



UNIVERSITÀ
DI TORINO

Leveraging deep neural networks for Text-to-SQL translation of microbial resources

Relatori:

- **Marco Beccuti**
- **Luigi Di Caro**

Correlatore:

- **Sandro Gepiro Contaldo**

Candidato:

- **Michele Metta**



Motivazione e Obiettivi

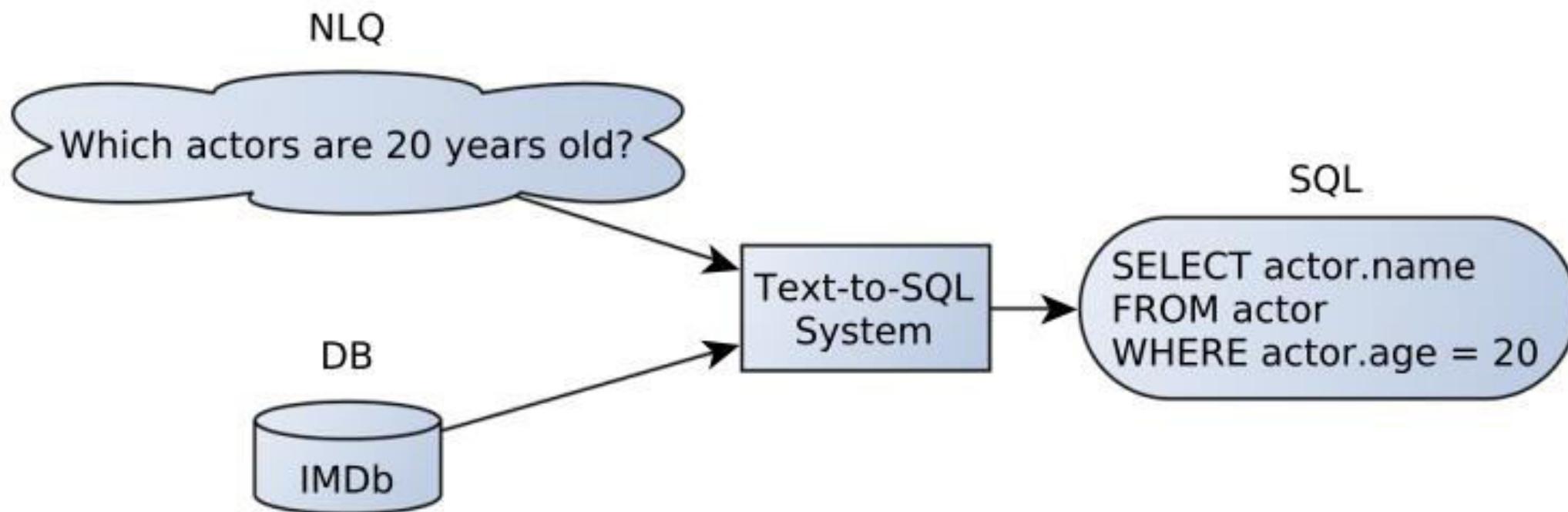


UNIVERSITÀ
DI TORINO

- Facilitare l'accesso ai dati
 - Nessuna conoscenza di SQL e Database
 - Ritrovamento informazioni
-
- ❖ Sperimentazione di modelli Transformer per text-to-SQL
 - ❖ Applicazione web con Next.js come supporto ricerca microbiologica

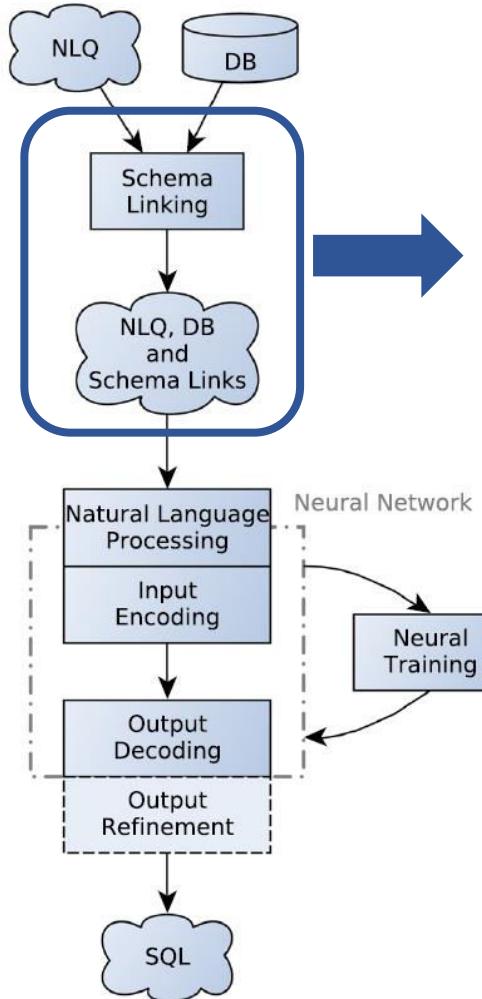
Text-to-SQL

- Formulazione del problema:



- **NLQ (o Question):** Domanda – Frase imperativa dichiarativa

Stato dell'arte – sistema text-to-SQL



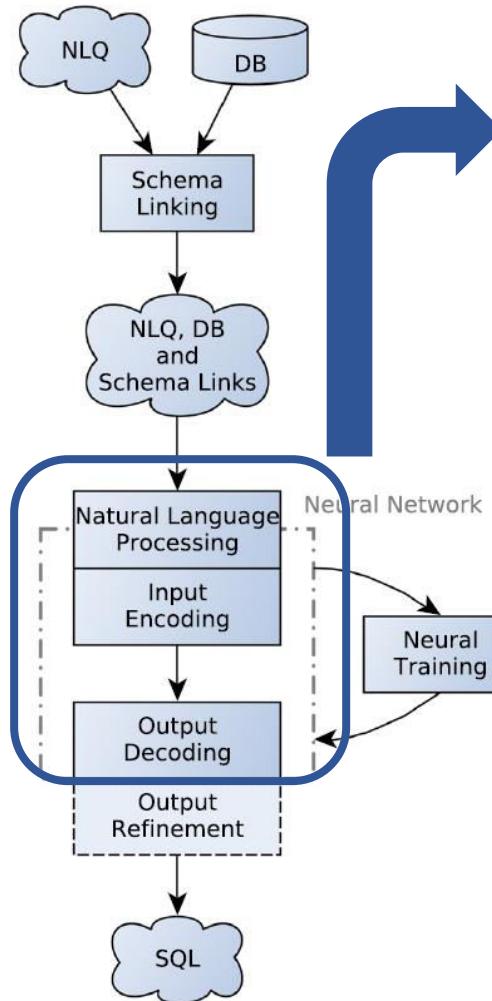
Input NLQ e DB:

1. Individuare elementi della NLQ in riferimento a componenti specifiche del DB:

Approccio utilizzato:

- ✓ **Funzione delegata ai meccanismi di attenzione** (Transformer)
- **"Attention is all you need (2017)"**

Stato dell'arte – sistema text-to-SQL



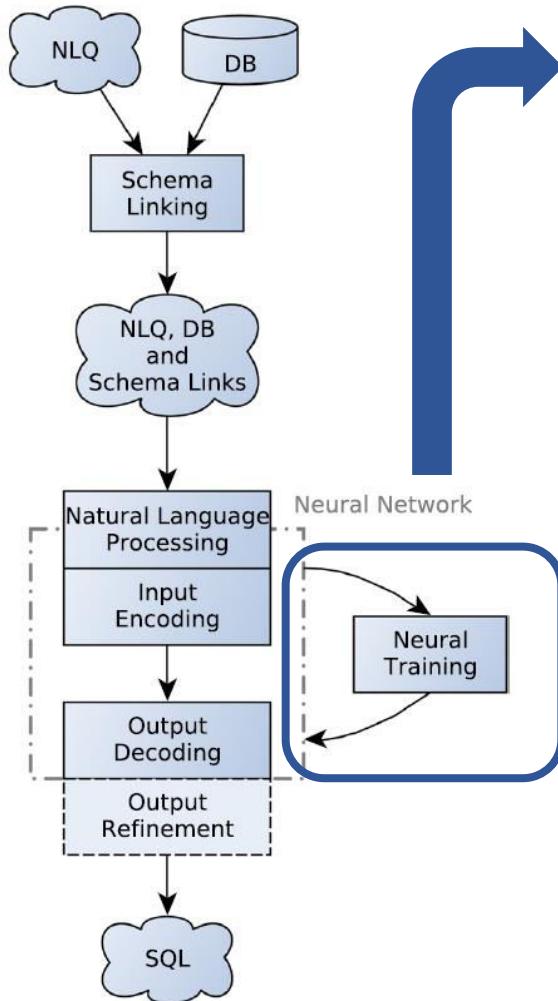
2. Trasformare il linguaggio naturale (formato testo) in una rappresentazione numerica
 - Pre-trained language models (PLMs)

