



UNIVERSITÀ DEGLI STUDI DI NAPOLI "FEDERICO II"

SCUOLA POLITECNICA E DELLE SCIENZE DI BASE

CORSO DI BIG DATA ENGINEERING - LAUREA MAGISTRALE IN INGEGNERIA INFORMATICA

ANNO ACCADEMICO 2022-2023

HOMEWORK 2

Big Data Engineering

Analytics su Database NoSQL MongoDB e Neo4J

Analisi delle competenze e dei Topic trattati nel corso degli anni applicate su una raccolta di dati relative a tutte le ricerche della Federico II di Napoli

Professore:

Ing. Vincenzo Moscato

Studenti:

Antonio Romano M63001315

Andriy Korsun M63001275

Giuseppe Riccio M63001314

Michele Cirillo M63001293

Indice

1 Raccolta e Preprocessing	1
1.1 Raccolta dei dati	1
1.1.1 Raccolta dei topic	2
1.2 Preprocessing dei dati	3
1.3 Tecnologie utilizzate	4
1.3.1 MongoDB	4
1.3.1.1 MongoDB Atlas	5
1.3.1.2 Workflow: Data collection → Python → MongoDB Atlas	5
1.3.2 Neo4J	7
1.3.2.1 Workflow: Raccolta dati → Cypher → Neo4j	7
2 Analytics	9
2.1 Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti	9
2.1.1 Implementazione	9
2.1.1.1 Caso specifico	10
2.1.1.2 Caso generale	11
2.1.2 Risultati	12
2.1.2.1 Esplorazione degli obiettivi sostenibili affrontati da 'Biological Sciences'	17
2.1.2.2 Esplorazione degli obiettivi sostenibili affrontati da 'Ingegneria'	18
2.2 Objective cycle dei progetti con obiettivi di sostenibilità affrontati dalla Federico II	19
2.2.1 Implementazione	19
2.2.2 Risultati	20
2.3 Top 10 degli ambiti di ricerca della Federico II per somma finanziata	21
2.3.1 Implementazione	21
2.3.1.1 Caso specifico	22
2.3.1.2 Caso generale	23
2.3.2 Risultati	24
2.3.2.1 Esplorazione degli ambiti con cui ha maggiormente collaborato 'Ingegneria'	26
2.3.2.2 Esplorazione degli ambiti con cui ha maggiormente collaborato 'Ingegneria Aereospaziale'	27
2.4 Wordcount dei topic dei primi 5 ambiti di ricerca della Federico II per numero progetti	28
2.4.1 Implementazione	28
2.4.2 Risultati	29
2.5 Linea temporale (per ogni anno) con evoluzione dei progetti iniziati per ogni ambito di ricerca della Federico II	31
2.5.1 Implementazione	31
2.5.2 Risultati	32
2.6 Top 5 Progetti con maggior interdisciplinarità tra Ambiti di Ricerca della Federico II	33
2.6.1 Implementazione	33
2.6.2 Risultati	34
2.7 Esecuzione definitiva: MongoDB - Neo4j	35
3 Conclusioni e Confronto tra le piattaforme	36
3.1 Il Confronto	36
3.2 Conclusioni	37
Elenco delle figure	38
Elenco delle tabelle	39
Bibliografia	40

1 Raccolta e Preprocessing

Contenuti

1.1	Raccolta dei dati	1
1.1.1	Raccolta dei topic	2
1.2	Preprocessing dei dati	3
1.3	Tecnologie utilizzate	4
1.3.1	MongoDB	4
1.3.2	Neo4J	7

1.1 Raccolta dei dati

Anche nel caso del secondo Homework, il dataset è una raccolta di tutti gli articoli di ricerca fatti dall'Università degli Studi di Napoli "Federico II" e comprende le seguenti colonne:

- **Rank** - Identificativo dell'Università;
- **Grant ID** - Codice univoco della sovvenzione da parte dell'UE;
- **Grant Number(s)** - Numero della legge con cui è stato sovvenzionato il progetto;
- **Title** - Titolo del progetto;
- **Title translated** - Titolo del progetto tradotto in inglese;
- **Abstract** - Descrizione del progetto;
- **Abstract translated** - Descrizione del progetto tradotto in inglese;
- **Keywords** - Parole chiave con cui è possibile sintetizzare la tematica trattata nel progetto;
- **Funding Amount** - Importo finanziato per il progetto;
- **Currency** - Valuta del finanziamento;
- **Funding Amount in EUR** - Importo finanziato per il progetto in euro;
- **Start Date** - Data inizio progetto;
- **Start Year** - Anno d'inizio del progetto;
- **End Date** - Data di fine del progetto;
- **End Year** - Anno di fine del progetto;
- **Researchers** - Ricercatori coinvolti nel progetto;
- **Research Organization - original** - Lista delle Istituzioni coinvolte nel progetto;
- **Research Organization - standardized** - Lista delle Istituzioni coinvolte nel progetto con Nomenclatura Ufficiale;

- **GRID ID** - Codice univoco assegnato dalla UE per identificare il progetto;
- **City of Research organization** - Città coinvolte nel progetto;
- **State of Research organization** - Regioni coinvolte nel progetto;
- **Country of Research organization** - Paesi coinvolti nel progetto;
- **Funder** - Ente finanziatore;
- **Funder Group** - Gruppo finanziatore;
- **Funder Country** - Paese finanziatore;
- **Program** - Ambito del programma di ricerca del progetto;
- **Resulting publications** - Identificativi delle pubblicazioni con i risultati del progetto;
- **Source Linkout** - URL della pagina del progetto;
- **Dimensions URL** - URL del progetto sulla piattaforma Dimensions;
- **Fields of Research (ANZSRC 2020)** - TOPIC del progetto, Standard Australiani;
- **RCDC Categories** - TOPIC sintetici del progetto, ricerche medicali nel sistema Americano;
- **HRCS HC Categories** - TOPIC sintetici del progetto, ricerche medicali nel sistema UK;
- **HRCS RAC Categories** - TOPIC sintetici del progetto, ricerche medicali nel sistema UK;
- **Cancer Types** - Tipo di tumore studiato nel progetto;
- **CSO Categories** - (Common Scientific Outline) rappresenta il tipo di cura studiata per trattare il cancro;
- **Units of Assessment** - Dipartimenti (Corsi di Laurea) di ricerca coinvolti nel progetto;
- **Sustainable Development Goals** - obiettivo di sostenibilità perseguito dal progetto.

L' Homework richiede inoltre di individuare, nella maniera più efficace possibile, le competenze della Federico II in termini di "topic" di ricerca affrontati nel corso degli anni.

Per svolgere questo procedimento si è proceduti a generare una colonna **Topic** a partire dalla colonna **Title translated** utilizzando le funzionalità offerte da GPT (Generative pre-trained transformers di OpenAI).

1.1.1 Raccolta dei topic

Si generano dei topic ipotetici per ogni titolo di progetto utilizzando il modello di generazione di testo "**text-davinci-003¹**" di OpenAI.

Viene creata una lista di topic generati e viene stampato il titolo di ogni progetto insieme al relativo topic.

A valle di questo procedimento si è dovuto necessariamente applicare anche il **lemmatizing²**.

¹ Il modello "**text-davinci-003**" è un modello di linguaggio di generazione di testo avanzato fornito da OpenAI. Si basa sull'architettura GPT-3.5 e rappresenta una versione specifica del modello GPT-3.5 di OpenAI addestrato per compiti di generazione di testo.

² Il **lemmatizing** è un processo di normalizzazione linguistica che consiste nel ridurre una parola alla sua forma base o lemma.

Topic generation
<pre>df = pd.read_csv('./Dataset/Dataset_Projects_Unina.csv', header=0) openai.api_key = "INSERIRE API-KEY" topic = [] for title in df['Title_translated']: prompt = "Given the title of the following research project: '{}'\n generate the hypothetic topic with a word limit of\n 10".format(title) if len(title) != 0: completion = openai.Completion.create(model="text-davinci-003", prompt=prompt, temperature=0, max_tokens=150) topic.append(completion['choices'][0]['text'])</pre>

Tabella 1.1: Generazione nuova colonna "Topic" a partire dalla colonna "Title Translated"

Lemmatizing of Topics
<pre>lemmatizer = WordNetLemmatizer() def lemmatize_text(text): words = word_tokenize(text) return ' '.join([lemmatizer.lemmatize(w) for w in words]) df['topic'] = df['topic'].apply(lemmatize_text)</pre>

Tabella 1.2: Fase di Lemmatizing

Il dataset allora presenta **38 colonne e 2964 righe**.

1.2 Preprocessing dei dati

Una fase critica della pipeline di qualsiasi applicazione di Big Data è quella di preprocessing dei dati, in cui vengono identificati e rimossi i dati errati, incompleti o duplicati, colonne costanti e/o irrilevanti ai fini della data analysis.

Nel nostro caso, il preprocessing ha portato alla seguente analisi:

- **Colonne eliminate**
 - **Rank** - valori costanti
 - **Grant ID** - è un codice identificativo
 - **Grant Number(s)** - è un numero identificativo
 - **Title** - è il titolo del progetto, viene mantenuto quello tradotto
 - **Abstract** - solo una descrizione del progetto

- **Abstract translated** - solo una descrizione del progetto
 - **Keywords** - valori spesso nulli
 - **Funding Amount** - viene mantenuto quello in EUR
 - **Currency** - non è di particolare interesse la valuta perché la maggior parte è EUR
 - **Researchers** - spesso non presenti
 - **GRID ID** - è un codice identificativo
 - **State of Research organization** - per molti paesi il valore è nullo
 - **City of Research organization** - non usato nell'analisi
 - **Program** - sigla del programma all'interno del quale è sovvenzionato il progetto
 - **Resulting publications** - è un codice identificativo
 - **Source Linkout** - è un URL, non esprime nessuna informazione utile
 - **Dimensions URL** - è un URL, non esprime nessuna informazione utile
 - **HRCS HC Categories** - spesso non presenti
 - **HRCS RAC Categories** - spesso non presenti
- **Colonne con valori multipli, NON eliminate**
- **Research Organization - original (standardized)**
 - **Country of Research organization**
 - **Fields of Research (ANZSRC 2020)**
 - **RCDC Categories**
 - **Cancer Types**
 - **CSO Categories**
 - **Units of Assessment**
 - **Sustainable Development Goals**

Si è fatta quest'analisi per ridurre le dimensioni del dataset in esame e per renderlo più facilmente gestibile in seguito. In particolare il dataset dopo il preprocessing presenta **18 colonne**.

ATTENZIONE: Non è stata effettuata alcuna **normalizzazione** sui dati in quanto non è richiesta nessuna pre-dizione, regressione o inferenza.

1.3 Tecnologie utilizzate

1.3.1 MongoDB

MongoDB è un database NoSQL orientato ai documenti che offre scalabilità, flessibilità ed è in grado di gestire enormi quantità di dati. Verranno di seguito descritte le principali proprietà di MongoDB:

- **Orientato ai documenti:** MongoDB immagazzina i dati come documenti, che sono strutture dati composte da coppie di chiavi e valori. I documenti sono simili agli oggetti JSON in termini di struttura e leggibilità, e sono raggruppati in collezioni, simili a tabelle in un database relazionale;

- **Schema flessibile:** A differenza dei database relazionali, i documenti in MongoDB possono avere campi diversi, permettendo una maggiore flessibilità nell'organizzazione dei dati;
- **Scalabilità orizzontale automatica:** MongoDB supporta la replica di dati e lo **sharding**, consentendo di distribuire i dati su più server per migliorare le prestazioni e la tolleranza ai guasti, il che lo rende altamente scalabile e affidabile;
- **Supporto per le transazioni:** A partire da MongoDB 4.0, è possibile eseguire transazioni multi-documento, simili a quelle in un database relazionale, che garantiscono proprietà ACID (Atomicity, Consistency, Isolation, Durability);
- **Driver in molteplici linguaggi:** MongoDB fornisce driver per una serie di linguaggi di programmazione, tra cui Java, Python, Node.js, C++ ed altri;
- **Alta disponibilità:** MongoDB offre alta disponibilità attraverso l'uso di replica sets, che sono gruppi di database che mantengono lo stesso set di dati. Se il database principale fallisce, un altro membro del replica set può prendere il suo posto;
- **Atomicità:** MongoDB supporta atomicità a livello di singolo documento, il che significa che le operazioni su un singolo documento sono atomiche: o avvengono completamente, o non avvengono affatto;
- **Aggregazioni:** MongoDB supporta varie operazioni di aggregazione, tra cui MapReduce. Quest'ultimo accetta una pipeline di operatori, dove le operazioni di filtraggio dovrebbero essere poste all'inizio per una maggiore efficienza.

1.3.1.1 MongoDB Atlas

Per l'utilizzo di MongoDB, si è proceduti con la configurazione di MongoDB Atlas. Esso è un servizio di gestione di database fornito da MongoDB che offre una piattaforma di gestione per le implementazioni di MongoDB, sia su piccola che su larga scala.

Proprietà che si aggiungono a quelle enunciate in precedenza per MongoDB sono:

- **Compatibilità:** MongoDB Atlas è completamente compatibile con i driver MongoDB e i client di MongoDB;
- **Database gestito:** Atlas gestisce automaticamente l'infrastruttura per il database MongoDB. Questo include il provisioning di server e la configurazione di MongoDB;
- **Sicurezza:** Atlas implementa un gran numero di funzioni di sicurezza, tra cui l'autenticazione, il controllo dell'accesso, la crittografia dei dati in riposo, la rete privata virtuale, l'auditing di sicurezza;
- **Monitoraggio e avvisi:** MongoDB Atlas fornisce strumenti di monitoraggio in tempo reale e avvisi configurabili;
- **Multi-Cloud:** Atlas offre la capacità di distribuire il tuo database su più provider di servizi cloud (AWS, Google Cloud, Azure), fornendo una maggiore flessibilità e ridondanza.

