo di ricerca utilizzato è completo, perché termina sempre.

**Ottimalità: Non esiste una soluzione ottima globale dell’algoritmo, ma cerca sempre la soluzione migliore corrente, in base agli stati precedenti.**

**Complessità temporale: La ricerca genererà sempre 10 nodi. Dunque, dato l = limite di nodi esplorabili, la complessità è O(l).**

**Complessità spaziale: Bisogna conservare le risposte date precedentemente insieme ai nodi delle categorie e dei livelli già esplorati. Una volta che un nodo è stato espanso, può essere rimosso dalla memoria non appena tutti i suoi discendenti sono stati esplorati completamente.**

1. **ROPRIETA’ DELL’AMBIENTE**

* **Completamente osservabile**
* **Agente Singolo**
* **Deterministico**
* **Sequenziale**
* **Statico**
* **Discreto**
* **Noto**