Approccio utilizzato:

- ✓ **Modelli LLaMA – Decoder-only:**

- Input serialization*
- Top-k sampling (Top_k=1, T=1)*

Stato dell'arte – sistema text-to-SQL



3. Scelta della metodologia da applicare per addestrare la rete neurale

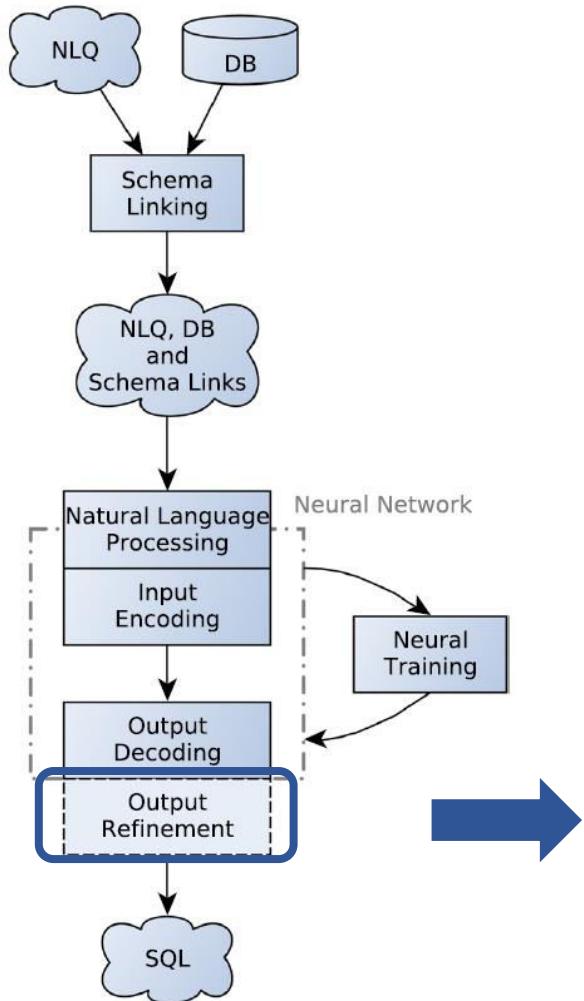
Approccio utilizzato:

- ✓ **Transfer Learning:**
 - Modelli pre-addestrati
 - Fine-tuning

- **Motivazione:**

- Risoluzione di altri task (QA) con aumento delle performance

Stato dell'arte – sistema text-to-SQL



4. "Correggere" query SQL

Approccio utilizzato:

✓ **Fase di Post-processing**

Fase di sperimentazione



UNIVERSITÀ
DI TORINO

LLama Models

- **LLaMA-2-13B**
- **LLaMA-3-8B-Instruct**
- **LLaMA-3-70B-Instruct**

✓ *Interpretare NL*

✗ *Conoscenza profonda DBs o SQL*



Single-FT

- **BioInfoDataset**
 - *conoscenza specifica DB-bio*

Double-FT

- **1° SQL-Create-Context-Instruction**
 - *sintassi e semantica SQL*
- **2° BioInfoDataset**



UNIVERSITÀ
DI TORINO

1° - Dataset Sperimentale: BioInfoDataset

- **Costruito in riferimento al DB-bio**
- Basato su **5 categorie di query**
 - ✓ **C1**: query semplici - su un solo campo e COUNT
 - ✓ **C2**: query con operatori logici - AND e OR su due o più campi
 - ✓ **C3**: query con nomi di colonne e valori scritti in maniera errata
 - ✓ **C4**: query sui campi data
 - ✓ **C5**: query con WHERE, GROUP BY e HAVING.
- **Esempi** generati partendo da **template differenti per ciascuna categoria**
 - ✓ **Template: struttura generica** - coppia: (Question, Query SQL)



BioInfoDataset – Distribuzione degli esempi

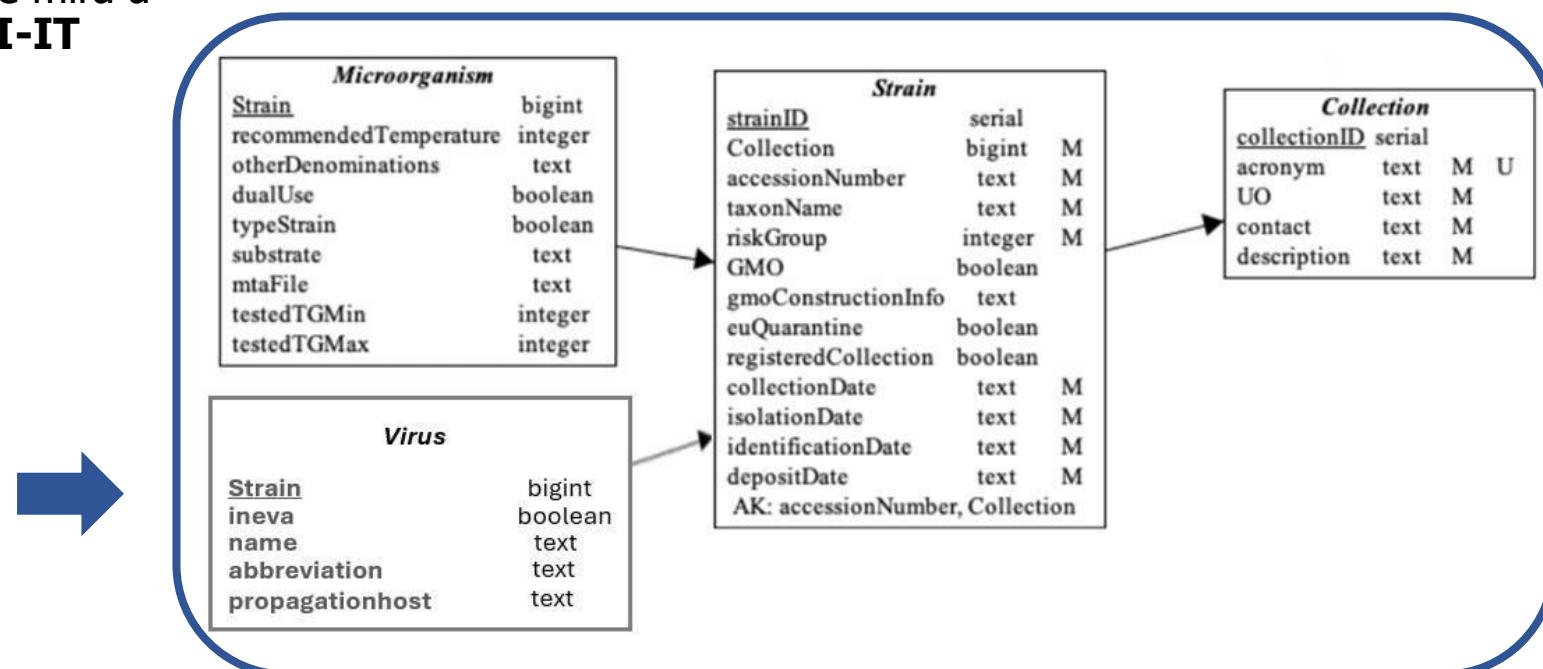
Dataset	Totale	Inglese	Italiano
Training Set	500	250	250
Validation Set	300	150	150
Test Set	300	150	150

1. **Dataset completamente bilanciati:**
 - Lingue (IT e EN) e Categorie
2. **In ciascun dataset – (Question, Query):**
 - Si basano sugli stessi template per entrambe le lingue
 - **Simmetrie garantiscono distribuzione uniforme dei dati**