1.3.1.2 Workflow: Data collection → Python → MongoDB Atlas

Di seguito verranno forniti i passaggi, con annessa descrizione, che sono stati effettuati per poter caricare il file .CSV dei progetti di ricerca della Federico II come collection in un database MongoDB, utilizzando Python e MongoDB Atlas:

1. **Configurazione di MongoDB Atlas:** su MongoDB Atlas viene creato un cluster;
2. **Creazione della Collezione:** All'interno del cluster, viene creata una collezione vuota chiamata "research";
3. **Connessione al Cluster da Python:** Utilizzando Python, viene stabilita una connessione al cluster MongoDB Atlas utilizzando la libreria **pymongo**;
4. **Lettura del file CSV:** Viene letto il file .CSV contenente informazioni sui progetti di ricerca della Federico II dal sistema locale;
5. **Trasformazione dei dati:** Dopo aver letto il file CSV, vengono trasformati i dati da un DataFrame di pandas in una lista di documenti JSON. Tale passaggio è necessario in quanto MongoDB è un database orientato ai documenti che lavora con dati in formato BSON, una rappresentazione binaria di JSON;
6. **Pulizia della Collezione:** Prima di caricare i nuovi dati, vengono eliminati tutti i documenti esistenti nella collezione "research";
7. **Caricamento dei dati:** Infine, vengono caricati i dati in MongoDB Atlas, inserendo ogni documento della lista nella collezione "research".

Di seguito, il codice Python utilizzato:

```
mongodB MongoDB

client = MongoClient('mongodb+srv://<USERNAME>:<PASSWORD>@cluster0.xxxxxxx.mongodb.net/')

db = client.dataset_unina_research
research = db.research

if list(research.find({})) == []:
    df = pd.read_csv('./Dataset/Dataset_Projects_Unina_with_Topic.csv',
                     header=0)
    docs = json.loads(df.to_json(orient='records'))
    research.delete_many({})
    research.insert_many(docs)
```

Tabella 1.3: Workflow: Raccolta dati - Python - MongoDB Atlas

1.3.2 Neo4J

Neo4j è un database a grafo nativo ad alte prestazioni per dati fortemente correlati. Esso si basa sulla teoria dei grafici ed utilizza una struttura di dati composta da nodi e relazioni.

I nodi rappresentano le entità nel database, mentre le relazioni, che sono direzionate e tipizzate, collegano i nodi tra loro, rappresentando come queste entità sono collegate o interagiscono tra loro. Entrambi, nodi e relazioni, possono avere proprietà, che sono coppie chiave-valore di informazioni aggiuntive.

Verranno di seguito descritte le principali proprietà di Neo4j:

- **Modello di grafico nativo:** Neo4j è una base dati a grafo nativo che consente di modellare i dati in modo intuitivo e flessibile senza simulazioni;
- **Linguaggio di interrogazione grafica (Cypher):** Neo4j utilizza un linguaggio di interrogazione specifico per i grafici chiamato Cypher, che rende facile l'interrogazione e la manipolazione dei dati nel grafo;
- **Alte prestazioni:** Neo4j è ottimizzato per velocità e scalabilità. È in grado di gestire interrogazioni complesse e computazionalmente onerose con elevata efficienza;
- **Transazionalità ACID:** Come i database relazionali, Neo4j supporta le transazioni ACID (Atomicità, Coerenza, Isolamento, Durabilità) per singole transazioni, garantendo l'affidabilità dei dati;
- **Schemaless:** Neo4j non presenta uno schema fisso, permettendo quindi di aggiungere al grafo nuovi nodi e proprietà.

1.3.2.1 Workflow: Raccolta dati → Cypher → Neo4j

Per l'utilizzo di Neo4j, si è proceduti con la configurazione di Neo4j Desktop, un'applicazione desktop che fornisce un ambiente di sviluppo integrato (IDE). Di seguito sono riportati i passaggi generali, con annessa descrizione, per creare un progetto su Neo4j Desktop, e per poter caricare il file .CSV dei progetti di ricerca della Federico II utilizzando Cypher:

1. **Creazione di un nuovo progetto;**
2. **Creazione di un DBMS locale:** Nel progetto, verrà creato un "Local DBMS" che conterrà le informazioni degli ambiti di ricerca della Federico II. NB: Oltre al nome, è possibile configurare anche la password e la versione di Neo4j;
3. **Avvio del database;**
4. **Apertura del terminale del database:** Si prosegue con l'apertura del terminale del database (denominato **Neo4j Browser**);
5. **Caricamento dei dati CSV:** Verrà poi caricato, tramite la directory di importazione di Neo4j, il file .CSV contenente le informazioni dei progetti di ricerca della Federico II. Verrà utilizzato il seguente comando per poter caricare i dati dal file CSV nel database:
`LOAD CSV WITH HEADERS FROM 'file:///path/to/csv_file.csv' AS row
CREATE (n:Project)
SET n = row`
6. **Creazione di nodi e relazioni:** Si proseguirà poi con i vari comandi, come ad esempio **MERGE**, **CREATE** o **MATCH** in Cypher per poter creare nodi e relazioni nel database basandosi sui dati caricati;
7. **Interrogazione dei dati:** Dopo aver caricato e organizzato i dati, si potranno eseguire interrogazioni Cypher per esplorare i dati, effettuando le analytics che discuteremo nei paragrafi successivi.

Segue l'implementazione del caricamento del file .CSV su Neo4j e la visualizzazione del database in forma di **metagrafo**, il quale mostra lo schema del database e le relazioni tra i nodi:

```
neo4j

LOAD CSV WITH HEADERS FROM
  'file:///Dataset_Projects_Unina_with_Topic.csv' AS row
MERGE (p:Project {Title_translated: row.Title_translated})
SET p.Funding_Amount_in_EUR =toFloat(row.Funding_Amount_in_EUR),
p.Start_Year = toInteger(row.Start_Year),
p.Topic = row.Topic

FOREACH (field IN SPLIT(row.Fields_of_Research_ANZSRC_2020, ';') |
  MERGE (a:Ambiti {Field: TRIM(field)})
  CREATE (p)-[:BELONGS_TO]->(a)
)

FOREACH (sust IN SPLIT(row.Sustainable_Development_Goals, ';') |
  MERGE (s:ObiettiviSostenibili {Sust: TRIM(sust)})
  CREATE (p)-[:FOCUSSED_ON]->(s)
)

FOREACH (dip IN SPLIT(row.Units_of_Assessment, ';') |
  MERGE (d:Dipartimenti {Dip: TRIM(dip)})
  CREATE (p)-[:REALIZED_BY]->(d)
)
```

Tabella 1.4: Workflow: Raccolta dati - Cypher – Neo4j

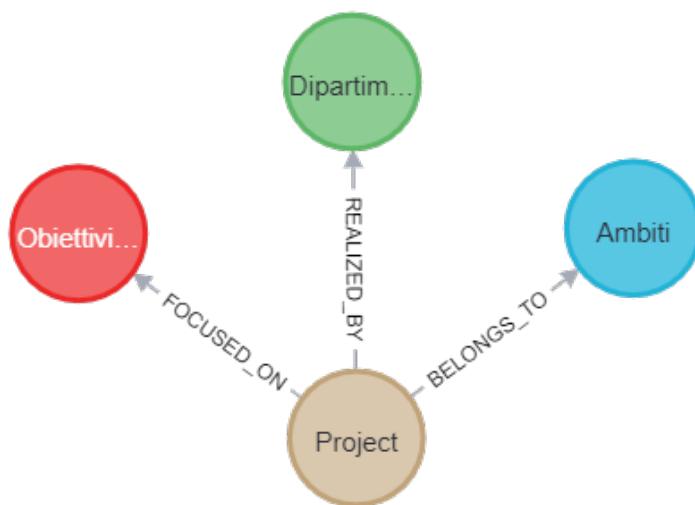


Figura 1.1: Metagrafo - Visualizzazione schema database, nodi e relazioni

2 Analytics

Contenuti

2.1	Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti	9
2.1.1	Implementazione	9
2.1.2	Risultati	12
2.2	Objective cycle dei progetti con obbiettivi di sostenibilità affrontati dalla Federico II	19
2.2.1	Implementazione	19
2.2.2	Risultati	20
2.3	Top 10 degli ambiti di ricerca della Federico II per somma finanziata	21
2.3.1	Implementazione	21
2.3.2	Risultati	24
2.4	Wordcount dei topic dei primi 5 ambiti di ricerca della Federico II per numero progetti	28
2.4.1	Implementazione	28
2.4.2	Risultati	29
2.5	Linea temporale (per ogni anno) con evoluzione dei progetti iniziati per ogni ambito di ricerca della Federico II	31
2.5.1	Implementazione	31
2.5.2	Risultati	32
2.6	Top 5 Progetti con maggior interdisciplinarità tra Ambiti di Ricerca della Federico II	33
2.6.1	Implementazione	33
2.6.2	Risultati	34
2.7	Esecuzione definitiva: MongoDB - Neo4j	35

Come richiesto dall'HW2, ed in parte fatto nell'HW1, di seguito verranno mostrate diverse analisi di dati relative alla somma finanziata, ai **topic/tematiche** affrontate nei progetti di ricerca, da cui è stato possibile ottenere un quadro dettagliato dell'**impegno dell'Università** nel corso degli anni verso la **comunità scientifica mondiale**. In questa raccolta di query, vengono selezionate le analisi più significative che permetteranno di comprendere la portata e la varietà delle attività di ricerca condotte dalla **Federico II**. Per Topic specifici, vengono inoltre mostrati gli ambiti di ricerca più gettonati, effettuando anche un WordCount delle parole più frequenti per ogni Topic ed un andamento temporale di questi ultimi per gli ambiti di ricerca più frequenti.

2.1 Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti

L'obiettivo della prima Analytic è quello di calcolare il numero di progetti della Federico II, e di graficare con l'ausilio di librerie e strumenti grafici, per ogni ambito di ricerca specifico (caratterizzati da un codice identificativo a 4 cifre) e ambito di ricerca generale (caratterizzati da un codice identificativo a 2 cifre), i primi 10 ambiti di ricerca.

2.1.1 Implementazione

Nell'implementazione:

- Viene eseguita la query di aggregazione sulla collezione **research**;
- Viene utilizzata la funzione **\$project** insieme alla funzione **\$split** per trasformare i dati di ingresso. In particolare, il campo **FieldsOfResearchANZSRC2020** viene diviso per ogni occorrenza del delimitatore ";", generando così un array;

- Viene utilizzata la funzione **\$unwind**, che scomponete l'array generato nel passaggio precedente in tanti documenti separati, uno per ogni elemento dell'array;
- Viene utilizzata la funzione **\$match** che filtra i documenti, selezionando solo i documenti per i quali il campo *FieldsOfResearchANZSRC2020* inizia con 2 o 4 cifre;
- Viene utilizzata la funzione **\$group** per raggruppare i documenti in base al campo *FieldsOfResearchANZSRC2020*, dopo aver rimosso gli eventuali spazi bianchi all'inizio o alla fine del campo con la funzione **\$trim**. Per ogni gruppo di documenti, viene conteggiato il numero di documenti che lo compongono;
- Viene eseguita la funzione **\$sort** che ordina i documenti in ordine decrescente per il campo *Numeri_Progetti*;
- Infine, viene utilizzata la funzione **\$limit** limitando il numero di documenti restituiti a 10.

2.1.1.1 Caso specifico

Per il caso specifico, viene utilizzata la funzione **\$match** (la quale volendo fare un confronto con il linguaggio SQL corrisponde alla clausola **WHERE**) per filtrare i documenti, selezionando solo i documenti per i quali il campo *FieldsOfResearchANZSRC2020* inizia con 4 cifre.

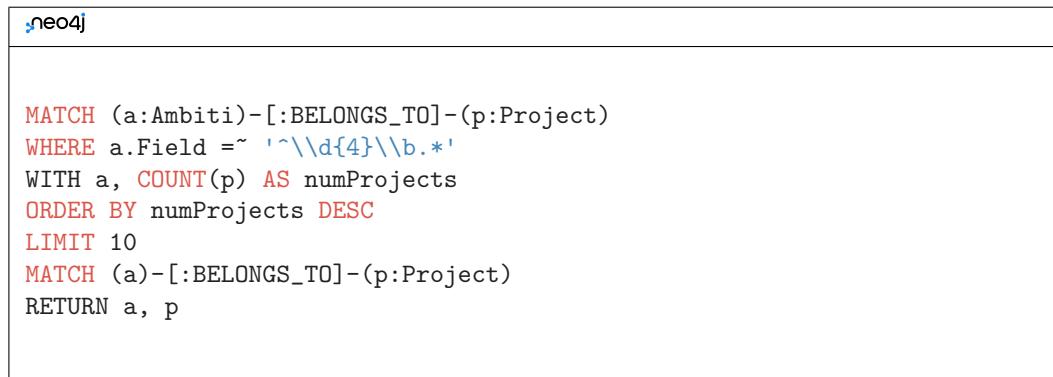
Le seguenti implementazioni in **Python - MongoDB, Cypher - Neo4j**:



MongoDB

```
query10tematiche = list(research.aggregate([
    {"$project": {"Fields_of_Research_ANZSRC_2020": {"$split": [
        "$Fields_of_Research_ANZSRC_2020", " ", "]}}},
    {"$unwind": "$Fields_of_Research_ANZSRC_2020"}, 
    {"$match": {"Fields_of_Research_ANZSRC_2020": {"$regex": r'^\d{4}'}}}, # Filtra i documenti dove Fields_of_Research_ANZSRC_2020 inizia con 4 cifre
    {"$group": {"_id": {"$trim": {"input": 
        "$Fields_of_Research_ANZSRC_2020"}}, "Numero_Progetti": {"$sum": 1}}}, # Raggruppa per Fields_of_Research_ANZSRC_2020 dopo aver tolto gli spazi bianchi e conta
    {"$project": {"_id": 0, "Ambito_di_Ricerca": "$_id",
        "Numero_Progetti": "$Numero_Progetti"}},
    {"$sort": {"Numero_Progetti": -1}}, # Ordina in ordine decrescente per Numero_Progetti
    {"$limit": 10}
]))
```

Tabella 2.1: Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso specifico) - **Python - MongoDB**



```

neo4j

MATCH (a:Ambiti)-[:BELONGS_TO]-(p:Project)
WHERE a.Field =~ '^\\d{4}\\b.*'
WITH a, COUNT(p) AS numProjects
ORDER BY numProjects DESC
LIMIT 10
MATCH (a)-[:BELONGS_TO]-(p:Project)
RETURN a, p
  
```

Tabella 2.2: Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso specifico) - **Cypher – Neo4j**

2.1.1.2 Caso generale

Per il caso generale, viene utilizzata la funzione **\$match** (la quale volendo fare un confronto con il linguaggio SQL corrisponde alla clausola **WHERE**) per filtrare i documenti, selezionando solo i documenti per i quali il campo *FieldsOfResearchANZSRC2020* inizia con 2 cifre.

