Progetto RAG

Lo scopo del progetto è quello di creare un servizio che permetta di caricare documenti (anche di vario formato), per poi interrogare il servizio in linguaggio naturale per ottenere risposte basate su tali documenti.

Questo servizio è costituito da tre componenti:

- Frontend
- Server API
- Database (con possibilità di salvare vettori)

Le 5 principali aree di sviluppo sono:

- Frontend: si occupa ovviamente di interagire con l'utente permettendogli di effettuare operazioni CRUD sui documenti (da discutere Update) e gestire le conversazioni, similmente a ChatGPT
- Parsing documenti (API): queste sono API che ricevono in ingresso un documento, e in base al formato ne estraggono il testo e lo dividono in chunk (sezioni). Quindi accettano in ingresso un form (quello di caricamento dei documenti) e restituiscono in uscita (in realtà inoltrano al servizio successivo) i metadati del documento e una lista di chunk.
- Embedding (API): in questa sezione del server, che riceve in ingresso ciò che restituisce Parsing, viene fatto l'embedding dei chunk e vengono creati tutti gli altri sistemi (es: tags) per permettere query più complesse sui documenti. Lavora a stretto contatto col database: è possibile che sia necessaria più di un'interazione con esso.
- Chat (API): queste API permettono di recuperare le chat di un utente e di continuarle, inviando messaggi e ricevendo risposte. Nel modulo viene gestito, oltre al salvataggio delle chat, il recupero del contesto dal database e le politiche di implementazione dello storico dei messaggi (i messaggi precedenti per garantire contesto). In linea di massima, per la parte di conversazione, in ingresso ci sono i messaggi dell'utente e in uscita le risposte con contesto.
- Database: il database, in una versione minima, deve preoccuparsi di salvare i documenti (quantomeno i metadati e i chunk), le chat e gli utenti. Sul database, di cui è necessario fare la progettazione E-R, vanno poi implementati tutti gli "indici" necessari a recuperare i chunk pertinenti. Lavora a stretto contatto con Embedding (per il salvataggio dei documenti e la loro indicizzazione) e Chat (per il recupero dei documenti pertinenti)

In aggiunta, è richiesto che il progetto sia ben documentato, anche come aiuto ai membri degli altri gruppi, ed eseguibile su Docker.