# Applicazioni Industriale AI

Gruppo 1

Andrea Ciccarello - Gian Marco Simonazzi - Jacopo Arcari

# Obiettivo iniziale

# Obiettivo

Creare uno strumento che:

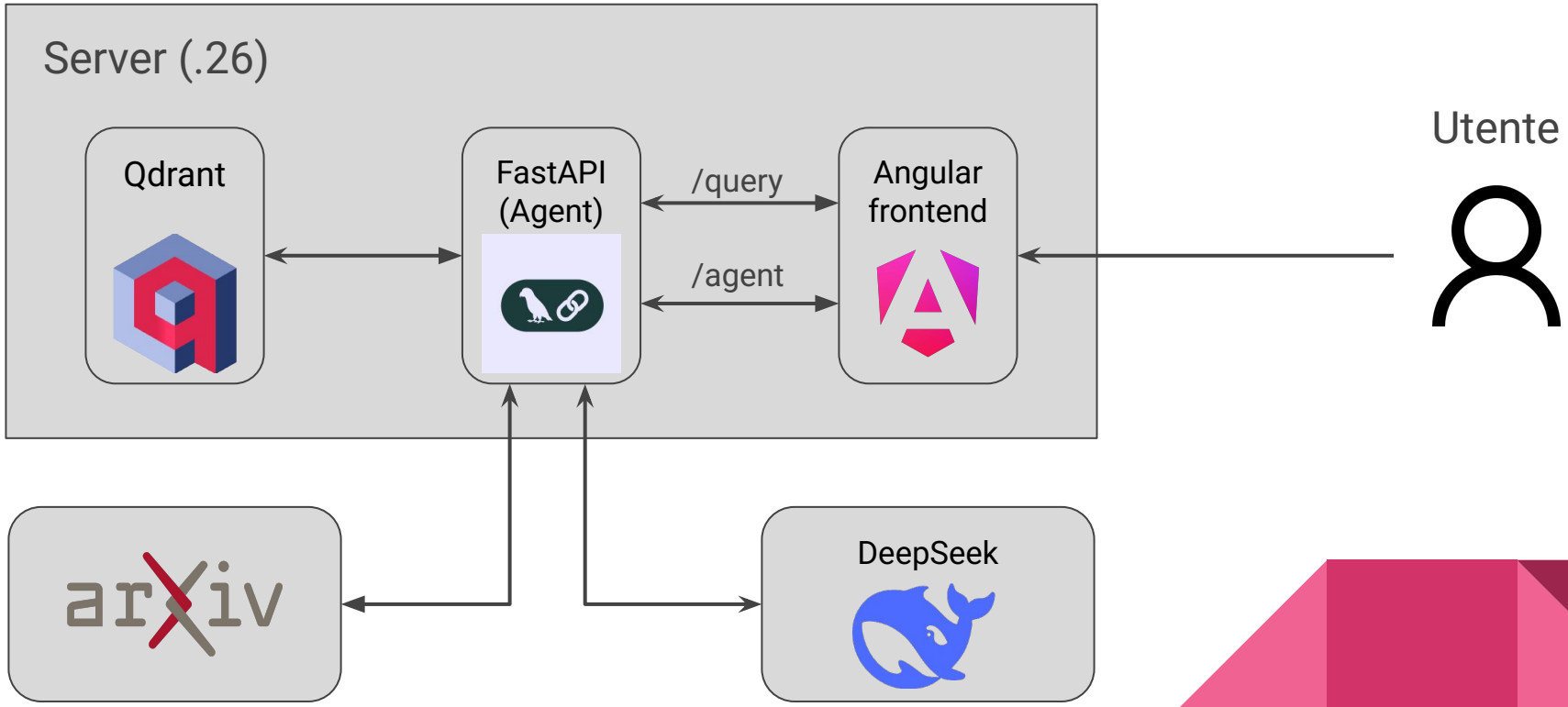
- Cerca degli articoli scientifici relativi ad un argomento richiesto
- Scrive una breve analisi degli articoli nell'ottica dell'argomento richiesto

Il sistema deve fare uso di:

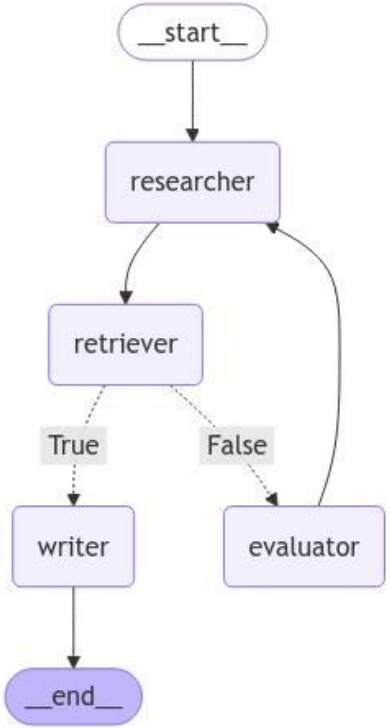
- Vector DB per RAG (in questo caso Qdrant)
- Agenti basati su LLM

# Architettura del sistema

# Architettura generale



# Struttura degli agenti (LangGraph)




- **Researcher**: Costruisce la query per Qdrant
- **Retriever**: Tool di interrogazione di Qdrant
- **Evaluator**: Se la ricerca non ha successo, riscrive la query
- **Writer**: Prende i paper scelti e genera il testo richiesto