Le seguenti implementazioni in **Python - MongoDB, Cypher - Neo4j**:



```

query10tematicheg = list(research.aggregate([
    {"$project": {"Fields_of_Research_ANZSRC_2020": {"$split": [
        "$Fields_of_Research_ANZSRC_2020", " "]}},
    {"$unwind": "$Fields_of_Research_ANZSRC_2020"}, 
    {"$match": {"Fields_of_Research_ANZSRC_2020": {"$regex": r'^\d{2}'}}, # Filtra i documenti dove Fields_of_Research_ANZSRC_2020 inizia con 2 cifre
    {"$group": {"_id": {"$trim": {"input": 
        "$Fields_of_Research_ANZSRC_2020"}}, "Numero_Progetti": {"$sum": 1}}},
    {"$project": {"_id": 0, "Ambito_di_Ricerca": "$_id",
        "Numero_Progetti": "$Numero_Progetti"}},
    {"$sort": {"Numero_Progetti": -1}}, # Ordina in ordine decrescente per Numero_Progetti
    {"$limit": 10}
])) 
  
```

Tabella 2.3: Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso generale) - **Python - MongoDB**

```

neo4j
MATCH (a:Ambiti)-[:BELONGS_TO]-(p:Project)
WHERE a.Field =~ '^\\d{2}\\b.*'
WITH a, COUNT(p) AS numProjects
ORDER BY numProjects DESC
LIMIT 10
MATCH (a)-[:BELONGS_TO]-(p:Project)
RETURN a, p
  
```

Tabella 2.4: Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso generale) - **Cypher – Neo4j**

2.1.2 Risultati

Dai risultati mostrati nelle due Figure seguenti, si può notare come, nel corso degli anni, l'Università Federico II ha avviato il maggior numero di progetti nel settore delle **Scienze Biologiche e Biomediche**, vedendo rispettivamente il campo "Biochemistry and Cell Biology", "Genetics" e "Oncology and Carcinogenesis" occupare le prime 3 posizioni per numero di progetti.

Negli ambiti generici, si può notare come il campo **Ingegneria** segue gli ambiti precedentemente specificati; si contano ben 60 progetti nei topic di Ingegneria dei Materiali e Civile.

Ambito_di.Ricerca	Numero_Progetti
3101 Biochemistry and Cell Biology	211
3105 Genetics	126
3211 Oncology and Carcinogenesis	88
3202 Clinical Sciences	81
4303 Historical Studies	78
3403 Macromolecular and Materials Chemistry	77
3705 Geology	62
4016 Materials Engineering	60
4005 Civil Engineering	57
3204 Immunology	50

Tabella 2.5: Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso specifico)

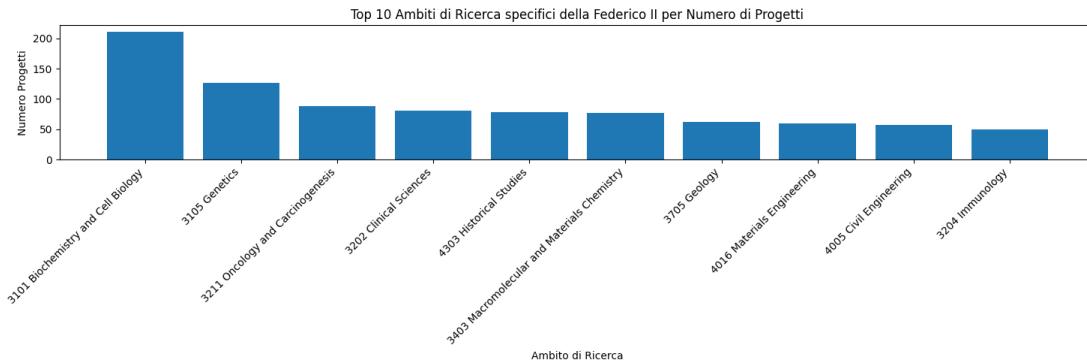


Figura 2.1: Bar Chart - Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso specifico)

Ambito.di.Ricerca	Numero.Progetti
31 Biological Sciences	581
32 Biomedical and Clinical Sciences	540
40 Engineering	489
34 Chemical Sciences	248
46 Information and Computing Sciences	164
30 Agricultural, Veterinary and Food Sciences	157
44 Human Society	153
37 Earth Sciences	135
33 Built Environment and Design	129
43 History, Heritage and Archaeology	125

Tabella 2.6: Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso generale)

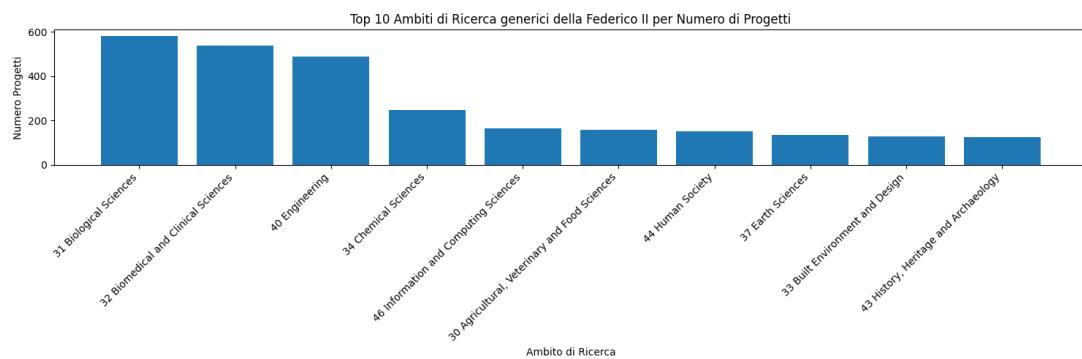


Figura 2.2: Bar Chart - Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso generale)

Nella **successiva Bar Chart Race**, si vuole mostrare come variano, **nel corso degli anni**, il numero di progetti per ciascun Topic ed ambito di ricerca.

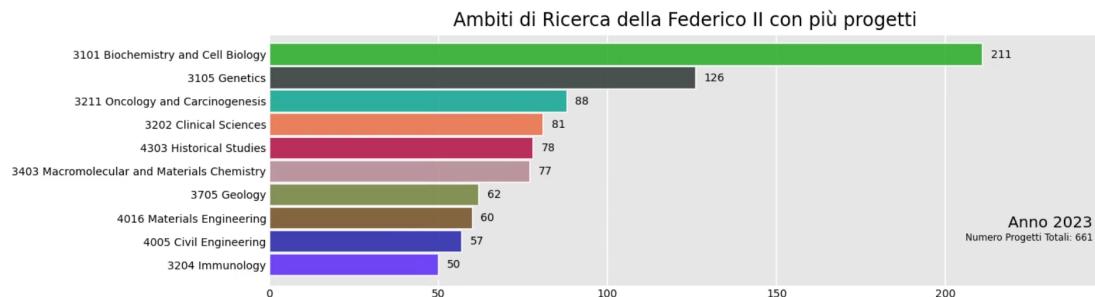


Figura 2.3: Bar Chart Race - Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso specifico)

Si noti: La Figura sovrastante mostrata è stata catturata all'interno di un video, che mostra l'andamento del numero di progetti per ciascun Ambito di ricerca. Per la visione del video completo si rimanda al seguente link:

<https://shorturl.at/mzQY8>

Nei grafi sottostanti, invece, è possibile visionare i risultati della prima analytic sotto forma di grafo, ottenuti da **Neo4J**.

Dalla Figura 2.4, si può notare una maggior affluenza/presenza di un gran numero di progetti di ricerca legati all'ambito (specifico) "Biochemistry and Cell Biology (3101)" e all'ambito "Genetics (3105)". Si può inoltre notare come, per l'ambito "Macromolecular and Materials Chemistry (3403)", vi sia una una forte collaborazione, intesa per numero di progetti, con gli ambiti "Materials Engineering (4016)" e "Civil Engineering (4005)":

- Progetti coinvolti nella prima collaborazione, si incentrano su tematiche riguardanti la verifica della qualità e affidabilità dei materiali compositi e delle strutture per applicazioni di trasporto, sulla progettazione e sviluppo di celle solari multicristalline in silicio ad alta efficienza, che sono una componente chiave per l'energia solare;
- Progetti coinvolti nella seconda collaborazione, si incentrano su tematiche riguardanti lo sviluppo di tecniche di costruzione in acciaio leggero, sicure, sostenibili e ad alta efficienza energetica, sulla modellizzazione chimico-fisico-meccanica della durabilità nelle strutture di calcestruzzo armato.

Mentre, per la Figura 2.5 si nota subito l'enorme quantità di progetti di ricerca sviluppati negli ambiti (caso generale) delle "Biomedical and Clinical Sciences (32)" e delle "Biological Sciences (31)". Si può inoltre notare come, tra i due, vi sia una una forte collaborazione, intesa al numero di progetti di ricerca:

- Progetti coinvolti tra la collaborazione di tali ambiti, prevedono la trattazione da varie aree di ricerca biomedica, includendo l'immunoterapia del carcinoma pancreatico, lo studio delle risposte infiammatorie legate a specifiche molecole, gli effetti metabolici e funzionali dell'inulina su sistema cardiovascolare e sistema nervoso centrale, e infine le alterazioni genetiche coinvolte nella predisposizione e progressione di tumori gastrointestinali, enfatizzando dunque su trattazioni di oncologia, la neurologia e l'immunologia.

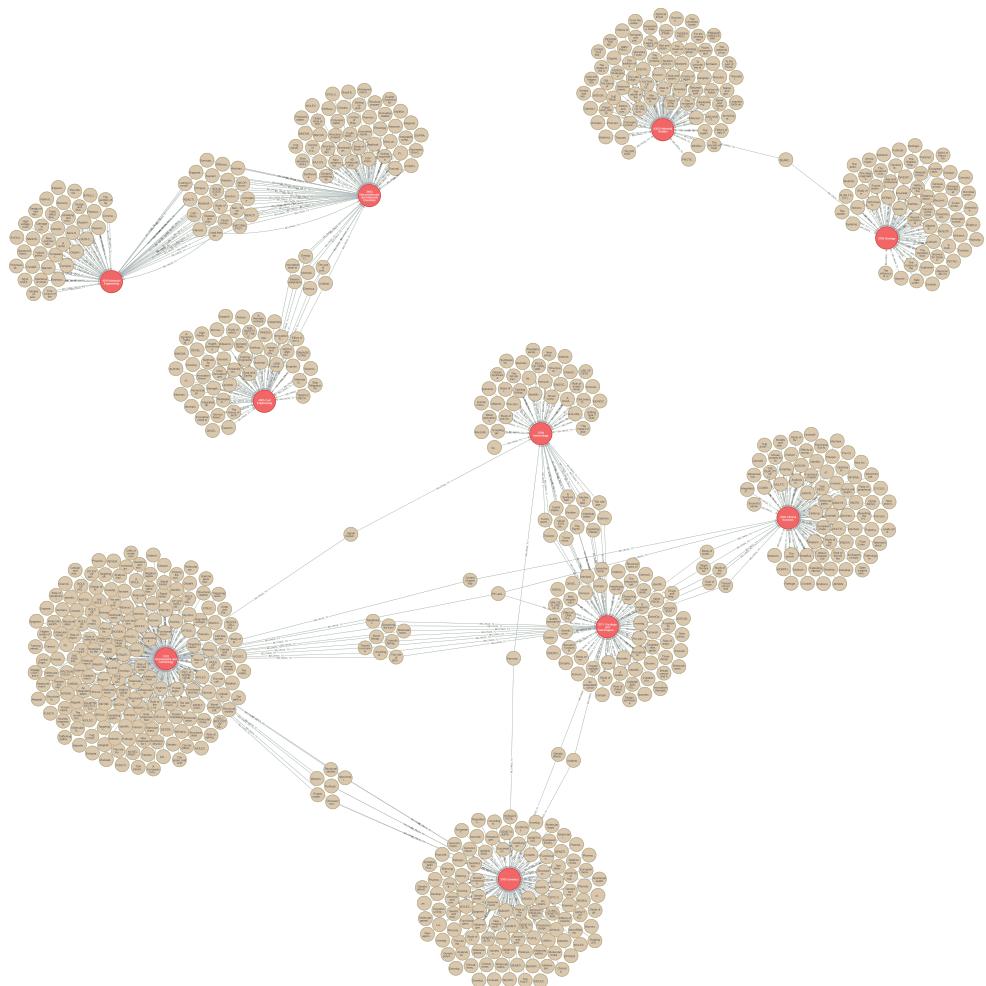


Figura 2.4: Grafo - Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso specifico)