Categoria	Training Set	Validation Set	Test Set
C1	100 (50 IT, 50 EN)	60 (30 IT, 30 EN)	60 (30 IT, 30 EN)
C2	100 (50 IT, 50 EN)	60 (30 IT, 30 EN)	60 (30 IT, 30 EN)
C3	100 (50 IT, 50 EN)	60 (30 IT, 30 EN)	60 (30 IT, 30 EN)
C4	100 (50 IT, 50 EN)	60 (30 IT, 30 EN)	60 (30 IT, 30 EN)
C5	100 (50 IT, 50 EN)	60 (30 IT, 30 EN)	60 (30 IT, 30 EN)

Database biologico

- Contiene **informazioni sulle collezioni microbiche Italiane**
- Sviluppato durante il *progetto SUS-MIRRI-IT* che mira a **rafforzare l'infrastruttura di ricerca MIRRI-IT**
 - **Nodo Italiano del Consorzio europeo**
- **Contiene: ~39k Tuple**
- ❖ **Intero diagramma logico complesso**
 - **Suddiviso in 7 sottodiagrammi:**
 1. Descrive i **campi** delle tabelle Strain, Microorganism, Virus e Collection

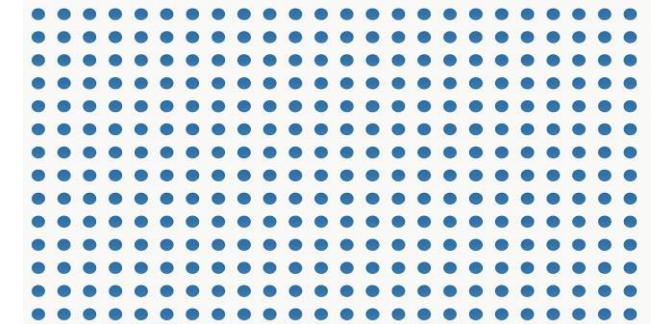


2° - Dataset sperimentale

- **SQL-Create-Context-Instruction**



Hugging Face



✓ Open

✓ Costruito dopo **data-cleaning** e **data-augmentation**
partendo da WikiSQL e Spider

✓ Contiene **78.577** esempi

□ Obiettivo

➤ **Prevenire** – Generazione di nomi errati associati a colonne o tabelle nelle query SQL (**Problema classico**)



Sperimentazione – Fine-tuning & Modelli

Modello	LoRA W^Q, W^V ($r = 64, \alpha = 16$)	Parametri addestrabili (%)	QLoRA (4 bit – NF4)
LLaMA-2-13B	✓	0.40%	✗
LLaMA-3-8B-Instruct	✓	0.34%	✗
LLaMA-3-70B-Instruct	✓	0.19%	✓

❑ LLaMA-3 da 8B e 70B – versione Instruct:

- Migliorare - Interpretazione NL e Coerenza risposte

❑ LLaMA-2-13B:

- Dimensione intermedia

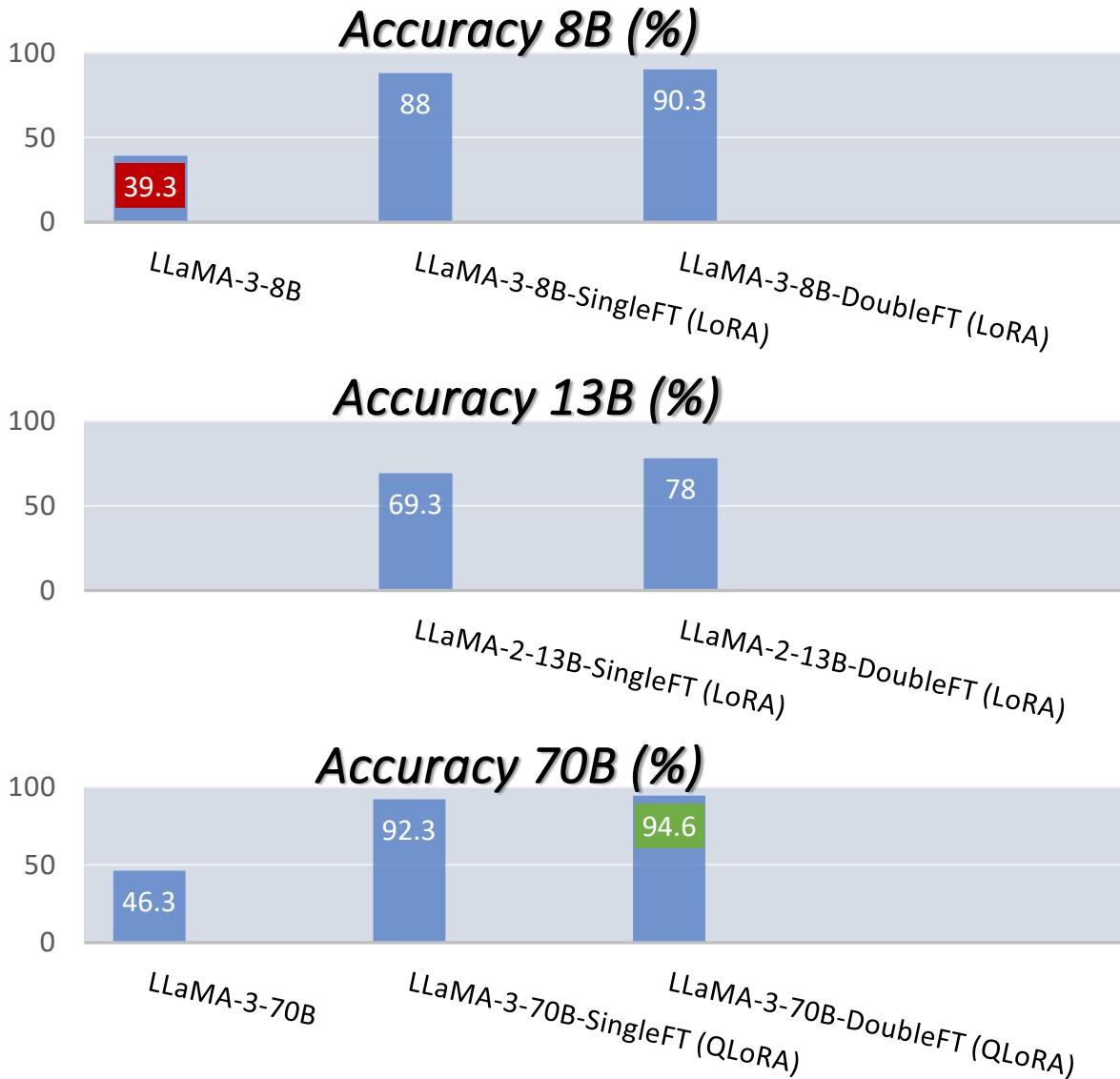
❑ LoRA (Low-Rank Adaptation):

- Riduzione parametri addestrabili

❑ QLoRA (Quantized LoRA) – Necessaria:

