МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

им. Н.Э. Баумана

Факультет «Информатика и системы управления»

Кафедра «Систем обработки информации и управления»

ОТЧЕТ

**Лабораторная работа №\_\_4\_\_**

по дисциплине«Технология обработки больших данных»

Тема: «Анализ графов в СУБД NebulaGraph»

ИСПОЛНИТЕЛЬ: \_\_Журавлев Н. В.\_\_

ФИО

группа ИУ5-44М \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись

"24"\_\_апреля\_\_\_2025 г.

ПРЕПОДАВАТЕЛЬ: \_\_\_\_Сухобоков А.В.\_\_\_\_\_

ФИО

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись

"24"\_апреля\_\_\_2025 г.

Москва - 2025

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Описание задачи и алгоритма

1. Создание схемы данных для анализа

Для работы с графом необходимо объявить его схему. Для этого воспользуемся графическим редактором схемы:

1. Открываем вкладку боковой панели Graph и нажимаем Create Graph Type

2. Задаем название и создаем типы узлов и ребер:

3. Создадим граф с только что объявленным типом

2. Импорт данных

Для загрузки данных для анализа воспользуемся инструментом Import. Он поддерживает csv файлы размером до 400МБ и позволяет гибко настроить отображение колонок csv файла в атрибуты выбранной схемы данных

Для каждого из файлов нужно будет сделать отдельную загрузку, т.к. в рамках импорта система делает проверки целостности, поэтому сначала нужно импортировать узлы, а после этого – ребра.

1. Заходим на вкладку Import –> Import Data

2. Загружаем файл scientists.csv, вверху страницы выбираем загрузку данных в граф, который вы создали на предыдущем этапе.

3. На странице конфигурирования файла выбираем опцию «With Header»

4. На странице Mapping выбираем, в какой тип сущности загружаем данные и выбираем маппинг колонок CSV в поля модели.

После этого жмем Import.

Этот сценарий надо будет повторить для узлов «publication» и ребер «wrote»!

3. Работа с запросами

На данном этапе мы поработаем с базовыми запросами в NebulaGraph.

Переходим на вкладку Query.

Давайте ответим на несколько базовых вопросов по данным.

1. Вывести всех ученых из указанного университета

USE sc

MATCH (s@scientist)

WHERE s.university = "University\_A"

RETURN s.scientist\_id, s.name, s.university

2. Найти все публикации конкретного ученого

USE sc

MATCH (s@scientist{scientist\_id: 1})-[@wrote]->(p@publication)

RETURN p.publication\_id, p.title, p.pub\_year

3. Подсчитать количество статей у каждого ученого

USE sc

MATCH (s@scientist)-[@wrote]->(p@publication)

RETURN s.scientist\_id, s.name, COUNT(p) AS papers\_count

GROUP BY s

ORDER BY papers\_count DESC

4. Обогащение модели данных

Для анализа соавторств в нашей сети мы создадим дополнительные ребра между учеными, которые показывают их соавторство. Назовем новый тип ребра «coauthored» и добавим ему атрибут joint\_publications: INT64, который покажет количество общих публикаций.

Для редактирования схемы вернемся в Graph -> Graph Type -> science -> Edit Graph Type.

При помощи запроса создадим новые ребра между учеными (все что ниже – единый запрос с указанием следующего этапа через ключевое слово NEXT:

USE SC

MATCH (s1@scientist)-[@wrote]->(p@publication)<-[@wrote]-(s2@scientist)

WHERE s1.scientist\_id <> s2.scientist\_id

RETURN

CASE WHEN s1.scientist\_id < s2.scientist\_id THEN s1 ELSE s2 END AS first\_author,

CASE WHEN s1.scientist\_id < s2.scientist\_id THEN s2 ELSE s1 END AS second\_author,

COUNT(p) AS joint

GROUP BY first\_author, second\_author

NEXT

FILTER joint > 0

RETURN first\_author, second\_author, joint

NEXT

INSERT (first\_author)~[@coauthored{joint\_papers: joint}]~(second\_author)

Проверим, что ребра действительно создались:

use sc

match (s1)-[e@coauthored]-(s2)

return \*

Теперь давайте ответим на несколько дополнительных вопросов по обогащенным данным:

1. Найти всех соавторов ученого с scientist\_id = 2

USE sc

MATCH (s1@scientist{scientist\_id: 2})-[e@coauthored]-(s2@scientist)

RETURN s2.scientist\_id, s2.name, e.joint\_papers as jp

ORDER BY jp DESC

2. Найти 5 самых продуктивных соавторских пар (продуктивность – это количество совместных публикаций)

USE sc

MATCH (s1@scientist)-[e@coauthored]-(s2@scientist)