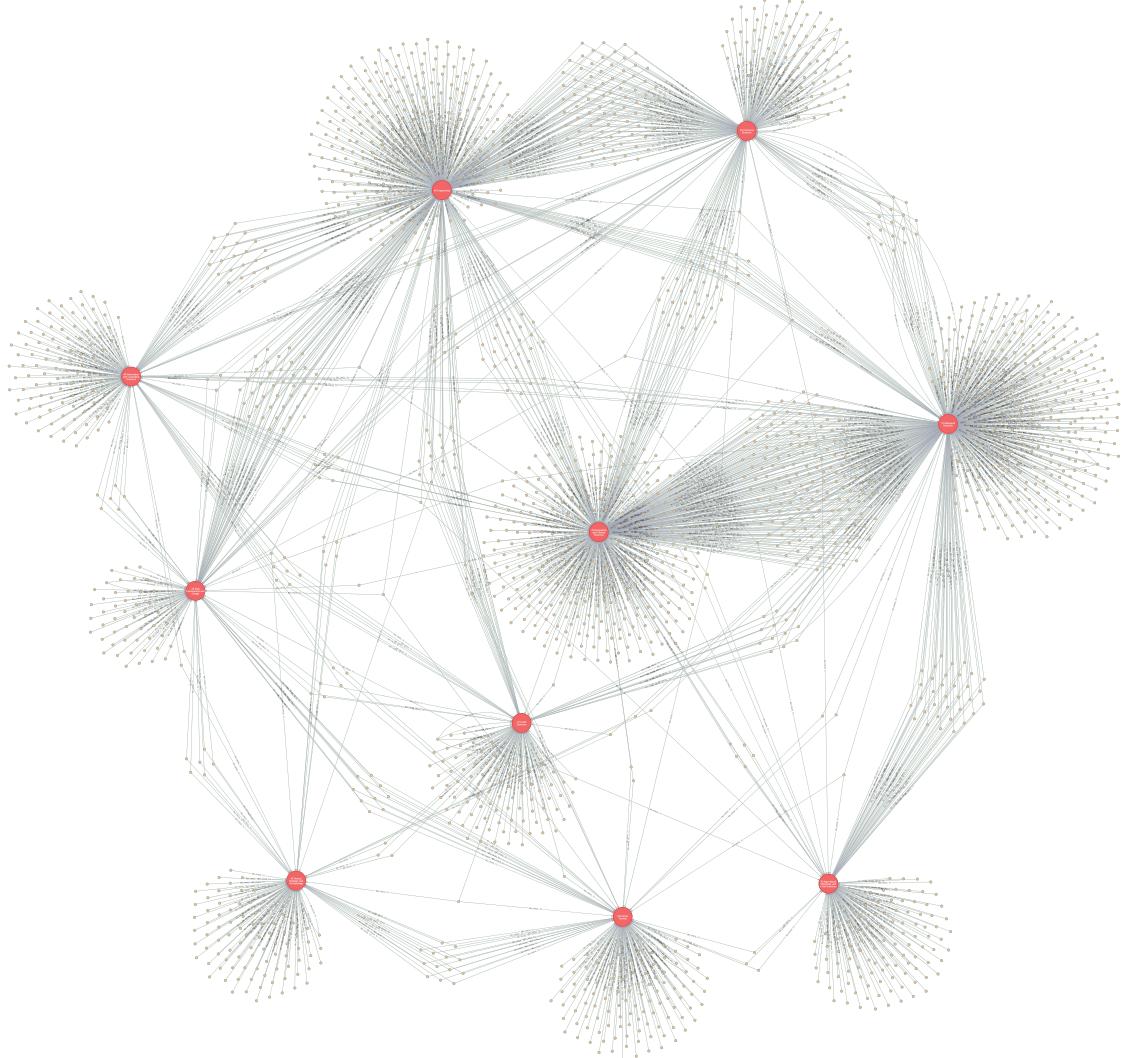


Figura 2.5: Grafo - Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso generale)

2.1.2.1 Esplorazione degli obiettivi sostenibili affrontati da 'Biological Sciences'

A partire dai risultati, si nota come l'ambito di ricerca "Biological Sciences" risulti essere il più attivo; si è voluto quindi evidenziare gli obiettivi sostenibili correlati. Infatti, attraverso la rappresentazione grafica sottostante, è possibile visualizzare in modo chiaro le connessioni tra l'ambito "Biological Sciences", i progetti correlati e gli obiettivi di sostenibilità e comprendere meglio come gli obiettivi vengono affrontati all'interno dell'ambito di ricerca. Si lascia l'implementazione e il grafo completo riguardante l'esplorazione degli obiettivi sostenibili affrontati da 'Biological Sciences':

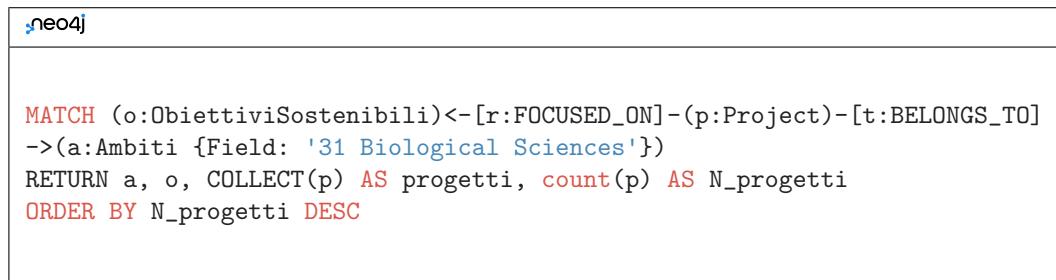


Tabella 2.7: Esplorazione degli obiettivi sostenibili affrontati da 'Biological Sciences' - **Cypher – Neo4j**



Figura 2.6: Grafo - Esplorazione degli obiettivi sostenibili affrontati da 'Biological Sciences'

2.1.2.2 Esplorazione degli obiettivi sostenibili affrontati da 'Ingegneria'

Un'analisi simile a quella fatta precedentemente per l'ambito di ricerca "Biological Sciences", è stato effettuato anche per l'ambito di ricerca "Engineering". Un esempio che è possibile discutere, come si nota dal grafo, è la collaborazione tra il campo Engineering e Climate Action, che hanno portato a una serie di ricerche incentrate sullo sviluppo di soluzioni sostenibili ed efficienti per la gestione dell'energia. Alcune ricerche includono lo sviluppo di tecnologie energetiche innovative, l'abilitazione di tecnologie di combustione sostenibili attraverso modelli ibridi basati sui dati e il miglioramento della gestione del calore e dell'efficienza energetica di mezzi di mobilità, come aerei e navi. L'obiettivo comune è appunto quello di ridurre l'impatto ambientale attraverso l'innovazione tecnologica e l'ingegneria sostenibile.

Si lascia l'implementazione e il grafo completo riguardante l'esplorazione degli obiettivi sostenibili affrontati da 'Engineering':

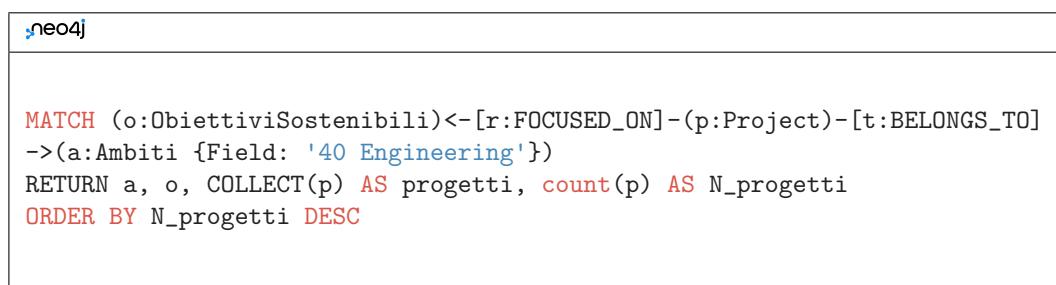


Tabella 2.8: Esplorazione degli obiettivi sostenibili affrontati da 'Engineering' - Cypher – Neo4j

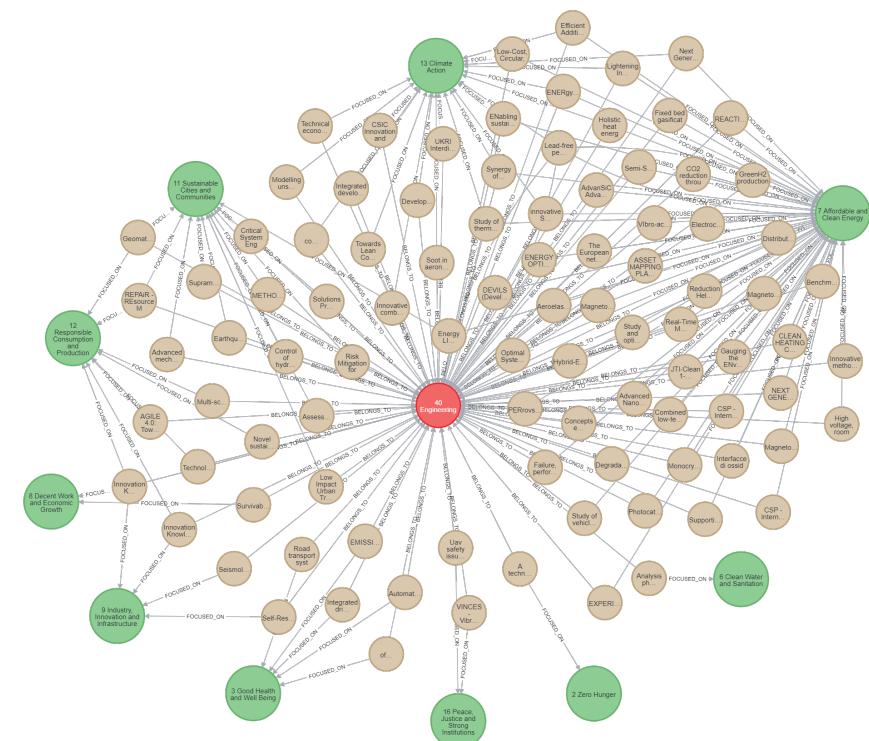


Figura 2.7: Grafo - Esplorazione degli obiettivi sostenibili affrontati da 'Engineering'

2.2 Objective cycle dei progetti con obbiettivi di sostenibilità affrontati dalla Federico II

L'obiettivo della seconda query, in accordo con le esplorazioni valutate nei precedenti risultati, è quello di estrarre un "Objective Cycle", che segue il concetto del grafico di Gartner "Hype Cycle", andando a mostrare le **competenze della Federico II in ambito di sviluppo sostenibile**.

2.2.1 Implementazione

Nell'implementazione:

- **\$project**: Viene utilizzato per proiettare solo il campo **Sustainable_Development_Goals** e dividerlo in una lista di obiettivi separati utilizzando il carattere ";" come delimitatore.
- **\$unwind**: Viene utilizzato per srotolare la lista di obiettivi, creando un documento separato per ciascun obiettivo.
- **\$group**: Viene utilizzato per raggruppare gli obiettivi sostenibili e calcolare il numero di progetti associati ad ogni obiettivo.
- **\$project**: Viene utilizzato per ridenominare i campi e rimuovere l'id del gruppo.
- **\$sort**: Viene utilizzato per ordinare i risultati in base al numero di progetti in ordine decrescente e agli obiettivi sostenibili in ordine crescente.
- **\$limit**: Viene utilizzato per limitare il numero di risultati restituiti a 10, ottenendo solo i primi 10 obiettivi più frequenti.

Di seguito l'implementazione in **Python - MongoDB**:



```
querySustanaibleGoals = list(research.aggregate([
    {"$project": {"Sustainable_Development_Goals": {"$split": [
        "$Sustainable_Development_Goals", "; "]}}},
    {"$unwind": "$Sustainable_Development_Goals"},
    {"$group": {"_id": {"$trim": {"input": 
        "$Sustainable_Development_Goals"}}, "Numero_Progetti": {"$sum": 
        1}}},
    {"$project": {"_id": 0, "obbiettivi_Sostenibili": "$_id",
        "Numero_Progetti": "$Numero_Progetti"}},
    {"$sort": {"Numero_Progetti": -1, "obbiettivi_Sostenibili": 1}},
    {"$limit": 10}
]))
```

Tavella 2.9: Objective cycle dei progetti con obbiettivi di sostenibilità affrontati dalla Federico II - **Python - MongoDB**

2.2.2 Risultati

È possibile notare come gli obiettivi di sostenibilità affrontati dalla Federico II collocati all'inizio sono quelli in fase emergente (pochi progetti), quelli sul picco sono in rapido sviluppo (numero progetti medio ma in crescita) quelli sul plateau finale sono gli obiettivi consolidati (molti progetti).

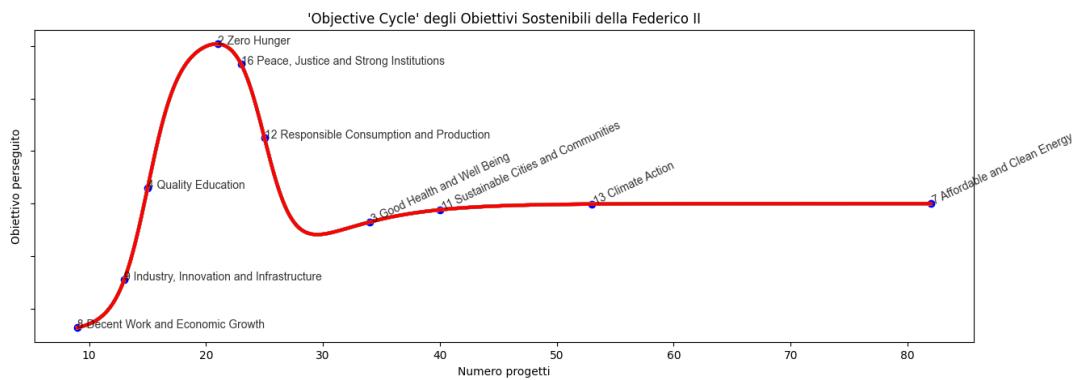


Figura 2.8: Objective Cycle dei progetti con obiettivi di sostenibilità affrontati dalla Federico II

In accordo con l'Objective Cycle, si lascia l'implementazione e il grafo completo riguardante l'esplorazione degli obiettivi sostenibili affrontati per numero di progetti (TOP 10):

```
neo4j

MATCH (s:ObiettiviSostenibili)-[:FOCUSED_ON]-(p:Project)
WITH s, COUNT(p) AS numProjects
ORDER BY numProjects DESC
LIMIT 10
MATCH (s)-[:FOCUSED_ON]-(p:Project)
RETURN s, p
```

Tabella 2.10: Top 10 obiettivi sostenibili degli ambiti di ricerca della Federico II con il maggior numero dei progetti - **Cypher – Neo4j**

Si può notare, nel grafo sottostante in Figura 2.9, come sono presenti, in numero maggiore, progetti legati all'ambito delle "Affordable and Clean Energy (7)" ed una grossa collaborazione di tale ambito con le "Climate Action (13)", per progetti che hanno l'obiettivo di fornire energia sostenibile, riducendo le emissioni di gas serra e promuovendo la transizione verso un'economia a basse emissioni di carbonio. Tutto ciò, per **mitigare gli impatti negativi del cambiamento climatico**, promuovendo una transizione energetica equa e sostenibile.