- Quantizzazione 70B a 4 bit

Sperimentazione – Risultati Generali



- Accuratezza sintattica media rispetto lingue
-

❖ **Modelli base (8B e 70B)**

- ❖ Non in grado di risolvere il task

• **Modelli LLaMA-3 - SingleFT**

- ✓ Miglioramento rispetto ai base

• **Modelli – DoubleFT rispetto a SingleFT**

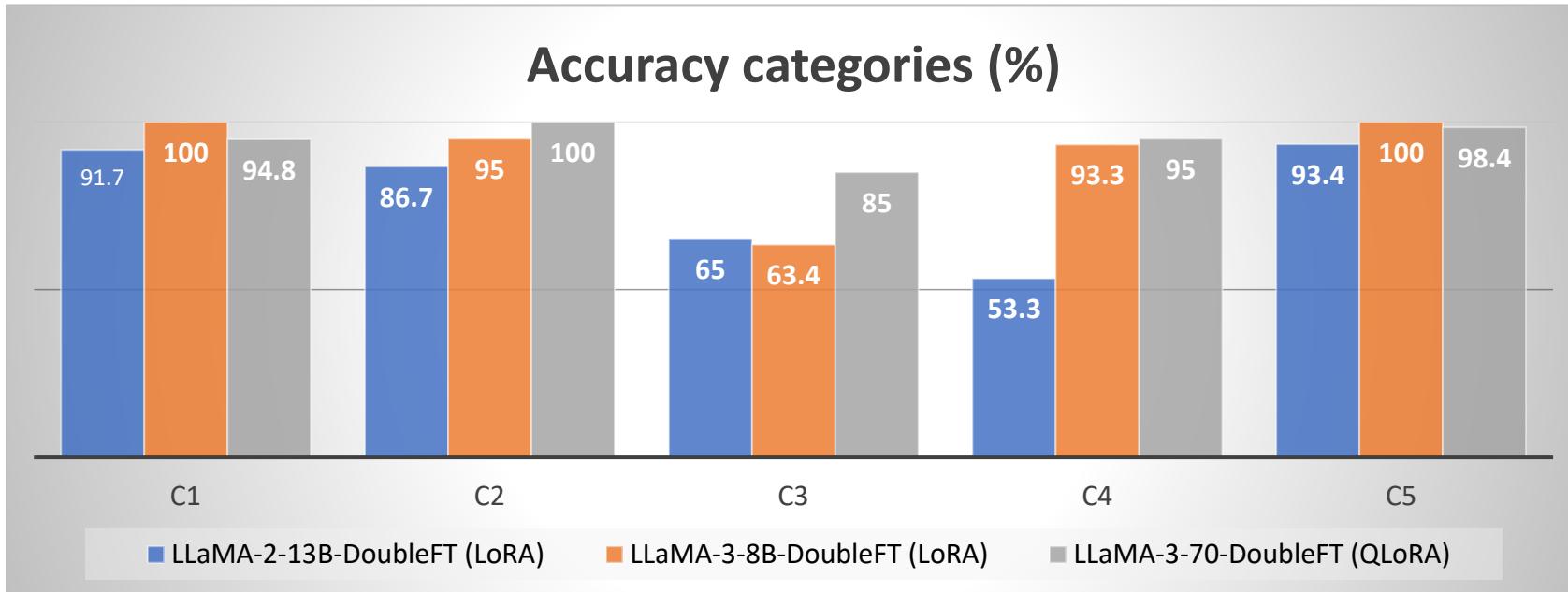
- ✓ Ulteriore incremento
-

□ **Production:**

- ✓ **LLaMA-3-70B-DoubleFT (QLoRA) – Migliore**
- ✓ **LLaMA-3-8B-DoubleFT (LoRA) – Tempi migliori**

❖ **No production:**

- ❖ **LLaMA-2-13B-Double (LoRA)**
 - ✓ Caratteristiche modelli Instruct
 - ❖ Inferiore in generale



- **LLaMA-3-8B-DoubleFT (LoRA) inferiore su C3 rispetto LLaMA-3-70B-DoubleFT (QLoRA):**
 - ❖ **Notevole perdita** accuratezza
 - Parametri inferiori (~9 volte)
 - Meno robusto su **errori** presenti nella **Question**



Post-processing – 3 step principali

1

Sanificazione query SQL

- Contro attacchi SQL-Injection
- LLM generano query dannose

2

Modifica query SQL

- Nomi colonne errati
- Distanza di Levenshtein

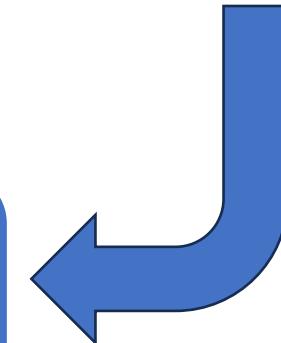
3

Creazione query SQL

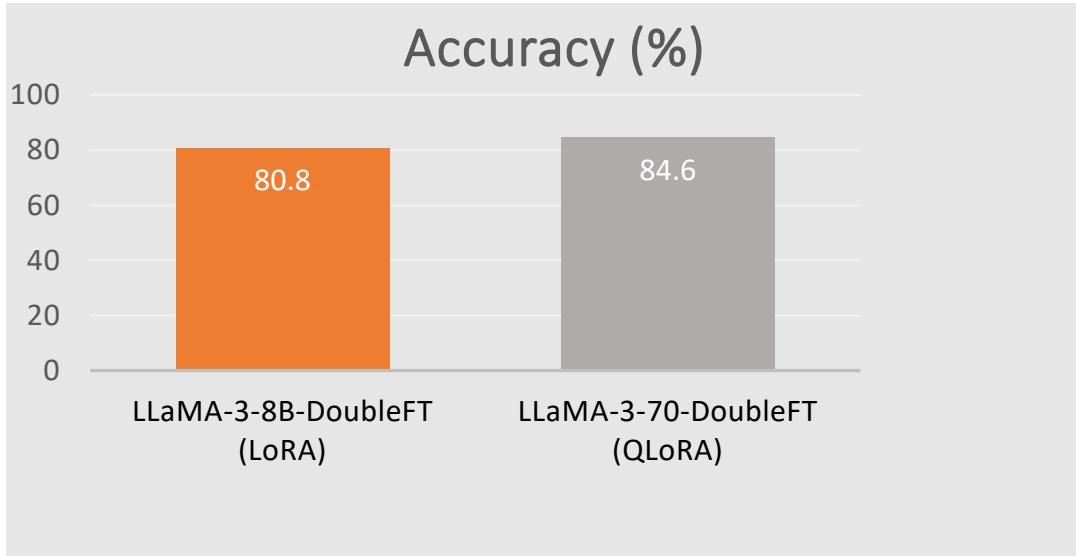
1. Original-query
2. **ILIKE-query**

ILIKE-Query

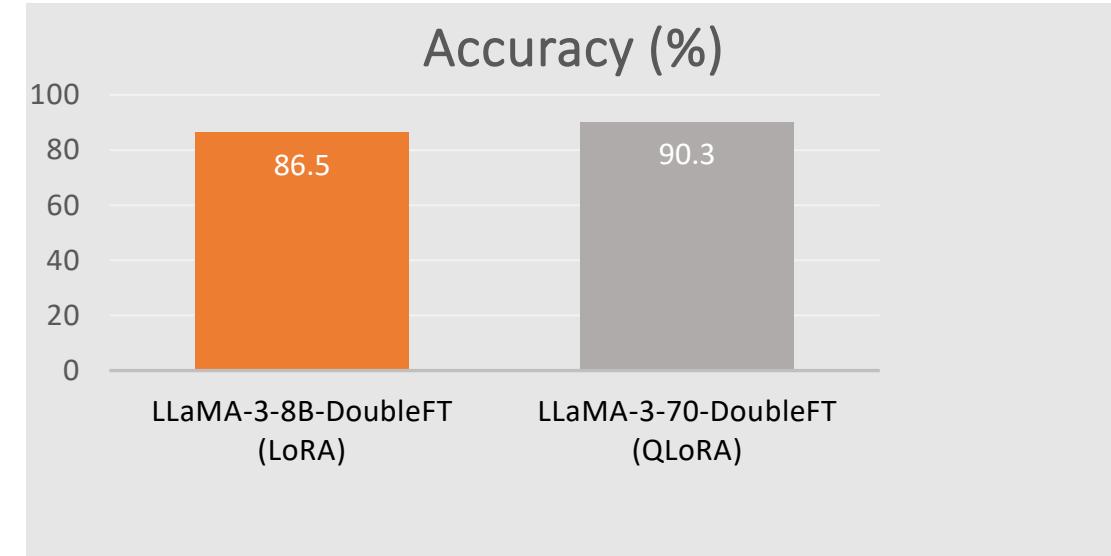
- **Sostituire** [column = 'value'] **con** [column **ILIKE** '%value%']
- No matching esatto



Risultati dataset “In the wild”