# DeepSeek-V3

Motivi per cui è stato utilizzato DeepSeek-V3:

- Supporto ai tools
- Modello pensato per avere un basso costo per Token
- Compatibile con le API di OpenAI
- Ideale per attività complesse, come l'analisi dei paper scientifici

# Limiti della finestra di contesto

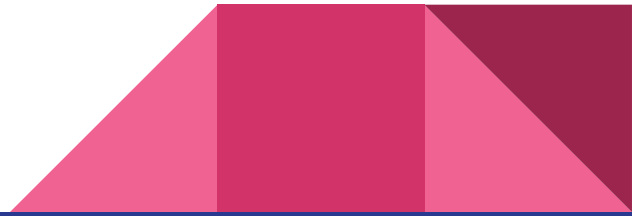
DeepSeek supporta una finestra di contesto fino a **64k token**.

Il writer dovrà inserire in questa finestra:

- La query dell’utente
- Il testo dei paper (top 5)
- Il testo da generare

Il numero di paper processabili in una singola chiamata API è limitato, rendendo necessario eliminare informazioni non essenziali, come la bibliografia.

Se necessario si diminuisce anche il numero di paper.



# Preparazione dati

# Sorgente dei dati e dataset scelto


- Base di partenza: arXiv Dataset (Kaggle)
  Contiene i metadati di tutti i paper di arXiv
  (titolo, autori, data di pubblicazione, categoria, abstract, …)

- Filtro per paper di
  - Categoria: Artificial Intelligence
  - Data di pubblicazione: 2012-2024
- Circa 400k paper

# Embedding e Metadati

Elementi utilizzati per l'embedding:

- Titolo
- Autori
- Abstract

Modello di embedding: *all-mpnet-base-v2*

Metadati associati:

- Titolo
- Autori
- Data di pubblicazione
- Id arXiv
- Categorie arXiv

Per 400k paper, il calcolo richiede ~40 min (1 GPU)

# All-Mpnet-base-V2

Miglior modello di piccole dimensioni a livello di performance.

Caratteristiche:

- Dimensione embedding: 768
- Max Sequence Length: 384 token

# Implementazione

# Implementazione - Qdrant Tool

Input: query, k

1. **Query a Qdrant** -> Top k vettori nella collezione
2. **Query ad arXiv** -> Link al paper e al pdf
3. **Download dei paper** ed estrazione del testo
4. **Pulizia del testo** (rimozione Bibliografia, - 25/40%)
5. Costruzione documento LangChain con testo e metadati
6. Troncamento a 50k token (per context window)

# Implementazione - Writer

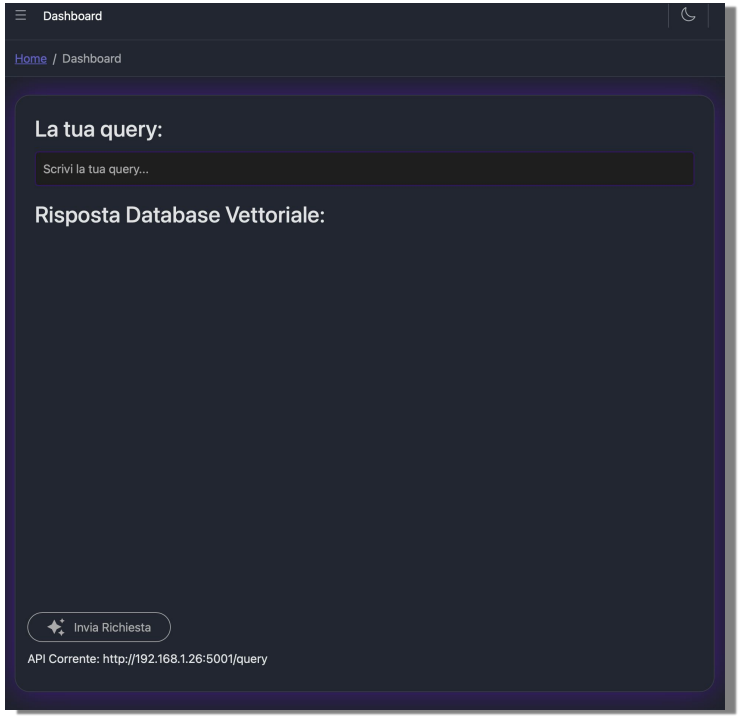
- Costruzione bibliografia dai metadati (procedura)
- Generazione analisi dal prompt fornito:
  - Paper estratti
  - Query utente
  - Panoramica sull’argomento, integrando il materiale nei paper
  - Breve riassunto per paper nel contesto dell’argomento richiesto
  - Istruzione accessorie (no bibliografia,ecc…)

# Possibili miglioramenti?

- Espandere il database su più categorie (al momento solo AI)
- Supportare fonti differenti per i dati oltre ad Arxiv
- Usare un LLM con una finestra di contesto maggiore
- Usare un modello differente per l’embedding
- Chiavi API personalizzate
- Chunking dei PDF

# DEMO

# Per poter provare l’agente




Per poter provare l’agente

- Link sotto VPN

Link Github:

- AndreaCicca/arXiv-vettorizzazione
- giammisimo/arxiv-summary-agent
- AndreaCicca/web-ui-applicazioni-industriali/