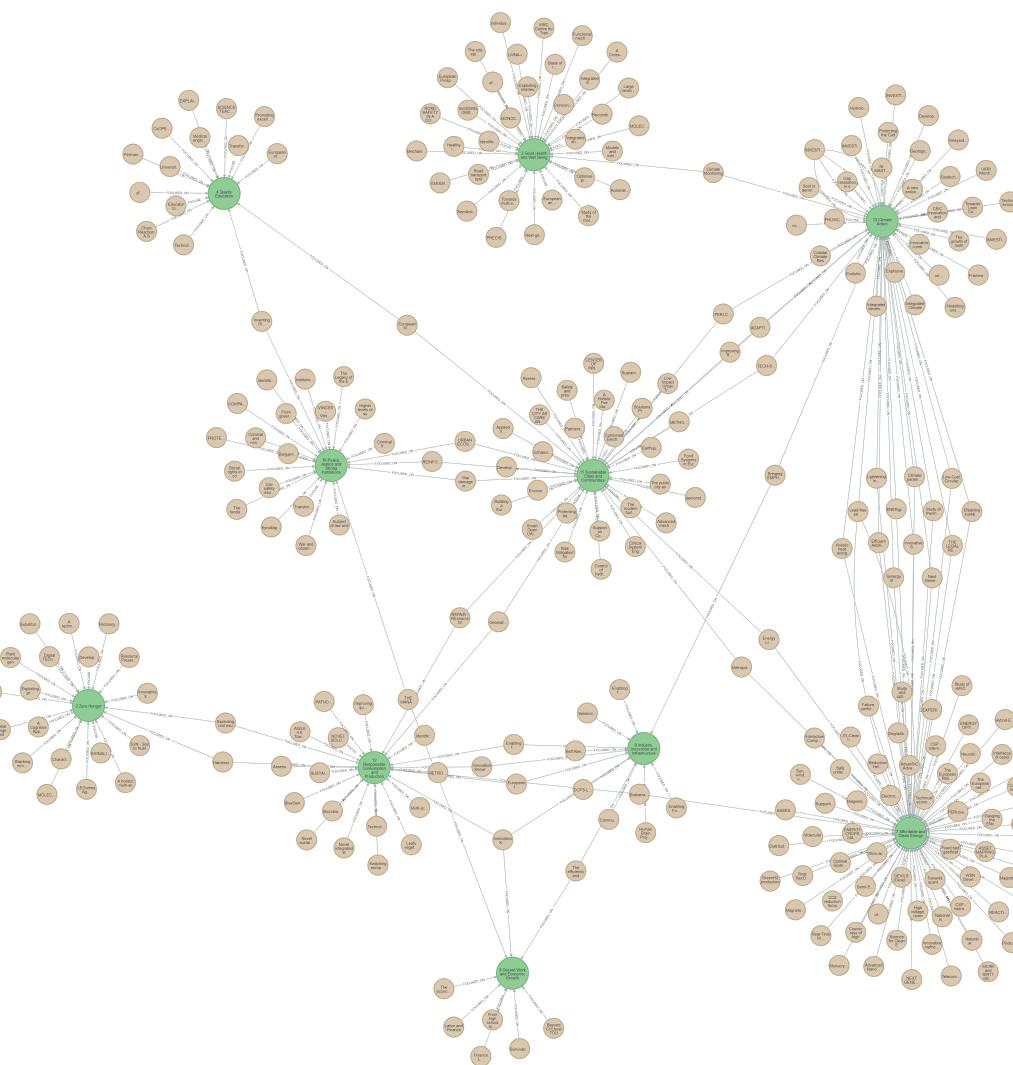


Figura 2.9: Grafo - Top 10 obiettivi sostenibili degli ambiti di ricerca della Federico II con il maggior numero dei progetti

2.3 Top 10 degli ambiti di ricerca della Federico II per somma finanziata

L'obiettivo della terza Analytic è quello di trovare, sia per l'ambito di ricerca specifico (caratterizzati da un codice identificativo a 4 cifre) che per l'ambito di ricerca generale (caratterizzati da un codice identificativo a 2 cifre), l'ambito maggiormente finanziato.

2.3.1 Implementazione

Nell'implementazione:

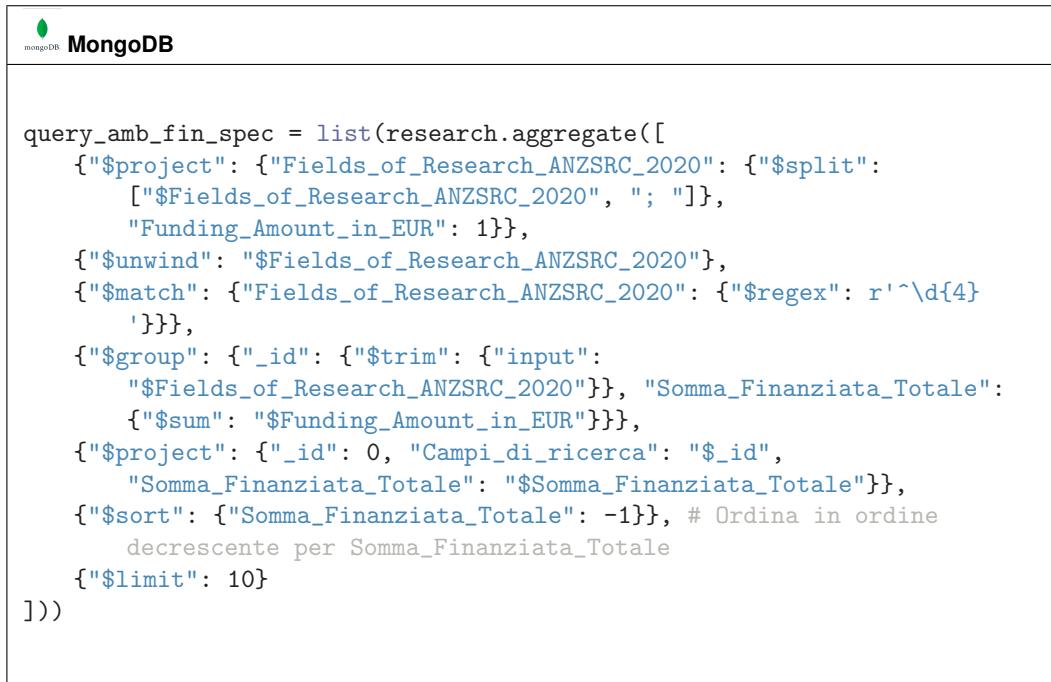
- Viene eseguita la query di aggregazione sulla collezione **research**;
- Viene utilizzata la funzione **\$project** sulla collezione **research** e vengono selezionati i campi **FieldsOfResearchANZSRC2020** e **Funding_Amount.in.EUR**. Il campo **FieldsOfResearchANZSRC2020** viene diviso in un array basandosi sul delimitatore "; ";

- Viene utilizzata la funzione **\$unwind** per scomporre l'array generato nel passaggio precedente in documenti separati, uno per ogni elemento dell'array;
- Viene utilizzata la funzione **\$match** per filtrare i documenti, selezionano solo quelli per i quali il campo FieldsOfResearchANZSRC2020 inizia con 2 o 4 cifre;
- Viene utilizzata la funzione **\$group** per raggruppare i documenti in base al campo FieldsOfResearchANZSRC2020 (dopo aver rimosso gli eventuali spazi bianchi con la funzione **\$trim**) e calcolando la somma di Funding_Amount_in_EUR per ogni gruppo;
- Viene utilizzata la funzione **\$project** per rinominare i campi _id in Campi_di_ricerca e mantenere il campo Somma_Finanziata_Totale;
- Viene utilizzata la funzione **\$sort** per ordinare i documenti in ordine decrescente per il campo SommaFinanziataTotale;
- Infine, viene utilizzata la funzione **\$limit** limitando il numero di documenti restituiti a 10.

2.3.1.1 Caso specifico

Per il caso specifico, viene utilizzata la funzione **\$match** che filtra i documenti, selezionando solo i documenti per i quali il campo FieldsOfResearchANZSRC2020 inizia con 4 cifre.

Le seguenti implementazioni in **Python - MongoDB, Cypher - Neo4j**:



```

MongoDB

query_amb_fin_spec = list(research.aggregate([
    {"$project": {"Fields_of_Research_ANZSRC_2020": {"$split": [
        "$Fields_of_Research_ANZSRC_2020", " ", ""}],
        "Funding_Amount_in_EUR": 1}},
    {"$unwind": "$Fields_of_Research_ANZSRC_2020"}, 
    {"$match": {"Fields_of_Research_ANZSRC_2020": {"$regex": r'^\d{4}"}},
        }},
    {"$group": {"_id": {"$trim": {"input": 
        "$Fields_of_Research_ANZSRC_2020"}}, "Somma_Finanziata_Totale": {
            "$sum": "$Funding_Amount_in_EUR"}}, 
        },
    {"$project": {"_id": 0, "Campi_di_ricerca": "$_id",
        "Somma_Finanziata_Totale": "$Somma_Finanziata_Totale"}},
    {"$sort": {"Somma_Finanziata_Totale": -1}}, # Ordina in ordine
        decrescente per Somma_Finanziata_Totale
    {"$limit": 10}
]))
```

Tabella 2.11: Top 10 ambiti di ricerca della Federico II per somma finanziata (caso specifico) - **Python - MongoDB**



```

MATCH (p:Project)-[:BELONGS_TO]->(a:Ambiti)
WITH a, sum(p.Funding_Amount_in_EUR) as TotalFunding
WHERE a.Field =~ '^\\d{4}\\b.*'
RETURN a.Field as AmbitoRicerca, TotalFunding
ORDER BY TotalFunding DESC
LIMIT 10

```

Tabella 2.12: Top 10 ambiti di ricerca della Federico II per somma finanziata (caso specifico) - Cypher – Neo4j

2.3.1.2 Caso generale

Per il caso generale, viene utilizzata la funzione **\$match** che filtra i documenti, selezionando solo i documenti per i quali il campo FieldsOfResearchANZSRC2020 inizia con 2 cifre.

Le seguenti implementazioni in **Python - MongoDB**, **Cypher - Neo4j**:



```

query_amb_fin_gen = list(research.aggregate([
    {"$project": {"Fields_of_Research_ANZSRC_2020": {"$split": [
        "$Fields_of_Research_ANZSRC_2020", " "; "]}, 
        "Funding_Amount_in_EUR": 1}},
    {"$unwind": "$Fields_of_Research_ANZSRC_2020"}, 
    {"$match": {"Fields_of_Research_ANZSRC_2020": {"$regex": r'^\d{2}'}},
    {"$group": {"_id": {"$trim": {"input": 
        "$Fields_of_Research_ANZSRC_2020"}}, "Somma_Finanziata_Totale": 
        {"$sum": "$Funding_Amount_in_EUR"}}, 
    {"$project": {"_id": 0, "Campi_di_ricerca": "$_id", 
        "Somma_Finanziata_Totale": "$Somma_Finanziata_Totale"}}, 
    {"$sort": {"Somma_Finanziata_Totale": -1}}, # Ordina in ordine 
        decrescente per Somma_Finanziata_Totale
    {"$limit": 10}
]))

```

Tabella 2.13: Top 10 ambiti di ricerca della Federico II per somma finanziata (caso generale) - Python - MongoDB

```
neo4j

MATCH (p:Project)-[:BELONGS_TO]-(a:Ambiti)
WITH a, sum(p.Funding_Amount_in_EUR) as TotalFunding
WHERE a.Field =~ '^\\d{2}\\b.*'
RETURN a.Field as AmbitoRicerca, TotalFunding
ORDER BY TotalFunding DESC
LIMIT 10
```

Tabella 2.14: Top 10 ambiti di ricerca della Federico II per somma finanziata (caso generale) - Cypher – Neo4j

2.3.2 Risultati

Dai risultati nelle figure sottostanti, si può notare che durante gli anni, la Federico II ha finanziato maggiormente i progetti nel settore di Ingegneria, nello specifico **Ingegneria Aerospaziale**, con una somma totale stanziata di ben 637.744.439 euro. A seguire, i macro settori "Information and Computing Sciences", "Biomedical and Clinical Sciences" e "Biological Sciences" occupano le successive posizioni per somma finanziata totale.

Ambito.di.ricerca	Somma_Finanziata.Totale
4001 Aerospace Engineering	637744439
3505 Human Resources and Industrial Relations	203378135
4608 Human-Centred Computing	190370338
3101 Biochemistry and Cell Biology	146432394
3105 Genetics	123547479
4010 Engineering Practice and Education	103466064
5202 Biological Psychology	102547016
4606 Distributed Computing and Systems Software	101484544
4601 Applied Computing	89847708
4008 Electrical Engineering	88425407

Tabella 2.15: Top 10 ambiti di ricerca della Federico II per somma finanziata (caso specifico)

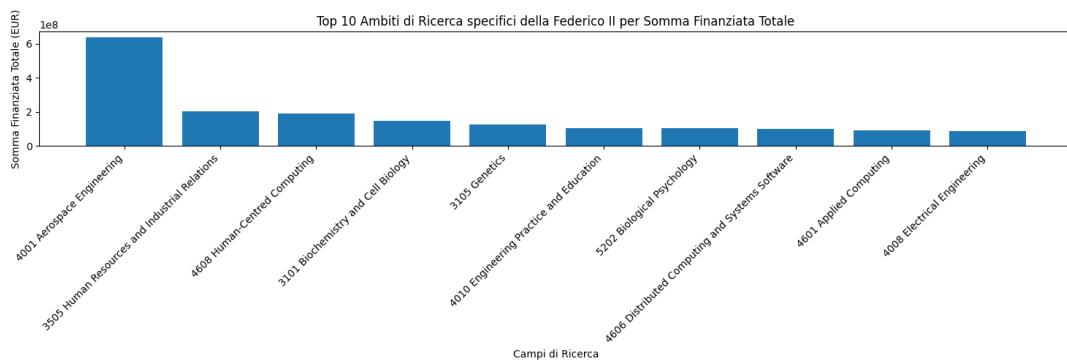


Figura 2.10: Bar Chart - Top 10 ambiti di ricerca della Federico II per somma finanziata (caso specifico)

Nella successiva Figura, si vuole mostrare come variano, nel corso degli anni, i progetti per ciascun Topic ed ambito di ricerca in base a quelli maggiormente finanziati attraverso un **Bar Chart Race**.