WHERE s1.scientist\_id <> s2.scientist\_id

RETURN s1.scientist\_id, s2.scientist\_id, e.joint\_papers as jp

ORDER BY jp DESC

LIMIT 5

3. Найти ученых без соавторов

USE sc

MATCH (s@scientist)

WHERE NOT EXISTS (

MATCH (s)-[@coauthored]-()

)

RETURN s.scientist\_id, s.name

4. Найти межвузовские соавторства

USE sc

MATCH (s1@scientist)-[@coauthored]-(s2@scientist)

WHERE s1.university <> s2.university

RETURN s1.scientist\_id, s1.university, s2.scientist\_id, s2.university

5. Проверить гипотезу «6 рукопожатий» через соавторство– кратчайший путь между любыми двумя учеными без циклов всегда не больше 6 ребер.

USE sc

MATCH p = ALL SHORTEST (s1@scientist)-[@coauthored]-{1,7}(s2@scientist)

WHERE s1.scientist\_id <> s2.scientist\_id

RETURN max(LENGTH(p)) group by ()

5. Работа с NebulaGraph Explorer

Для более наглядного анализа переходим по ссылке Explore и создаем новое исследование над нашим графом.

Можно попросить систему показать кусочек графа, чтобы иметь представление о данных.

Доступные инструменты:

1. Запросы к графу

a. Поиск узлов

b. Поиск отношений между узлами

c. Поиск путей в графе

2. Фильтры отображаемых данных

3. Отображение маркеров

В данной работе нас интересует инструмент запросов к графу. Настройками отображений маркеров можно воспользоваться по желанию, чтобы граф сообщал больше полезной информации.

Доступные инструменты:

1. Query Node

Позволяет найти узлы определенного типа на основе значений их атрибутов.

2. Explore Relation

Позволяет отобразить на рабочей области все ребра для выбранного узла(-ов), которые соответствуют определенным правилам (тип, направление ребра, глубина поиска)

3. Query Path

Позволяет найти путь (пути) между двумя узлами в графе. Доступны условия:

1. Тип ребра

2. Направление ребра при поиске пути

3. Количество выводимых путей и их разновидность

4. Алгоритм поиска (наивный, без повторяющихся ребер, без повторяющихся узлов, без повторяющихся узлов кроме первого и последнего)

Используя эти инструменты, ответим на несколько вопросов:

1. Исследование связей конкретного ученого

1. Найдите ученого с scientist\_id = 1 через Query Node.

2. Используя Explore Relation:

a. Покажите всех его соавторов (ребро coauthored).

b. Определите, сколько совместных публикаций у него с каждым соавтором.

3. Визуализируйте граф и ответьте:

a. С каким университетом чаще всего сотрудничает этот ученый?

2. Анализ публикаций ученого

1. Найдите публикации ученого scientist\_id = 5 через Query Node (тип scientist).

2. Используя Explore Relation:

a. Покажите связи wrote между ученым и его публикациями.

b. Отфильтруйте публикации за последние 3 года (самая молодая публикация в датасете – 2023 года).

3. Определите:

a. Какой год был самым продуктивным для этого ученого?

3. Поиск пути между учеными

1. Найдите кратчайший путь между учеными scientist\_id = 1 и scientist\_id = 10 через соавторства:

a. Укажите тип ребра coauthored.

b. Ограничьте длину пути до 6 шагов.

2. Проанализируйте результат:

a. Сколько промежуточных ученых в пути?

b. Есть ли в цепочке ученые из одного университета?

# Ход работы

В начале необходимо создать инсталляцию СУБД рис. 1. Результат представлен на рис. 2

