# Homework #3 - Neo4j

*Analisi della Rete di Collaborazione Scientifica su arXiv*

## Indice

# 1. Introduzione

Questo elaborato presenta l'analisi di un dataset contenente articoli scientifici pubblicati su *arXiv* nel dominio dell'Intelligenza Artificiale. L'obiettivo principale è stato l'implementazione di un grafo di proprietà in Neo4j, un database orientato ai grafi ottimizzato per la gestione di relazioni complesse tra entità.

Il dataset originale, denominato *Artificial Intelligence Publication Trends*, contiene 8.000 articoli scientifici con informazioni dettagliate su titolo, abstract, data di pubblicazione, autori e link alla pagina arXiv. Inoltre, sono stati forniti file aggiuntivi contenenti l'associazione di ogni paper ai topic estratti tramite **BERTopic**, una tecnica di topic modeling basata su embedding semantici.

# 2. Creazione del Grafo

## 2.1 Struttura dei Nodi

Il grafo è stato modellato con tre tipologie di nodi, ciascuna rappresentante un'entità distinta del dominio:

| Tipo Nodo | Proprietà | Descrizione |
|---|---|---|
| **Paper** | url, title, abstract, date | Articolo scientifico su arXiv |
| **Author** | name | Autore di articoli scientifici |
| **Topic** | id, name | Topic estratto da BERTopic |

**Paper:** Rappresenta un articolo scientifico pubblicato su arXiv. Le proprietà includono l'URL univoco (usato come identificatore), il titolo, l'abstract (troncato a 5000 caratteri per efficienza), e la data di pubblicazione (estratta dal timestamp originale).

**Author:** Rappresenta un autore di articoli scientifici. Il nome completo è usato come identificatore univoco. Gli autori sono estratti dal campo 'authors' del dataset, separati da virgola.

**Topic:** Rappresenta un topic tematico estratto dagli abstract tramite BERTopic. Ogni topic ha un ID numerico e un'etichetta descrittiva (es. '0_neural_network_deep_learning'). Sono stati identificati 132 topic distinti.

## 2.2 Tipologie di Relazioni

### Relazione AUTHORED

Collega ogni nodo **Author** ai nodi **Paper** di cui è autore o co-autore. La direzione semantica va dall'autore al paper: `(Author)-[:AUTHORED]->(Paper)`. Questa relazione permette di navigare dalla prospettiva dell'autore verso le sue pubblicazioni.

### Relazione ON_TOPIC

Collega ogni **Paper** ai suoi **top-3 Topic** più rilevanti, come determinato da BERTopic. La relazione include la proprietà **probability**: valore float [0,1] che indica la probabilità che il topic rappresenti il contenuto dell'abstract

Questa proprietà è fondamentale per il calcolo della similarità pesata tra paper.

### Relazione COAUTHORED_WITH

Relazione **non direzionale** tra coppie di **Author** che hanno collaborato agli stessi paper. Include la proprietà **papers_count**: numero di paper che i due autori hanno scritto insieme.

Data l'elevata cardinalità (oltre 1.4 milioni di relazioni), è stata utilizzata la procedura `apoc.periodic.iterate` (con il plugin APOC) per processare le coppie in batch, evitando timeout e problemi di memoria (che altrimenti lo rendevano impossibile).

### Relazione SIMILAR_TOPIC

Relazione tra **Paper** che condividono topic simili. La similarità è calcolata tramite la **Similarità di Jaccard Pesata (o Similarità di Ruzicka)**, che tiene conto delle probabilità di assegnamento:

$$simT(p, q) = \Sigma_{i \in UT} \, min(p_i, q_i) \, / \, \Sigma_{i \in UT} \, max(p_i, q_i)$$

dove $p$ e $q$ sono due articoli, $p_i$ è la probabilità che il topic $i$ sia assegnato al paper $p$, e $U_T$ è l'unione dei topic assegnati a $p$ e/o $q$.

**Soglia di filtraggio:** sono state create solo relazioni con similarità > 0.3, per ridurre il numero di archi e mantenere solo connessioni significative.