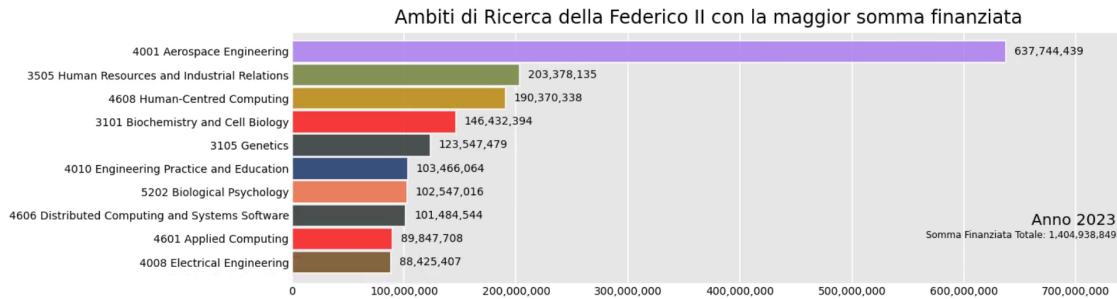


Figura 2.11: Bar Chart Race - Top 10 ambiti di ricerca della Federico II per somma finanziata (caso specifico)

Si noti: La Figura sovrastante mostrata è stata catturata all'interno di un video, che mostra l'andamento dei progetti, per ciascun Ambito di ricerca, maggiormente finanziati. Per la visione del video completo si rimanda al seguente link:

<https://shorturl.at/NXY59>

Nel **plot successivo** viene evidenziato ulteriormente il caso degli **ambiti di ricerca generici** in accordo con la maggior somma finanziata dall'Aerospace Engineering.

Ambito_di_Ricerca	Somma_Finanziata_Totale
40 Engineering	1278322446
46 Information and Computing Sciences	654215909
32 Biomedical and Clinical Sciences	438764614
31 Biological Sciences	414934151
35 Commerce, Management, Tourism and Services	311740252
30 Agricultural, Veterinary and Food Sciences	235068490
41 Environmental Sciences	196804433
34 Chemical Sciences	142652652
33 Built Environment and Design	127413609
37 Earth Sciences	124966240

Tabella 2.16: Top 10 ambiti di ricerca della Federico II per somma finanziata (caso generale)

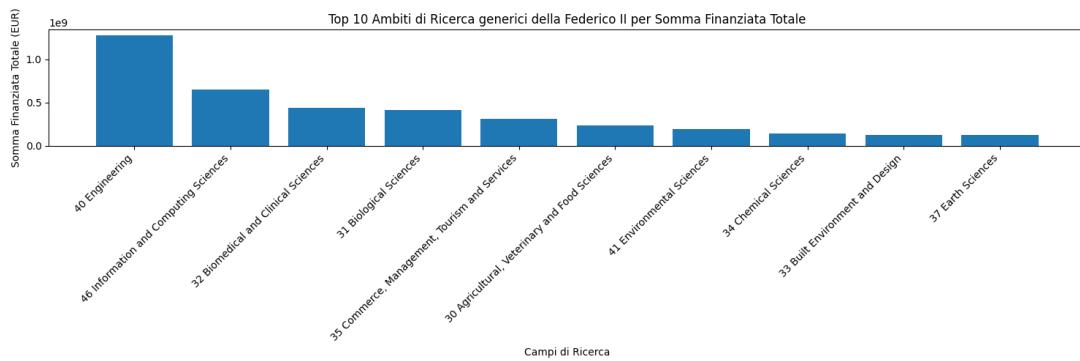


Figura 2.12: Bar Chart - Top 10 ambiti di ricerca della Federico II per somma finanziata (caso generale)

2.3.2.1 Esplorazione degli ambiti con cui ha maggiormente collaborato 'Ingegneria'

Nel seguente grafo si mostrano gli ambiti con cui l'ambito di ricerca "Engineering" ha maggiormente collaborato. Una grossa affluenza di progetti di ricerca la si può notare con la collaborazione tra Ingegneria e l'ambito di ricerca "Chemical Sciences" ed "Built Environment and Design".

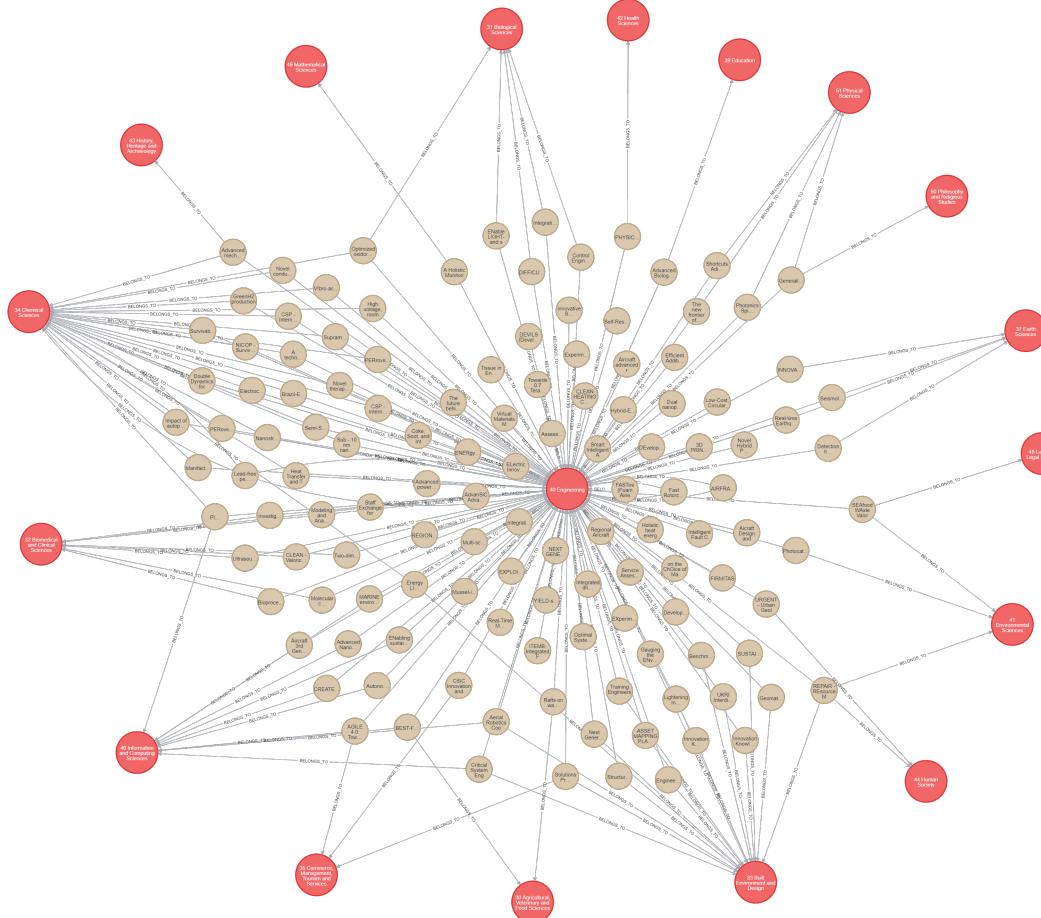


Figura 2.13: Grafo - Esplorazione degli ambiti con cui ha maggiormente collaborato 'Ingegneria'

2.3.2.2 Esplorazione degli ambiti con cui ha maggiormente collaborato 'Ingegneria AereaSpaziale'

Nel seguente grafo si mostrano gli ambiti con cui l'ambito di ricerca "Aerospace Engineering" ha maggiormente collaborato.

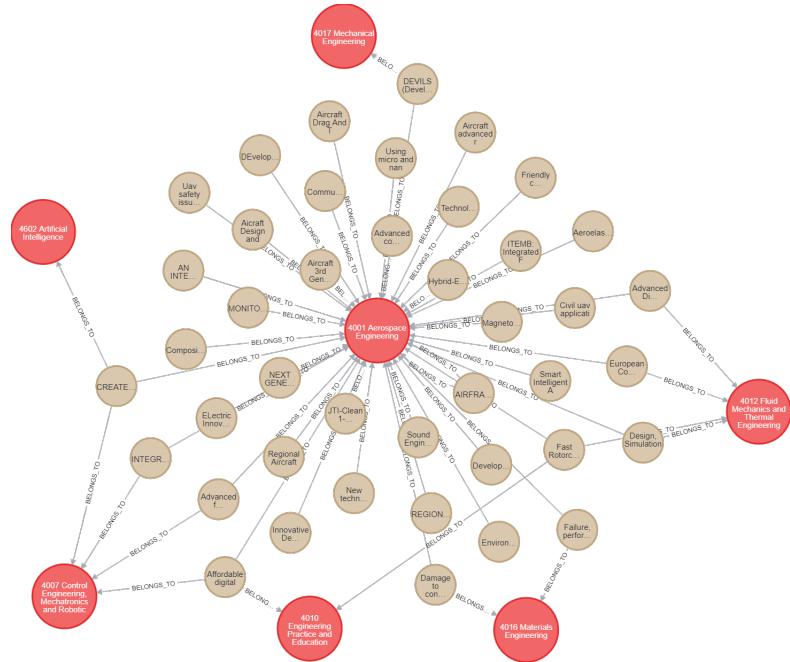


Figura 2.14: Grafo - Esplorazione degli ambiti con cui ha maggiormente collaborato 'Ingegneria Aeroospaziale'

Nonostante l'ambito di Aerospace Engineering veda la collaborazione con altri sei di ambiti, si nota da subito come è presente una collaborazione più frequente negli ambiti di ricerca di "Control Engineering, Mechatronics and Robotics (4007)" e "Fluid Mechanics and Thermal Engineering (4012)", con quattro progetti di ricerca per ciascun ambito:

- Per la collaborazione tra “Aerospace Engineering” e “Control Engineering, Mechatronics and Robotics”, si evidenziano progetti di ricerca incentrati sullo sviluppo di tecnologie autonome affidabili per migliorare le operazioni che si affidano ai sistemi di aeromobili senza pilota, sistemi di controllo di volo digitali economici, integrazione e sperimentazione di sistemi di navigazione autonomi, progetti per migliorare l'autonomia e la sicurezza del volo, migliorando il controllo attivo del rumore negli aeromobili;
 - Per la collaborazione tra “Aerospace Engineering” e “Fluid Mechanics and Thermal Engineering”, si evidenziano progetti di ricerca incentrati sullo sviluppo di elicotteri o droni veloci, migliorando la progettazione dei rotori e la dinamica del volo, diagnostica i flussi d'aria nei progetti aeronautici, soluzioni avanzate di alta portanza per migliorare l'efficienza del volo, l'ottimizzazione dell'efficienza delle celle solari multicristalline in silicio.

2.4 Wordcount dei topic dei primi 5 ambiti di ricerca della Federico II per numero progetti

L'obiettivo della quarta Analytic è quello di effettuare, per i primi 5 ambiti di ricerca specifici della Federico II (caratterizzati da un codice identificativo a 4 cifre) trovati dalla Analytic 1, il wordcount sul topic di ricerca estratto nella fase Raccolta dati a partire dal titolo del progetto con le **API di OpenAI**. Diverse sono le query sulla quale viene effettuato tale Wordcount, verrà mostrata solamente quella legata al topic **3211 Oncology and Carcinogenesis**, ma la stessa implementazione è stata effettuata anche per i seguenti Topic:

- **3105 Genetics;**
- **3101 Biochemistry and Cell Biology;**
- **3202 Clinical Sciences;**
- **4303 Historical Studies;**

Verranno tuttavia mostrate le immagini dei 5 ambiti di ricerca per numero progetto nella sezione **Risultati**.

In particolare, le parole più frequenti verranno visualizzate in caratteri più grandi, mentre quelle meno frequenti verranno visualizzate in caratteri più piccoli.

2.4.1 Implementazione

Nell'implementazione:

- Viene eseguita la query di aggregazione sulla collezione **research**;
- Viene utilizzata la funzione **\$project** sui dati di ingresso. Vengono selezionati i campi **FieldsOfResearchANZSRC2020** e **topic**, ciascuno dei quali viene diviso in un array tramite la funzione **\$split** utilizzando i delimitatori ";" e " " rispettivamente;
- Viene utilizzata la funzione **\$unwind** due volte, una volta per scomporre l'array **FieldsOfResearchANZSRC2020** e una volta per scomporre l'array **Topic.words** in documenti separati;
- Viene utilizzata la funzione **\$match** tre volte consecutivamente per filtrare i documenti. La prima per selezionare i documenti per i quali il campo **FieldsOfResearchANZSRC2020** corrisponde esattamente a "3101 Biochemistry and Cell Biology". La seconda e la terza per eliminare i caratteri speciali e le parole che si trovano nella lista **english_stopwords**;
- Viene utilizzata la funzione **\$group** per raggruppare i documenti in base alla parola del topic (convertita in minuscolo) e contando il numero di volte che ogni parola appare;
- Viene utilizzata la funzione **\$project** per rinominare il campo **_id** in **Word** e per mantenere il campo **Count**;
- Viene utilizzata la funzione **\$sort** per ordinare i documenti in ordine decrescente per il campo **Count**;
- Infine, viene utilizzata la funzione **\$limit** per limitare il numero di documenti restituiti a 20.