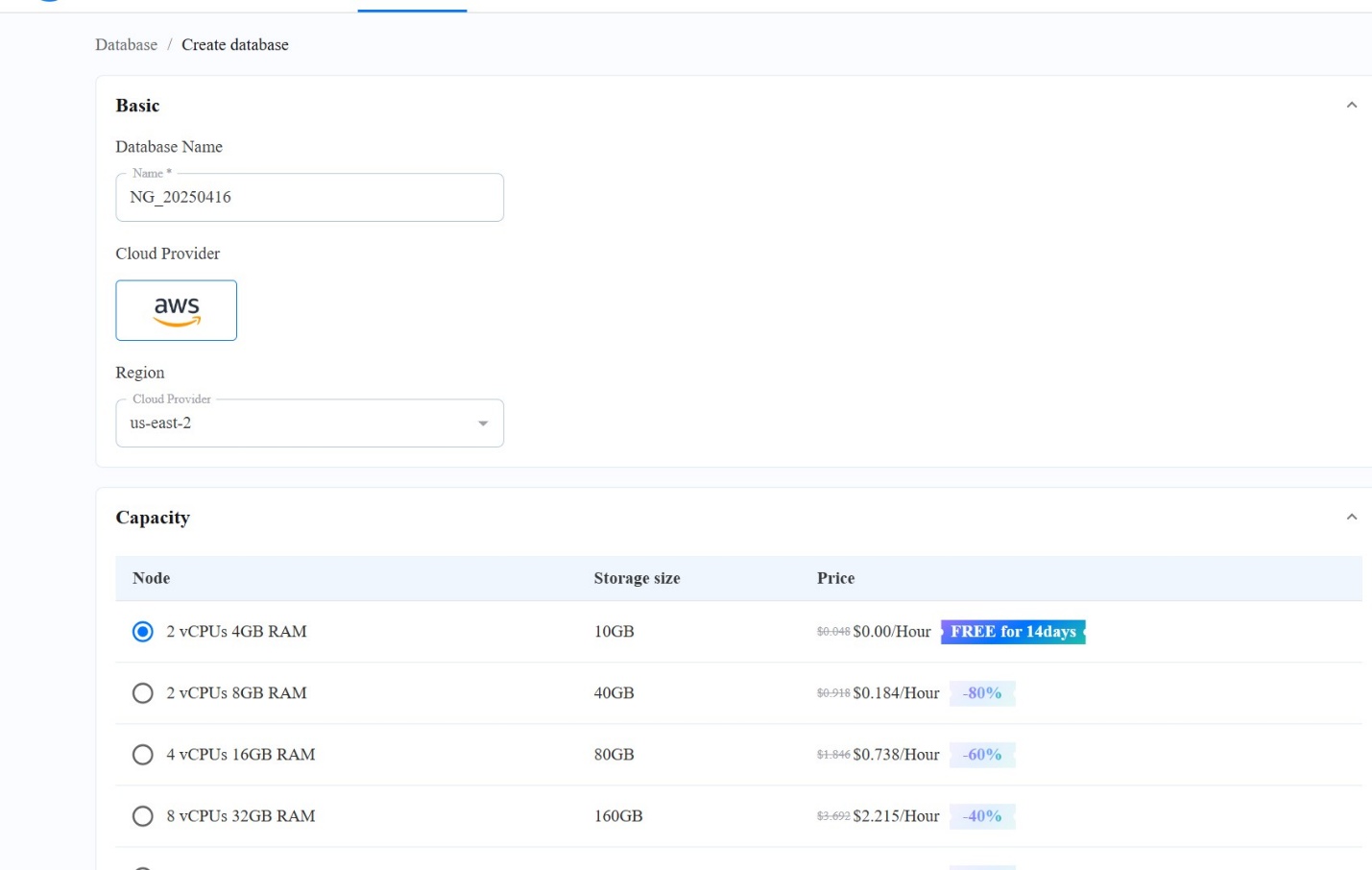


Рисунок . Создание

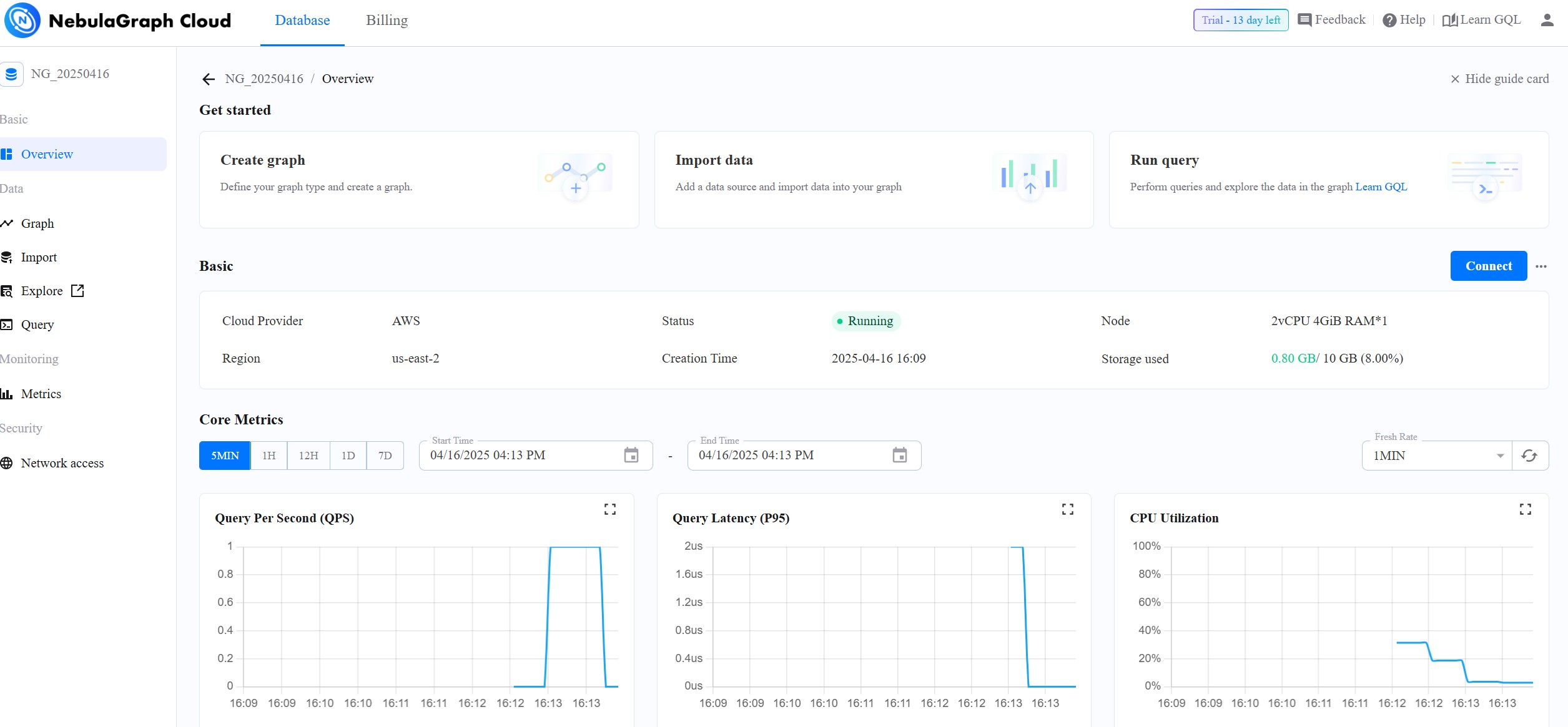


Рисунок . Созданная инсталляция

После создаём граф. Результат представлен на рис. 3.

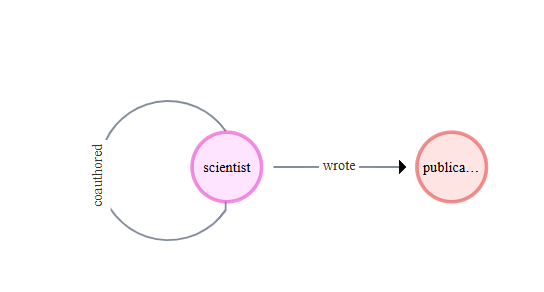