## 2.3 Vincoli e Indici

Per garantire l'integrità dei dati e ottimizzare le performance delle query, sono stati creati vincoli di unicità su ciascun tipo di nodo:

1. `CREATE CONSTRAINT FOR (p:Paper) REQUIRE p.url IS UNIQUE`
2. `CREATE CONSTRAINT FOR (a:Author) REQUIRE a.name IS UNIQUE`
3. `CREATE CONSTRAINT FOR (t:Topic) REQUIRE t.id IS UNIQUE`

## 2.4 Processo di Caricamento

Il caricamento dei dati è stato effettuato in fasi sequenziali per garantire l'integrità referenziale:

1. **Creazione nodi Topic**: 132 nodi creati dal file topic_definitions.csv
2. **Creazione nodi Paper e Author**: processati in batch da 500 record, con creazione simultanea delle relazioni AUTHORED
3. **Creazione relazioni ON_TOPIC**: per ogni paper, collegamento ai top-3 topic con relative probabilità
4. **Creazione relazioni COAUTHORED_WITH**: tramite APOC periodic iterate
5. **Creazione relazioni SIMILAR_TOPIC**: processate in batch confrontando coppie di paper

**Statistiche finali del grafo:**

| Elemento | Conteggio |
|---|---|
| Vincoli | 3 |
| Nodi Paper | 8,000 |
| Nodi Author | 37,604 |
| Nodi Topic | 132 |
| Relazioni COAUTHORED_WITH | 2,844,978 |
| Relazioni AUTHORED | 44,460 |
| Relazioni ON_TOPIC | 24,000 |
| Relazioni SIMILAR_TOPIC | 1,393,172 |