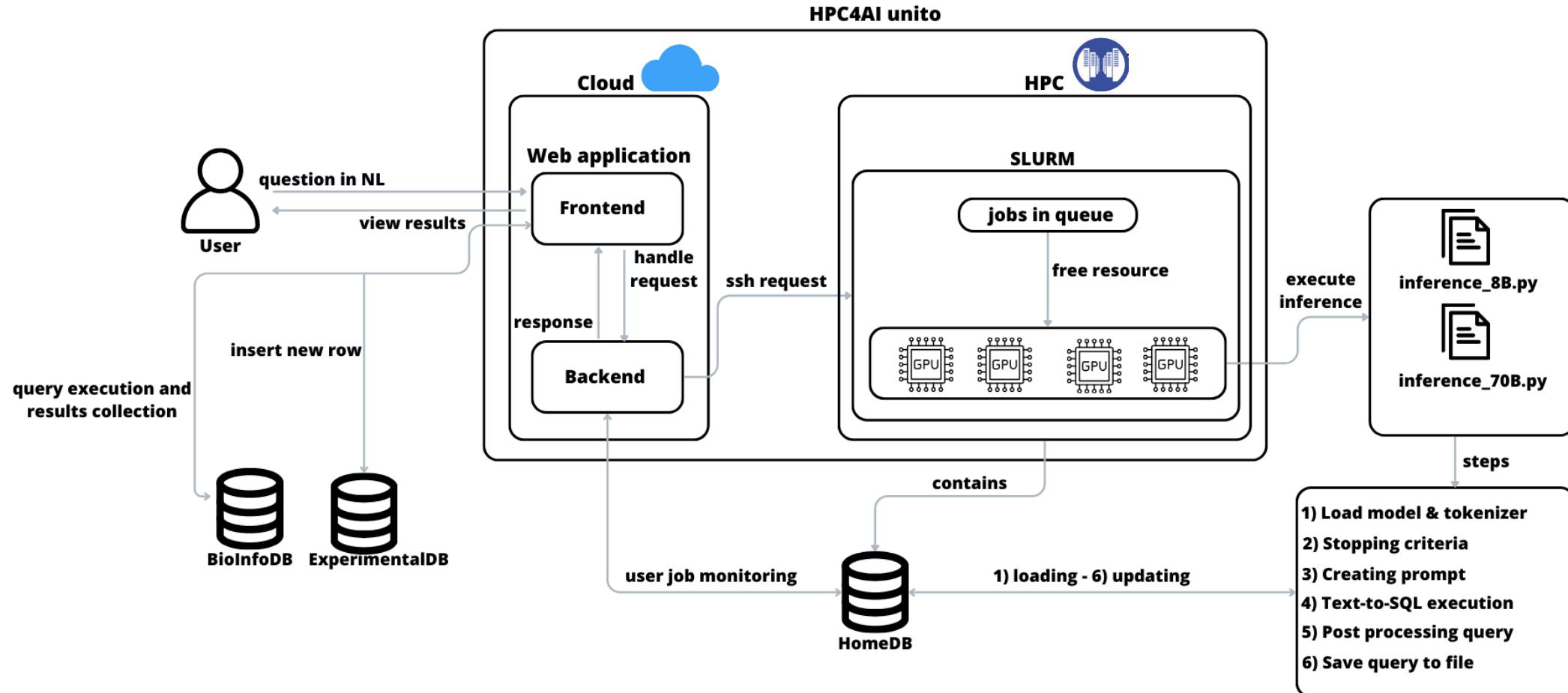
senza Post-processing



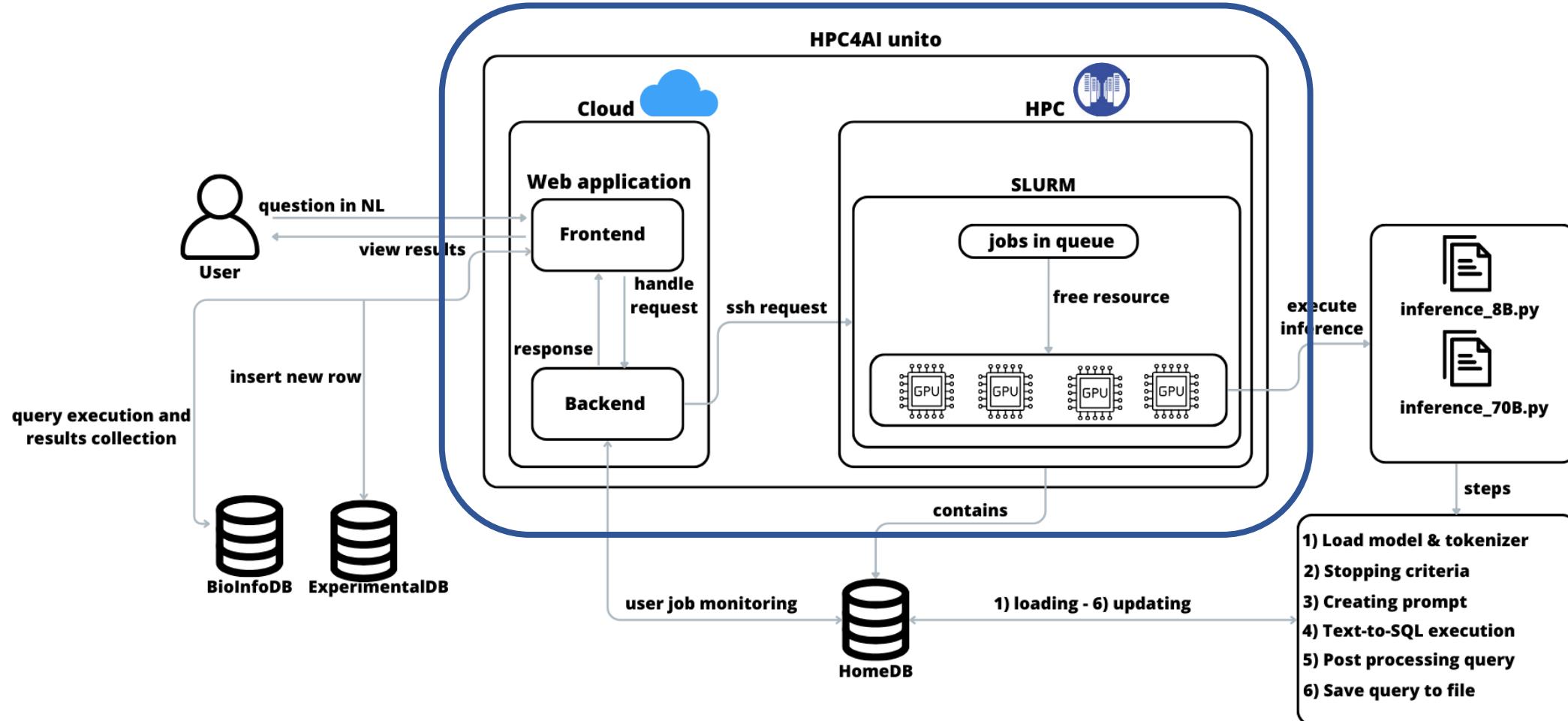
con Post-processing

- Contenente 52 esempi raccolti in production
- ❖ Accuracy inferiore per entrambi rispetto al test set di BioInfoDataset
 - Modello **8B performance vicine al 70B**
- Post-processing: **8B superiore a 70B senza P-p**

Architettura sistema text-to-SQL



Architettura sistema text-to-SQL



- 1° Componente – Web application
- 2° Componente – Gestione inferenze modelli – SLURM



UI – Web application

SQL Query Generator

First select the model to be used during inference, then write the query in natural language (English/Italian) and finally click on the button to generate the SQL query and obtain the results.

Choose LLM model:

Model 8B ▾

Selezione modello

Dammi gli strain con organismstype uguale a bacteria e otherdenomination uguale a mvpc

Question in NL

Generate SQL

SQL query generated:

```
SELECT syntheticstrain.* FROM syntheticstrain WHERE strainid IN (SELECT strainid FROM detailedstrain WHERE organismstype ILIKE '%bacteria%' AND otherdenominations ILIKE '%mvpc%');
```

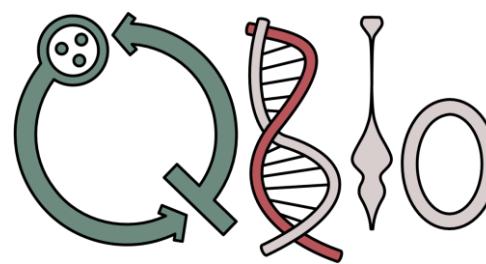
Time taken for response:

20.70 seconds

Query Results (Displayed 100 of 139 results):

collection	accessionnumber	fullaccessionnumber	organismstype	taxonname	strainid
EMCC	0122	EMCC 0122	Bacteria	Burkholderia ambifaria	128
EMCC	0166	EMCC 0166	Bacteria	Burkholderia cenocepacia	172
EMCC	0189	EMCC 0189	Bacteria	Burkholderia cenocepacia	195
EMCC	0106	EMCC 0106	Bacteria	Burkholderia ambifaria	112

Conclusioni e Sviluppi Futuri



UNIVERSITÀ
DI TORINO

- Obiettivi raggiunti

- **Sperimentazione:** Fine-Tuning modelli LLama su text-to-SQL
- **Implementazione:** Applicazione web supporto alla ricerca microbiologica