Di seguito l'implementazione in **Python - MongoDB**:

```
mongodB MongoDB

query_wc_Onc = list(research.aggregate([
    {"$project": {"Fields_of_Research_ANZSRC_2020": {"$split": [
        ["$Fields_of_Research_ANZSRC_2020", " ", ""]}, "Topic_words": {
        {"$split": ["$topic", " "]}}},
    {"$unwind": "$Fields_of_Research_ANZSRC_2020", "$unwind": "$Topic_words"}, 
    {"$match": {"Fields_of_Research_ANZSRC_2020": "3211 Oncology and Carcinogenesis"}},
    {"$match": {"Topic_words": {"$nin": list(english_stopwords)}}}, # Filtra le parole che non sono nelle stopwords
    {"$match": {"Topic_words": {"$nin": [& ',', ',']}}},
    {"$group": {"_id": {"$toLower": "$Topic_words"}, "count": {"$sum": 1}}},
    {"$project": {"_id": 0, "Word": "$_id", "Count": "$count"}},
    {"$sort": {"Count": -1}},
    {"$limit": 20}
]))
```

Tabella 2.17: Wordcount dei topic dei primi 5 ambiti di ricerca della Federico II per numero progetti - Python - MongoDB

2.4.2 Risultati

Vengono di seguito rappresentate le parole più frequenti per il topic **3211 Oncology and Carcinogenesis**.

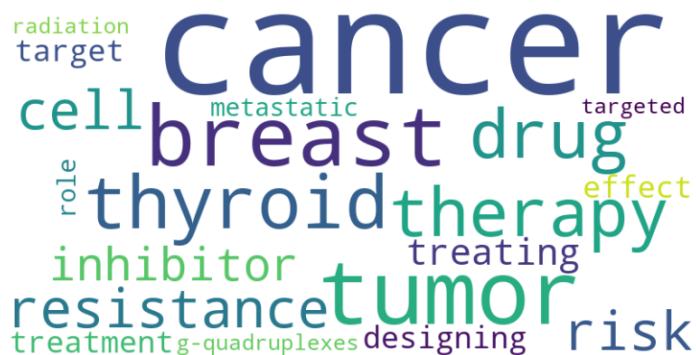


Figura 2.15: WordCount - Top 5 ambiti di ricerca della Federico II per numero progetti, Topic **3211 Oncology and Carcinogenesis**

In accordo con il wordcloud precedente, data la sua rilevanza nell'ambito medico-scientifico, è naturale che molti progetti di ricerca in ambito "Oncology e Carcinogenesis" si concentrino sullo studio del cancro e sullo sviluppo di nuove terapie, diagnostica e approcci preventivi; si possono anche notare i tipi di cancro più trattati dalla Federico II siano quelli alla tiroide ed al seno.

Come si è anticipato inizialmente, vengono mostrate anche le parole più frequenti per i Topic **3105 Genetics**, **3101 Biochemistry and Cell Biology**, **3202 Clinical Sciences**, **4303 Historical Studies**.

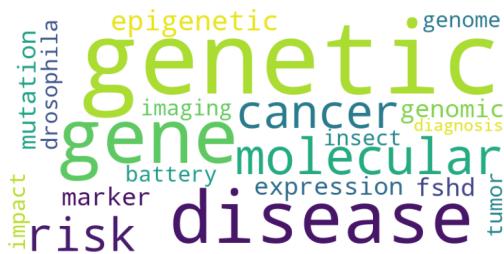


Figura 2.16: Wordcount dei topic dei primi 5 ambiti di ricerca della Federico II per numero progetti, **Topic 3105 Genetics**

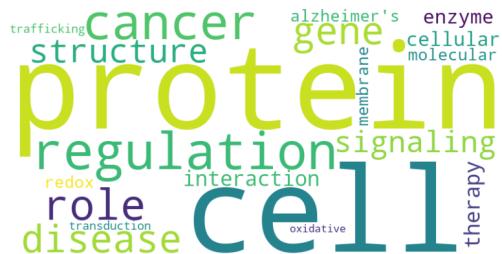


Figura 2.17: Wordcount dei topic dei primi 5 ambiti di ricerca della Federico II per numero progetti, **Topic 3101 Biochemistry and Cell Biology**

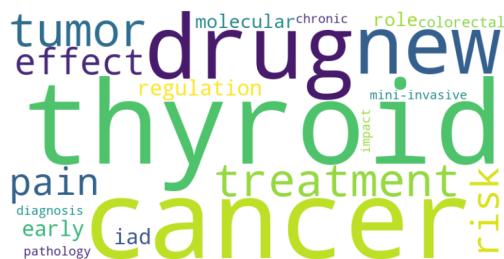


Figura 2.18: Wordcount dei topic dei primi 5 ambiti di ricerca della Federico II per numero progetti, Topic **3202 Clinical Sciences**



Figura 2.19: Wordcount dei topic dei primi 5 ambiti di ricerca della Federico II per numero progetti, Topic **4303 Historical Studies**

Dei seguenti topic, analizzando in dettaglio le argomentazioni sulla base dei wordcount ottenuti nelle Figure precedenti, è possibile affermare che:

- “3105 Genetics” si focalizza sulla genetica, con un particolare interesse per le malattie di tipo “genetic” ed il cancro. Termini come “epigenetic”, “marker”, “mutation” suggeriscono che ci si concentrati su dettagli tecnici della genetica;
 - “3101 Biochemistry and Cell Biology” si concentra sull’intersezione tra la biochimica e la biologia cellulare, ambiti che trattano le proteine (“protein”) sul come possano regolare la funzione cellulare. Il cancro, come anche l’alzheimer, sembrano essere un argomento di interesse, come suggerisce il numero di volte in cui appare la parola “cancer” ed “alzheimer”. Altri termini come “signaling”, “trafficking”, “transduction”, suggeriscono discussioni su processi cellulari specifici;
 - “3202 Clinical Sciences” si concentra su ricerche cliniche nel campo della medicina, con particolare attenzione a “cancer”, “thyroid” e i principali “risk”. L’accento sulle parole “drug” e “treatment” suggerisce un focus sui metodi terapeutici;
 - “4303 Historical Studies” affronta studi della storia italiana (“italian” “history”), con riferimenti specifici alla storia antica greca e romana. Parole come “medieval”, “century”, “papyrus” suggeriscono ricerche sulla storia attraverso varie epoche. La presenza di “restoration” potrebbe indicare un interesse per i periodi di cambiamento o recupero nel corso della storia.

2.5 Linea temporale (per ogni anno) con evoluzione dei progetti iniziati per ogni ambito di ricerca della Federico II

L'obiettivo della quinta Analytic è quello di mostrare una linea/andamento temporale, per anno, del numero dei progetti per i primi 5 ambiti di ricerca specifici (caratterizzati da un codice identificativo a 4 cifre) appartenenti agli ambiti con più progetti trovati nella prima Analytic.

2.5.1 Implementazione

Nell'implementazione:

- Viene eseguita la query di aggregazione sulla collezione **research**;
- Viene utilizzata la funzione **\$match** per filtrare i documenti che hanno un valore non nullo nel campo Start_Year;
- Viene utilizzata la funzione **\$project** per selezionare i campi Start_Year e FieldsOfResearchANZSRC2020, quest'ultimo viene diviso in un array tramite la funzione **\$split** utilizzando il delimitatore ";";
- Viene utilizzata la funzione **\$unwind** per scomporre l'array FieldsOfResearchANZSRC2020 in documenti separati;
- Vengono eseguite due funzioni **\$match**. La prima seleziona i documenti per i quali il campo FieldsOfResearchANZSRC2020 inizia con 4 cifre. La seconda seleziona i documenti per i quali il campo FieldsOfResearchANZSRC2020 si trova in un elenco specifico di campi di ricerca;
- Viene utilizzata la funzione **\$group** per raggruppare i documenti in base all'anno di inizio e al campo di ricerca, e contando il numero di progetti per ciascuna combinazione;
- Viene utilizzata la funzione **\$project** per rinominare i campi `_id`, `Year`, `id.FieldsOfResearchANZSRC2020` in `FieldsOfResearchANZSRC2020` e mantenendo il campo `Projects_Count`;
- Infine, viene utilizzata la funzione **\$sort** per ordinare i documenti per anno e campo di ricerca;

La seguente implementazione in **Python - MongoDB**:



MongoDB

```

queryTemporalProject = list(research.aggregate([
    {"$match": {"Start_Year": {"$ne": None}}},
    {"$project": {"Start_Year": "$Start_Year",
                  "Fields_of_Research_ANZSRC_2020": {"$split": ["$Fields_of_Research_ANZSRC_2020", " ; "]}},
    {"$unwind": "$Fields_of_Research_ANZSRC_2020"},
    {"$match": {"Fields_of_Research_ANZSRC_2020": {"$in": [3101
        "Biochemistry and Cell Biology", 3105 Genetics", "3211 Oncology
        and Carcinogenesis", "3202 Clinical Sciences", "4303 Historical
        Studies"]}}},
    {"$group": {"_id": {"Year": "$Start_Year",
                      "Fields_of_Research_ANZSRC_2020": "$Fields_of_Research_ANZSRC_2020"},
               "Projects_Count": {"$sum": 1}}},
    {"$project": {"_id": 0, "Year": "_id.Year",
                  "Fields_of_Research_ANZSRC_2020": "$_id.Fields_of_Research_ANZSRC_2020",
                  "Projects_Count": "$Projects_Count"}},
    {"$sort": {"Year": 1, "Fields_of_Research_ANZSRC_2020": 1}}
]))

```

Tabella 2.18: Linea temporale (per ogni anno) con evoluzione dei progetti effettuati per ogni ambito di ricerca della Federico II - Python - MongoDB

2.5.2 Risultati

Nella Figura sottostante, viene mostrata la linea temporale che raffigura l'evoluzione dei progetti, (indicandone il numero sull'Asse Y), effettuati per ogni ambito di ricerca della Federico II per ogni anno (Asse X).

In particolare, si possono notare alcune tendenze relative ai progetti di ricerca condotti all'Università Federico II relative ai seguenti ambiti di ricerca specifici:

- “3101 Biochemistry and Cell Biology”: oltre ad essere più popolare, in termini di numero di progetti condotti in tutti gli anni considerati, rappresenta l’ambito in cui c’è un aumento significativo dei progetti, a partire dal 1999, raggiungendo un picco di 22 progetti nel 2004;
- “3105 Genetics”: compare per la prima volta nel 1991 e dal 1999 in poi presenta un aumento dei progetti, con un picco di 15 progetti nel 2004;
- “3202 Clinical Sciences”: introdotto come ambito di ricerca nel 1993, sembra essere cresciuto in popolarità negli anni 2000. Tuttavia, il numero di progetti in questo ambito rimane relativamente basso rispetto ad altri campi;
- “3211 Oncology and Carcinogenesis”: introdotto nel 1996, sembra avere un andamento in crescita negli anni 2000, con un picco di 16 progetti nel 2020;

- "4303 Historical Studies": introdotto come ambito di ricerca nel 1998, sembra essere presente in modo costante con almeno un progetto all'anno, con picchi di 5-6 progetti tra il 1998 e 2001/2002, il cui massimo picco è stato raggiunto nel 2003 con 8 progetti.

Si noti come dopo il 2009 avvenga una diminuzione del numero di progetti in tutti gli ambiti. Ciò può esser dovuto ad una riduzione dei finanziamenti o dei ricercatori in quegli ambiti a favore di altri.

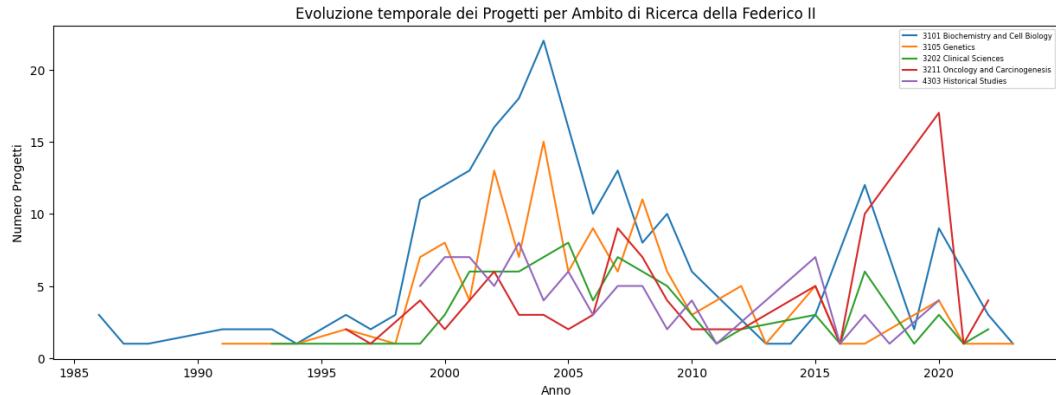


Figura 2.20: Linea temporale (per ogni anno) con evoluzione dei progetti effettuati per ogni ambito di ricerca della Federico II

2.6 Top 5 Progetti con maggior interdisciplinarità tra Ambiti di Ricerca della Federico II

L'obiettivo della sesta analytic è quello di effettuare una verifica dei progetti che coinvolgono più ambiti di ricerca. Questi progetti possono aiutare a identificare le intersezioni tra discipline diverse e promuovere la collaborazione tra ricercatori provenienti da ambiti diversi. Ciò può portare a una maggiore integrazione delle conoscenze e ad approcci più completi per affrontare le sfide complesse.