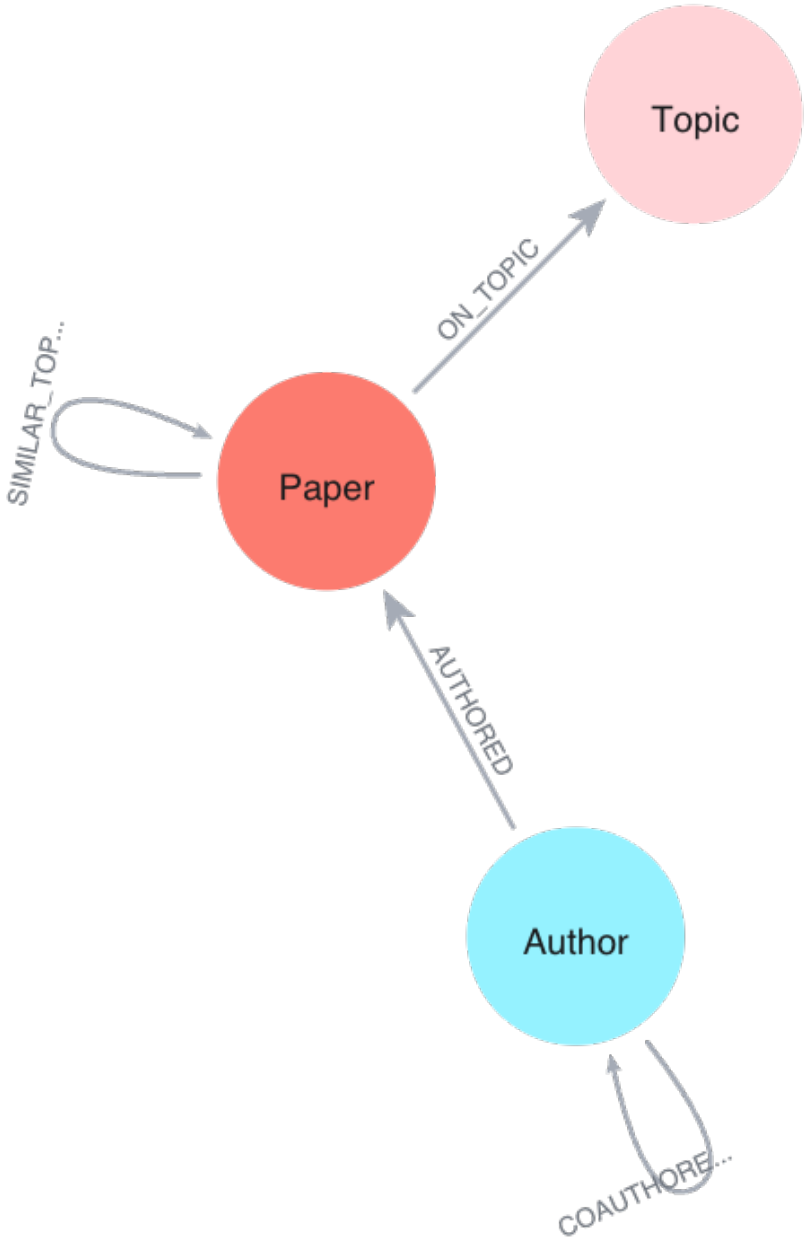
# 3. Interrogazioni Analitiche

## 3.1 Query 1: Schema del Database

Lo schema del database può essere visualizzato in Neo4j Desktop tramite il comando:

```
CALL db.schema.visualization()
```

Lo schema mostra una struttura a stella centrata sui nodi Paper, che fungono da hub connettendo autori (tramite AUTHORED) e topic (tramite ON_TOPIC). Le relazioni COAUTHORED_WITH e SIMILAR_TOPIC creano invece auto-archi rispettivamente tra autori e tra paper.



## 3.2 Query 2: Top 5 Autori per Numero di Articoli

### Metodologia

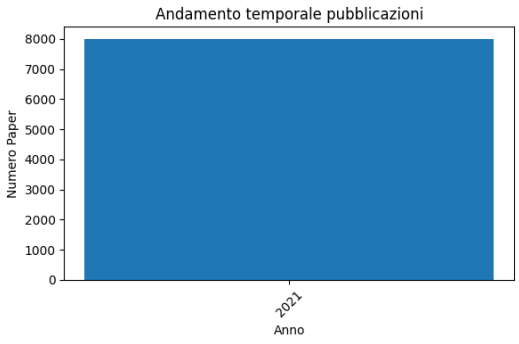
La query utilizza un pattern matching per contare le relazioni AUTHORED in uscita da ogni nodo Author:

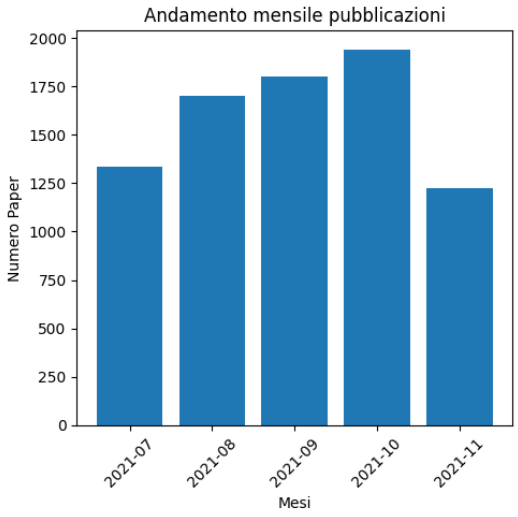
```
MATCH (a:Author)-[:AUTHORED]->(p:Paper)
RETURN a.name as author, count(p) as papers
ORDER BY papers DESC LIMIT 5
```

### Risultati

| # | Autore | N. Articoli |
|---|---|---|
| 1 | Wei Wang | 19 |
| 2 | Yang Li | 19 |
| 3 | Yang Liu | 19 |
| 4 | Xin Wang | 15 |
| 5 | H. Vincent Poor | 13 |

## 3.3 Query 3: Andamento Temporale delle Pubblicazioni




### Metodologia

L'analisi estrae l'anno dalla proprietà date di ogni paper e aggrega per conteggio:

```
MATCH (p:Paper)
WHERE p.date IS NOT NULL AND p.date <> ''
RETURN substring(p.date, 0, 4) as year,
count(p) as papers
ORDER BY year
```

Il grafico mostra che le pubblicazioni salvate risalgono al solo 2021, distibuite tra I mesi di luglio e novembre, con un'accelerazione particolarmente marcata in ottobre.