- Limiti principali - Applicazione attuale

1. **Limite architetturale - Nodi gestiti da SLURM**

- ❖ Indisponibilità temporanea nodi
- ❖ Caricamento modello in VRAM - 30s modello 8B e 70s modello da 70B
 - <7s in media su categorie

- ✓ **Soluzione:** Server dedicato

2. **Limite modelli - Mancata conoscenza valori DB**

- ✓ **Aggiunta valori nei prompt**

- ❖ Problema: Finestra contesto

- ✓ **Implementazione** algoritmo basato su un **Meccanismo di sostituzione**:

- i. Costruire un dizionario – (K: valori, V: colonna più probabile)
- ii. Step di sostituzione – (mapping colonna errata con quella più probabile)

- ✓ **Aggiunta nuovi template - BioInfoDataset** – Enfatizzare caratteristiche con più esempi

- ✓ **Testing di altri modelli:** Open weights o API per rapporto costi-benefici



**UNIVERSITÀ
DI TORINO**

Grazie



UNIVERSITÀ
DI TORINO

RIFERIMENTI

- A survey on deep learning approaches for text-to-SQL. URL: <https://link.springer.com/article/10.1007/s00778-022-00776-8>
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, Illia Polosukhin. "Attention is All You Need." *Advances in Neural Information Processing Systems*, 2017. URL: <https://arxiv.org/abs/1706.03762>
- IMG Hugging Face: <https://huggingface.co/brand>
- IMG Data cleaning: <https://datacleaningservices.com/data-cleaning-for-government-agencies/>



**UNIVERSITÀ
DI TORINO**

Materiale di Supporto

Vista – Detailedstrain

- In fase di sperimentazione, è stata utilizzata una **vista**, del **database** appena descritto, chiamata "**Detailedstrain**".
- Contenente **73 colonne** che rappresentano le **informazioni più rilevanti** per un generico strain.
- Vantaggi:**
 - ✓ *Prompt semplice e compatto, senza riferimenti a tabelle differenti*
 - ✓ *Minor sovraccarico con informazioni di contesto (nomi tabelle, chiavi primarie, riferimenti esterni, ecc..) per i modelli di linguaggio*



strainid	collection	accessionnumber
gmo	gmoconstructioninfo	registeredcollection
collectiondate	isolationdate	identificationtechnique
depositdate	otherdenominations	typestrain
inclusiondate	remarks	pathogenicity
beforenagoya	availablefordis	otherccnumbers
commentontaxonomy	mutantinformation	userrestrictions
nagoyaconditions	geoorigin	cacronym
confirmed	organismtype	riskgroup
euquarantine	strain	recommendedtemperature
dualuse	substrate	testedtgmin
testedtgmax	organismtype	sexualstate
mtafile	prodofmetabolites	genus
species	meuquarantine	isolationhabitat
status	infrasubspecificnames	qps
axenic	genotype	mriskgroup
historyofdeposit	enzymeproduction	applications
plasmids	plasmidscollectionsfields	ploidy
interspecifichybrid	scientificname	ineva
inmirri	othernames	isolationhost
name	abbreviation	propagationhost
cultivar	lyticcicle	pathotypeserotypetype
storageconditions	symptomatologyisolationhost	transmissionby
contamination	infectivitytested	restrictions
nagoyaproto		



1° - Dataset sperimentale

▪ **SQL-Create-Context-Instruction** (open su Hugging-Face)

- Basato su *WikiSQL* e *Spider*
- *WikiSQL*:
 - ✓ **80k coppie di QNL e Query SQL** raccolte tramite crowd-sourcing
 - ❖ **Complessità query bassa**: query riferimento singola tabella e NO clausole complesse
- *Spider*:
 - ✓ **Annotato a mano** da 11 studenti di Computer Science madrelingua Inglese – Università di Yale
 - ❖ Contiene circa **10k domande in NL, 5k query SQL**
 - ✓ Query su **4 livelli di difficoltà**: facile, medio, difficile, estremamente difficile
 - ✓ **Tutti i costrutti del linguaggio SQL** compreso la nidificazione



UNIVERSITÀ
DI TORINO

1° - Dataset sperimentale

▪ **SQL-Create-Context-Instruction**

- **Obiettivo**

- **Specializzare** un LLM sul task text-to-SQL – **prevenire** generazione errata nomi colonne e tabelle (Problema classico)

- ✓ Costruito dopo **data-cleaning** e **data-augmentation** sui dataset WikiSQL e Spider
 - ✓ **Contiene 78.577 esempi**

- **Utilizzo**

- ✓ **Fine-tuning** per apprendere una **rappresentazione migliore e più completa linguaggio SQL**

SQL-Create-Context-Instruction: *Struttura esempi*



UNIVERSITÀ
DI TORINO

PROMPT

[INST] Instruction/context [/INST] Question-QuerySQL

- Seguono lo stile prompt dei modelli Llama
- 3 componenti principali



UNIVERSITÀ
DI TORINO

SQL-Create-Context-Instruction

PROMPT

[INST]

Instruction/context

[/INST]

Question-QuerySQL

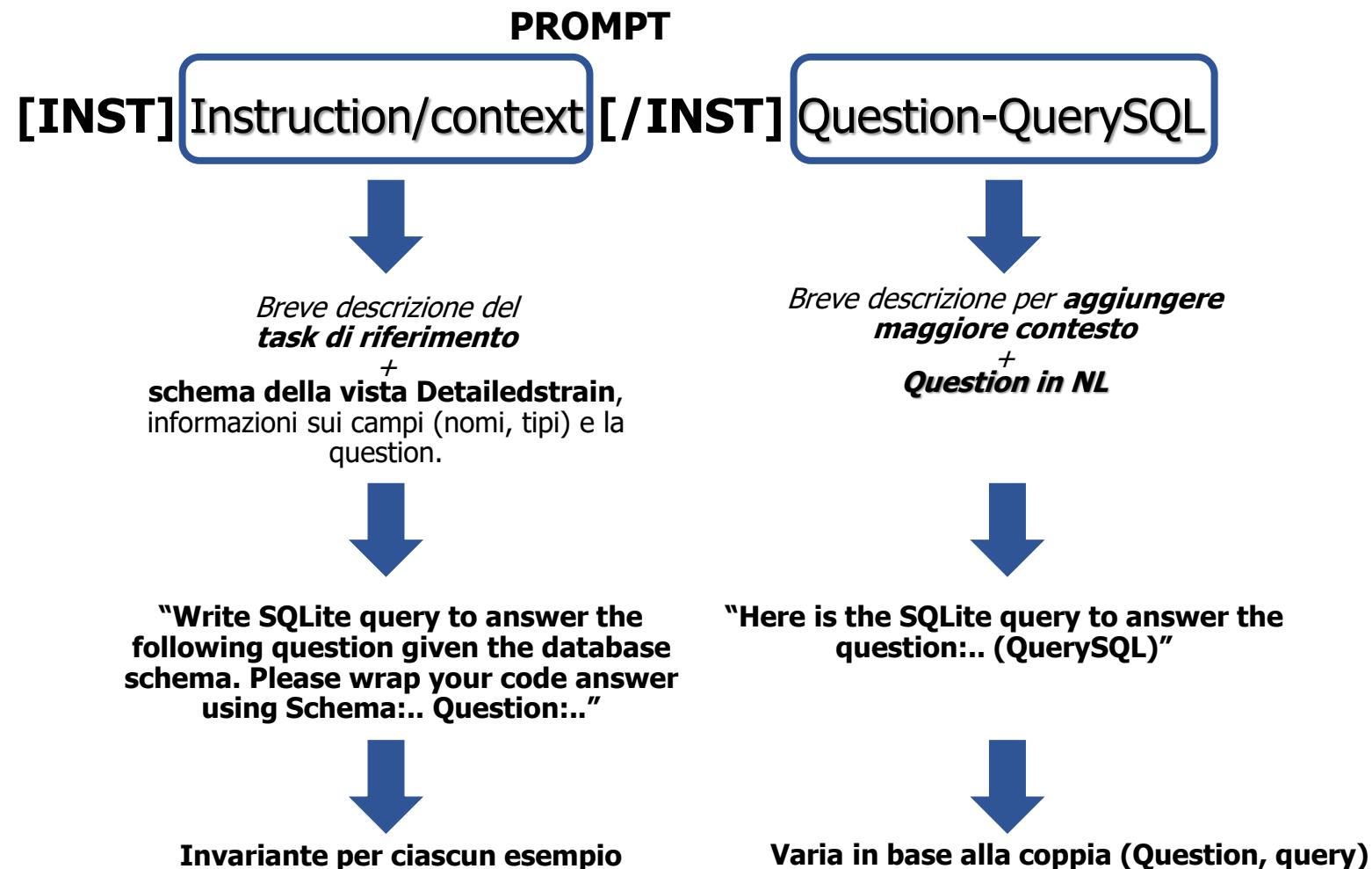


Delimitatori



UNIVERSITÀ
DI TORINO

SQL-Create-Context-Instruction





UNIVERSITÀ
DI TORINO

SQL-Create-Context-Instruction: Fine-tuning

PROMPT

[INST] Instruction/context [/INST] Question-QuerySQL



Input

Llama model (Decoder-only)