2.6.1 Implementazione

Nell'implementazione:

- **MATCH:** La clausola MATCH viene utilizzata per trovare i nodi di tipo "Project" (rappresentati dalla variabile p) che hanno una relazione di tipo "BELONGS_TO" (rappresentata dalla variabile r) con i nodi di tipo "Ambiti" (rappresentati dalla variabile a).
- **WHERE:** La clausola WHERE viene utilizzata per filtrare i nodi "Ambiti" in base a una condizione. In questo caso, la condizione verifica che la proprietà "Field" dei nodi "Ambiti" inizi con un numero di quattro cifre con l'uso dell'espressione regolare "d4" e deve corrispondere a una parola completa.
- **RETURN:** La clausola RETURN specifica quali dati restituire come risultato della query. In questo caso, vengono restituiti i nodi "Project" (rappresentati dalla variabile p), un insieme raccolto dei nodi "Ambiti" (rappresentati dalla variabile a e rinominati come AmbitiTitoli), e il conteggio dei nodi "Ambiti" distinti (rappresentato come NumAmbiti).
- **ORDER BY:** La clausola ORDER BY viene utilizzata per ordinare i risultati in base al numero di ambiti in ordine decrescente. Ciò significa che i progetti con il numero maggiore di ambiti saranno elencati per primi.

- **LIMIT:** La clausola LIMIT limita il numero di risultati restituiti. In questo caso, vengono restituiti solo i primi 5 risultati, ovvero i progetti con il maggior numero di ambiti.

```
neo4j
MATCH (p:Project)-[r:BELONGS_TO]->(a:Ambiti)
WHERE a.Field =~ '^\\d{4}\\b.*'
RETURN p, collect(distinct(a)) AS AmbitiTitoli, COUNT(distinct(a)) AS
      NumAmbiti
ORDER BY NumAmbiti DESC
LIMIT 5
```

Tabella 2.19: Top 5 Progetti in cui sono stati coinvolti più Ambiti di Ricerca della Federico II - Cypher – Neo4j

2.6.2 Risultati

Il grafo nella Figura sottostante mostra i 5 progetti con più coinvolgimenti di ambiti di ricerca. In particolare al massimo si sono avute 4 collaborazioni tra ambiti sullo stesso progetto.

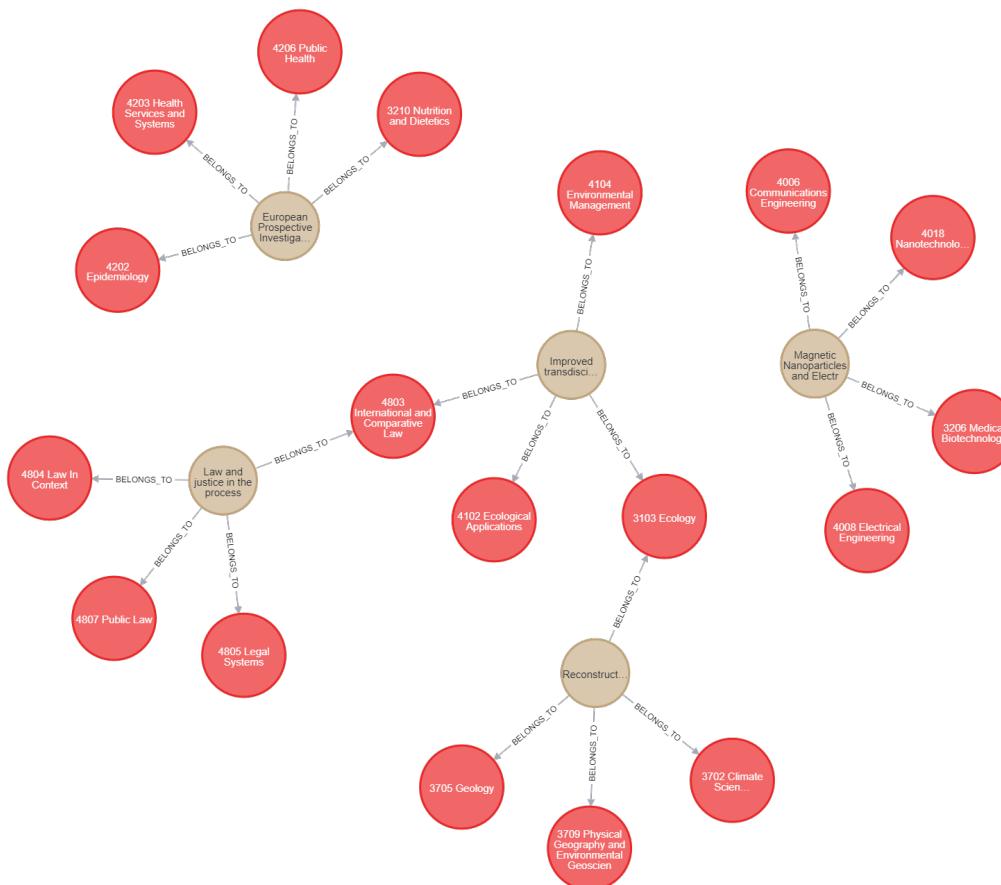


Figura 2.21: Grafo - Top 5 Progetti in cui sono stati coinvolti più Ambiti di Ricerca della Federico II

Ciò permette di capire quali progetti sono correlati ad un maggior numero di ambiti di ricerca distinti. Un esempio di cui è possibile effettuare una breve descrizione, è quello legato al progetto di "European Prospective Investigation". Tale progetto infatti coinvolge principalmente l'ambito di ricerca generico delle "Health Sciences (42)", ed in particolare i rispettivi ambiti di ricerca specifici:

- 4203 Health Services and Systems;
- 4206 Public Health;
- 4202 Epidemiology;
- 3210 Butrition and Dietetics, facente parte dell'ambito di ricerca generico delle "Biomedical and Clinical Sciences (32)".

2.7 Esecuzione definitiva: MongoDB - Neo4j

Si lascia il lettore ad eseguire il seguente notebook Python per MongoDB:

MongoDB

<https://github.com/giuseppericcio/BigData/blob/main/HW2/MongoDB/HW2-Mongo.ipynb>

Per gli script di Neo4J, si allega il link su GitHub:

Neo4j

<https://github.com/giuseppericcio/BigData/tree/main/HW2/Neo4j>

È possibile inoltre visualizzare il report realizzato con **Streamlit** al seguente link:

<https://progettiunina.streamlit.app>

3 Conclusioni e Confronto tra le piattaforme

Contenuti

3.1 Il Confronto	36
3.2 Conclusioni	37

3.1 Il Confronto

Per confrontare le piattaforme utilizzate nella seguente trattazione si possono considerare i seguenti fattori:

- **Modello dei dati:**

- **MongoDB:** utilizza un modello di dati orientato ai documenti. I dati in esame sono organizzati in collezioni di documenti BSON (Binary JSON), che possono contenere strutture annidate e campi multipli. Pertanto risulta essere flessibile e scalabile per gestire questi dati semi-strutturati;
- **Neo4j:** utilizza un modello di dati basato sul grafo. I dati sono organizzati in nodi, relazioni e proprietà.

- **Scalabilità:**

- **MongoDB:** offre scalabilità orizzontale, consentendo di distribuire i dati su più server in un cluster e di gestire volumi elevati di dati. Nel caso in esame, i dati sono distribuiti su shard collocati a Francoforte:



Figura 3.1: Sharding del dataset in esame

- **Neo4j:** offre scalabilità orizzontale solo fino a un certo punto. Nel caso in esame non si è avvalsi della scalabilità offerta dalla piattaforma.

- **Query e analisi dei dati:**

- **MongoDB:** supporta un'ampia gamma di query, inclusi filtri, proiezioni, aggregazioni e join tra collezioni. Nel caso in esame si è usata una configurazione Python to MongoDB, pertanto si è potuto integrare facilmente funzioni di plotting e altro per una migliore lettura delle analytics;
- **Neo4j:** le query sono di tipo grafo. Pertanto nel caso in esame, essendo che i dati non sono strettamente relazionati, si è limitati a semplici analytics.

- **Uso:**

- **MongoDB:** È adatto per la gestione di grandi quantità di dati semi-strutturati o non strutturati. Pertanto risulta essere particolarmente efficace nell'analisi del caso in esame;

- **Neo4j:** essendo che è ampiamente utilizzato in scenari che richiedono l'analisi di dati connessi, come social network, raccomandazioni personalizzate, rilevamento di frodi, analisi delle reti e knowledge graph.

Entrambe le soluzioni offrono una configurazione ed installazione semplice.

3.2 Conclusioni

Dati i confronti effettuati nel sottoparagrafo precedente e sulla base delle analytics presentate, ai fini dello svolgimento dell'homework, è risultato più flessibile ed estendibile la piattaforma e il framework **MongoDB** in quanto:

- è risultato più **efficace** nella esposizione ed estrazioni delle analytics a partire dal dataset in esame;
- le **funzionalità** di MongoDB si **adattano** meglio al **tipo di dati e alle esigenze analitiche** del caso d'uso in esame;
- grazie alle sue proprietà (precedentemente enunciate), MongoDB ha permesso un'organizzazione efficiente dei dati in collezioni di documenti BSON, facilitando l'esecuzione delle query;
- grazie all'**integrazione tra Python e MongoDB**, viene facilitata l'implementazione di funzioni di plotting e altri strumenti di analisi, contribuendo ad un'interpretazione più chiara e significativa delle analytics.

Neo4j, pur essendo una piattaforma efficace per l'analisi di dati, non si è dimostrato ottimale per il nostro caso d'uso. Le sue funzionalità applicate all'estrazione delle sue potenti query di tipo grafo, pur essendo valide, non hanno risposto in modo adeguato alle esigenze specifiche del nostro dataset.

Da precisare che entrambe le piattaforme hanno vantaggi e svantaggi; in questo caso specifico, **MongoDB si è dimostrato essere più flessibile e intuitivo**.

Elenco delle figure

1.1 Metagrafo - Visualizzazione schema database, nodi e relazioni	8
2.1 Bar Chart - Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso specifico)	13
2.2 Bar Chart - Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso generale)	13
2.3 Bar Chart Race - Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso specifico)	14
2.4 Grafo - Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso specifico)	15
2.5 Grafo - Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso generale)	16
2.6 Grafo - Esplorazione degli obiettivi sostenibili affrontati da 'Biological Sciences'	17
2.7 Grafo - Esplorazione degli obiettivi sostenibili affrontati da 'Engineering'	18
2.8 Objective Cycle dei progetti con obiettivi di sostenibilità affrontati dalla Federico II	20
2.9 Grafo - Top 10 obiettivi sostenibili degli ambiti di ricerca della Federico II con il maggior numero dei progetti	21
2.10 Bar Chart - Top 10 ambiti di ricerca della Federico II per somma finanziata (caso specifico)	24
2.11 Bar Chart Race - Top 10 ambiti di ricerca della Federico II per somma finanziata (caso specifico)	25
2.12 Bar Chart - Top 10 ambiti di ricerca della Federico II per somma finanziata (caso generale)	26
2.13 Grafo - Esplorazione degli ambiti con cui ha maggiormente collaborato 'Ingegneria'	26
2.14 Grafo - Esplorazione degli ambiti con cui ha maggiormente collaborato 'Ingegneria Aereospaziale'	27
2.15 WordCount - Top 5 ambiti di ricerca della Federico II per numero progetti, Topic 3211 Oncology and Carcinogenesis	29
2.16 Wordcount dei topic dei primi 5 ambiti di ricerca della Federico II per numero progetti, Topic 3105 Genetics	30
2.17 Wordcount dei topic dei primi 5 ambiti di ricerca della Federico II per numero progetti, Topic 3101 Biochemistry and Cell Biology	30
2.18 Wordcount dei topic dei primi 5 ambiti di ricerca della Federico II per numero progetti, Topic 3202 Clinical Sciences	30
2.19 Wordcount dei topic dei primi 5 ambiti di ricerca della Federico II per numero progetti, Topic 4303 Historical Studies	30
2.20 Linea temporale (per ogni anno) con evoluzione dei progetti effettuati per ogni ambito di ricerca della Federico II	33
2.21 Grafo - Top 5 Progetti in cui sono stati coinvolti più Ambiti di Ricerca della Federico II	34
3.1 Sharding del dataset in esame	36

Elenco delle tabelle

1.1 Generazione nuova colonna "Topic" a partire dalla colonna "Title Translated"	3
1.2 Fase di Lemmatizing	3
1.3 Workflow: Raccolta dati - Python - MongoDB Atlas	6
1.4 Workflow: Raccolta dati - Cypher – Neo4j	8
2.1 Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso specifico) - Python - MongoDB	10
2.2 Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso specifico) - Cypher – Neo4j	11
2.3 Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso generale) - Python - MongoDB	11
2.4 Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso generale) - Cypher – Neo4j	12
2.5 Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso specifico)	12
2.6 Top 10 degli ambiti di ricerca della Federico II con il maggior numero dei progetti (caso generale)	13
2.7 Esplorazione degli obiettivi sostenibili affrontati da 'Biological Sciences' - Cypher – Neo4j	17
2.8 Esplorazione degli obiettivi sostenibili affrontati da 'Engineering' - Cypher – Neo4j	18
2.9 Objective cycle dei progetti con obiettivi di sostenibilità affrontati dalla Federico II - Python - MongoDB	19
2.10 Top 10 obiettivi sostenibili degli ambiti di ricerca della Federico II con il maggior numero dei progetti - Cypher – Neo4j	20
2.11 Top 10 ambiti di ricerca della Federico II per somma finanziata (caso specifico) - Python - MongoDB	22
2.12 Top 10 ambiti di ricerca della Federico II per somma finanziata (caso specifico) - Cypher – Neo4j	23
2.13 Top 10 ambiti di ricerca della Federico II per somma finanziata (caso generale) - Python - MongoDB	23
2.14 Top 10 ambiti di ricerca della Federico II per somma finanziata (caso generale) - Cypher – Neo4j	24
2.15 Top 10 ambiti di ricerca della Federico II per somma finanziata (caso specifico)	24
2.16 Top 10 ambiti di ricerca della Federico II per somma finanziata (caso generale)	25
2.17 Wordcount dei topic dei primi 5 ambiti di ricerca della Federico II per numero progetti - Python - MongoDB	29
2.18 Linea temporale (per ogni anno) con evoluzione dei progetti effettuati per ogni ambito di ricerca della Federico II - Python - MongoDB	32
2.19 Top 5 Progetti in cui sono stati coinvolti più Ambiti di Ricerca della Federico II - Cypher – Neo4j	34

Bibliografia

[1] Autori del seguente homework. *Slide del corso.* 2023.