## 3.4 Query 4: Percorso Più Breve tra Due Autori

### Metodologia

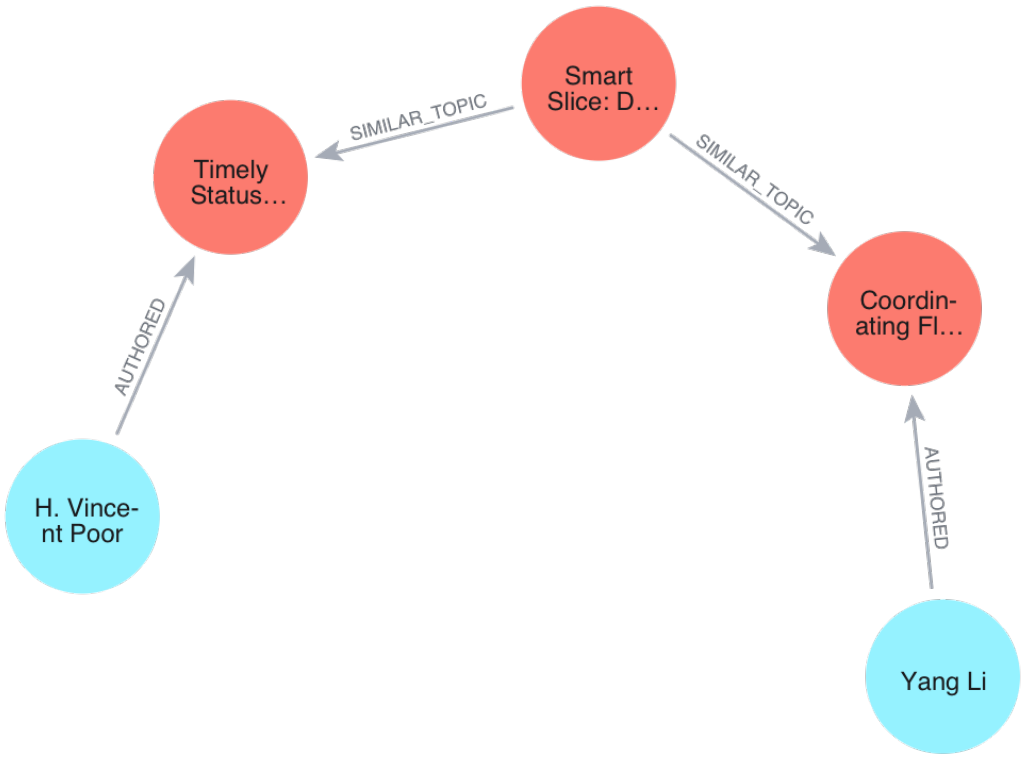
Sono stati selezionati due autori tra i più prolifici e utilizzato l'algoritmo **shortestPath** di Neo4j:

```
MATCH (a1:Author {name: 'Wei Wang'}),
      (a2:Author {name: 'Lajos Hanzo'})
MATCH path = shortestPath((a1)-[*]-(a2))
RETURN path
```

### Risultati

Il percorso più breve trovato ha **lunghezza 4**, strutturato come:

1. **Yang Li** (Author)
2. Paper: "Coordinating Flexible Demand Response and Renewable…"
3. Paper: "SmartSlice: Dynamic, self-optimization of applicat..."
4. Paper: " Timely Status Updating Over Erasure Channels Using..."
5. **H. Vincent Poor** (Author)



## 3.5 Query 5: Similarità tra Coppie di Autori

### Metodologia

La similarità tra due autori a1 e a2 è calcolata secondo la formula specificata nella consegna:

$$Sim(a1, a2) = (1 / |P1| \cdot |P2|) \cdot \Sigma \Sigma simT(pi, pj)$$

dove P1 e P2 sono gli insiemi di paper associati ai due autori, e simT è la similarità di Jaccard pesata tra paper. La formula considera tutte le coppie di paper, assegnando valore 0 alle coppie senza similarità.

### Implementazione

Per ridurre la complessità computazionale, e i limiti di memoria che ne impedivano l'esecuzione, è stato adottato un approccio ibrido Neo4j + Python:

1. Estrazione da Neo4j del numero di paper per autore.
2. Estrazione da Neo4j delle similarità tra paper tramite la relazione SIMILAR_TOPIC, impostando a 0 le coppie senza relazione.
3. Aggregazione e normalizzazione in Python con defaultdict, calcolando la media secondo la formula.

## 3.6 Query 6: Rete di Co-authorship

### Metodologia

La query estrae la rete di co-authorship limitata ai top 20 autori più prolifici.

Sono state trovate **13 connessioni** tra i top 20 autori. Tra essi la Top collaborazione è:  “Robert Schober - Derrick Wing Kwan Ng: 3 paper”