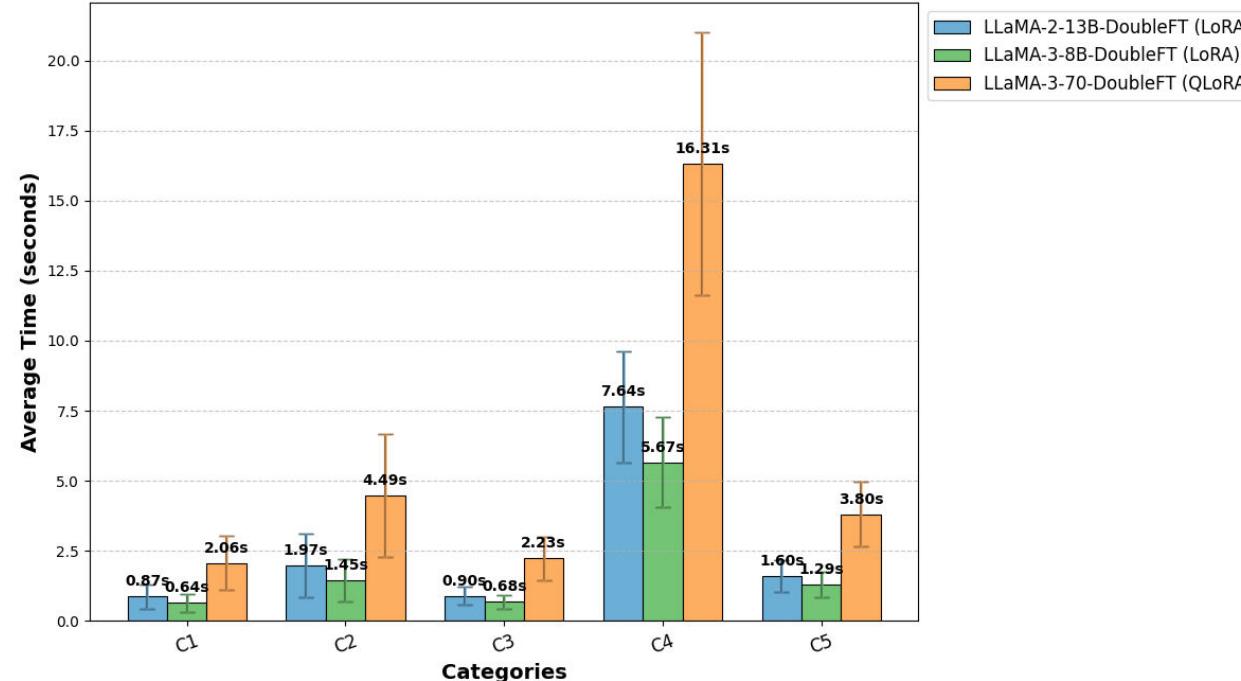


Generazione query SQL

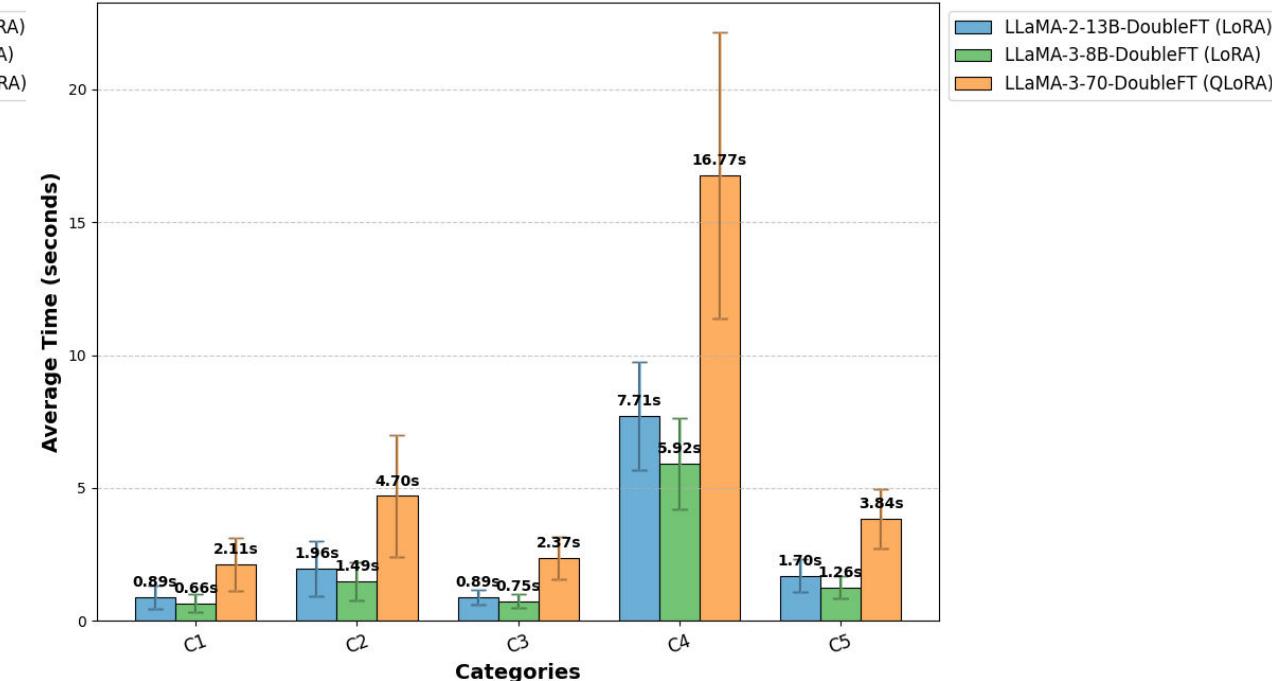
SELECT ...

Sperimentazione – Distribuzione tempi di inferenza

Average Inference Time per Category with std Deviation (Comparison of 3 Models) - English



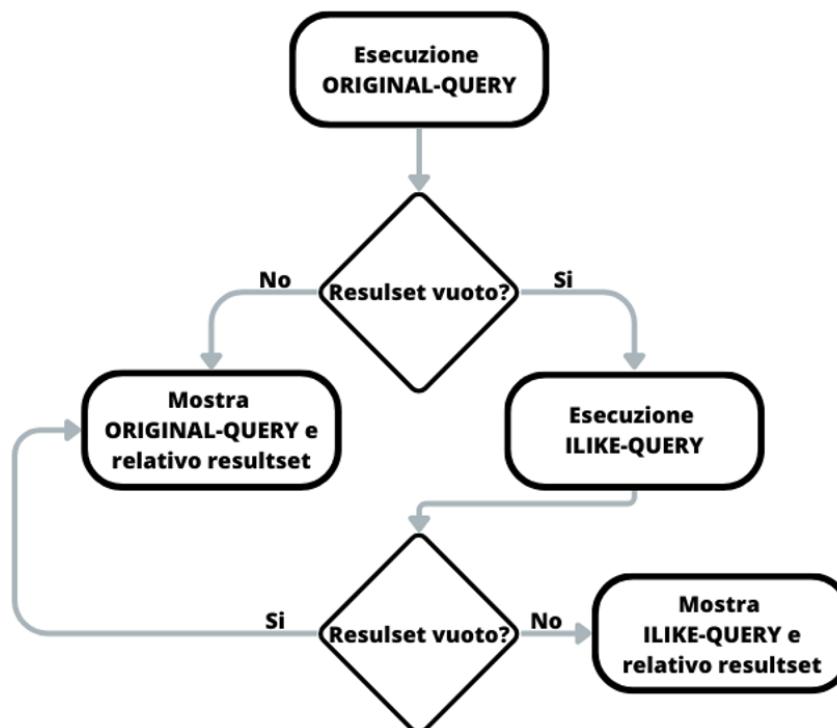
Average Inference Time per Category with std Deviation (Comparison of 3 Models) - Italian



- Tempi simili per entrambe le lingue
- Per tutti i modelli:
 - ✓ Tempi al di sotto dei 7s per quasi tutte le categorie
 - ✓ Categoria C4 richiede tempi maggiori – Causa: generazione numero maggiore di token per query (Date)
 - ✓ Dev. std > 0: complessità variabile query interne a ciascuna categoria

Post-processing

- **Fase rilevante** (script inferenza) per migliorare le performance dei modelli in production – Alcuni step:
 1. Sanificazione query SQL (contro **attacchi SQL-injection**)
 2. Modifica di eventuali nomi errati per le colonne tramite **distanza di Levenshtein** (Fuzzy/approximate string matching – stato dell’arte task)
 3. Creazione di due query SQL:
 - **Original-Query:** generata dal modello
 - **ILIKE-Query:** sostituisce ogni espressione [column = 'value'] con [column ILIKE '%value%'] (NO matching esatto - valore come sottostringa)



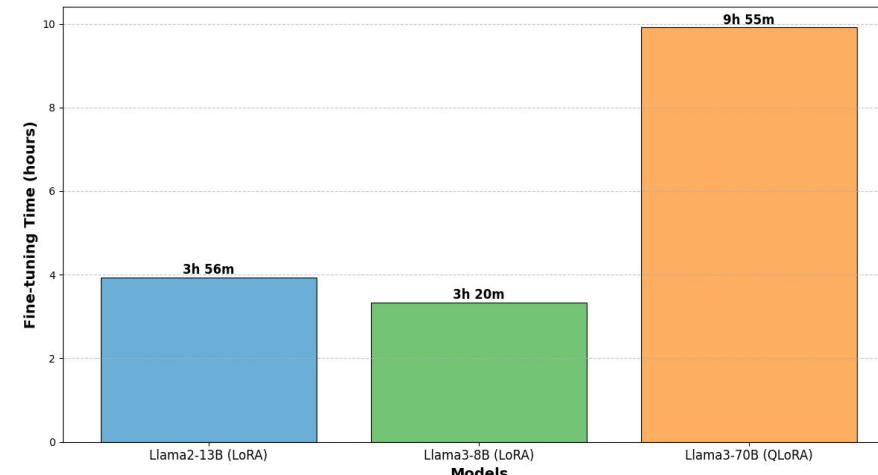
4. Le due query saranno **innestate**:
 - In questo modo la query principale potrà fare riferimento ad un’altra **vista** chiamata **Syntheticstrain**
 - **Syntheticstrain:** contiene un numero *limitato* di colonne ma è preferita dai biologi per la visualizzazione dei risultati

Sperimentazione – Occupazione memoria fine-tuning

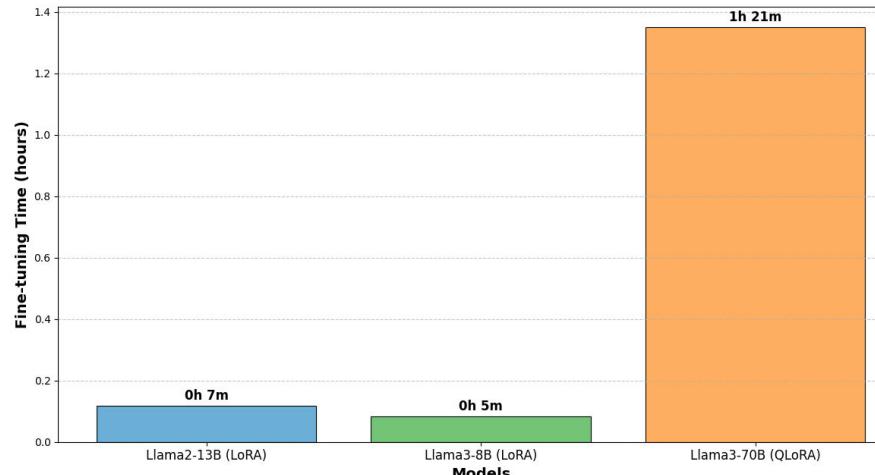


UNIVERSITÀ
DI TORINO

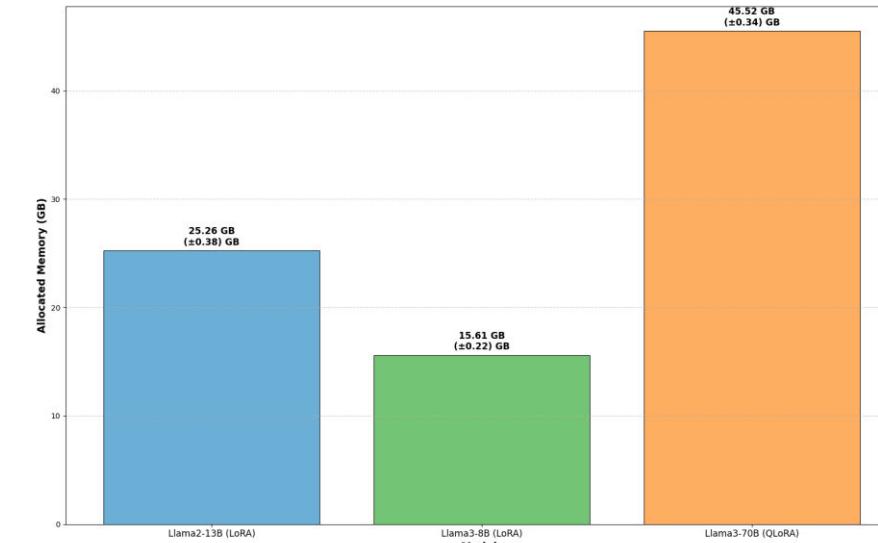
Total time of first fine-tuning per model (Large dataset: SQL-Create-Context-Instruction)



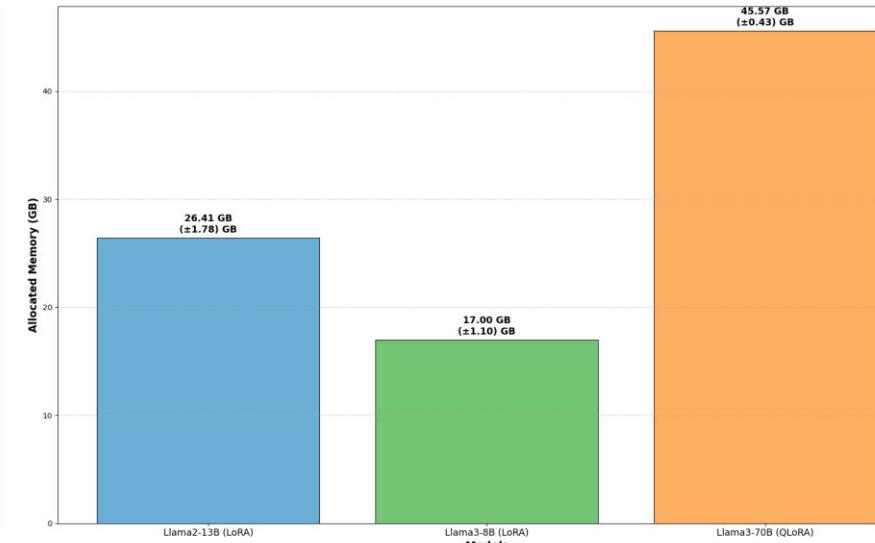
Total time of second fine-tuning per model (Small dataset: BioInfoDataset)



Average Allocated Memory per Model (with Std Deviation) during first fine-tuning (Large dataset: SQL-Create-Context-Instruction)



Average Allocated Memory per Model (with Std Deviation) during second fine-tuning (Small dataset: BioInfoDataset)



Tempi di addestramento:

- ❖ Primo fine-tuning (sx) - tempi maggiori - (1 epoca)
- ✓ Secondo fine-tuning (dx) - Tempi inferiori - (4 epoche per 13B e 8B – 5 epoche 70B)
- Dimensioni dataset differenti

Occupazione memoria:

- ✓ Occupazione di memoria simile per entrambi i fine-tuning
- ✓ Il Modello da 70B quantizzato richiede intorno ai 45 GB di VRAM (160 GB iniziali - risparmio di 115 GB di memoria)