Рисунок . Созданный граф

После заполняем граф. Результат заполнения представлен на рис. 4.

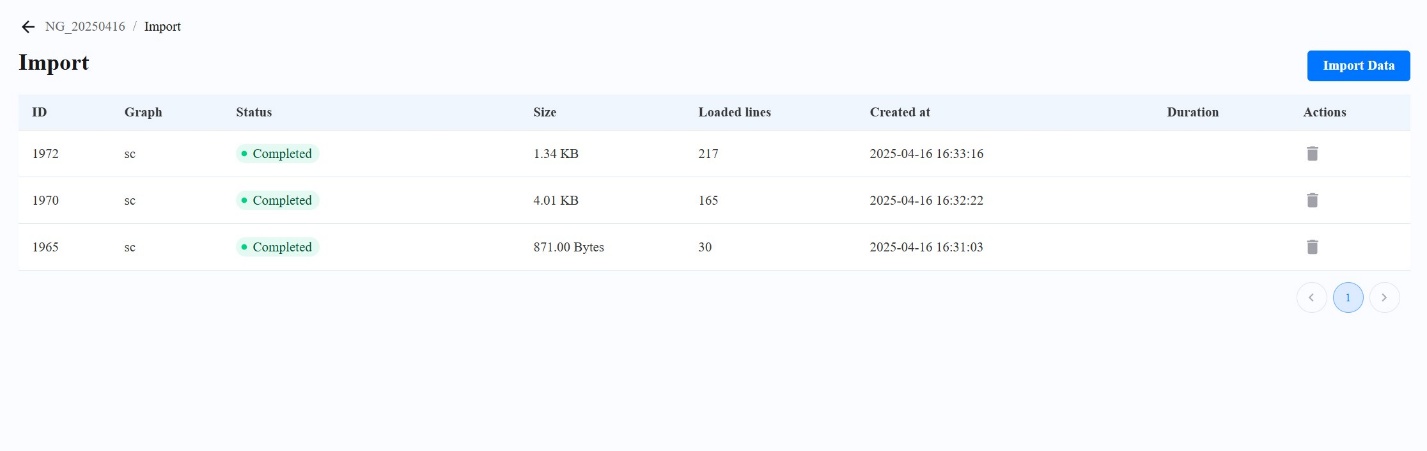


Рисунок . Заполнение

Выведем граф, результат представлен на рис. 5.

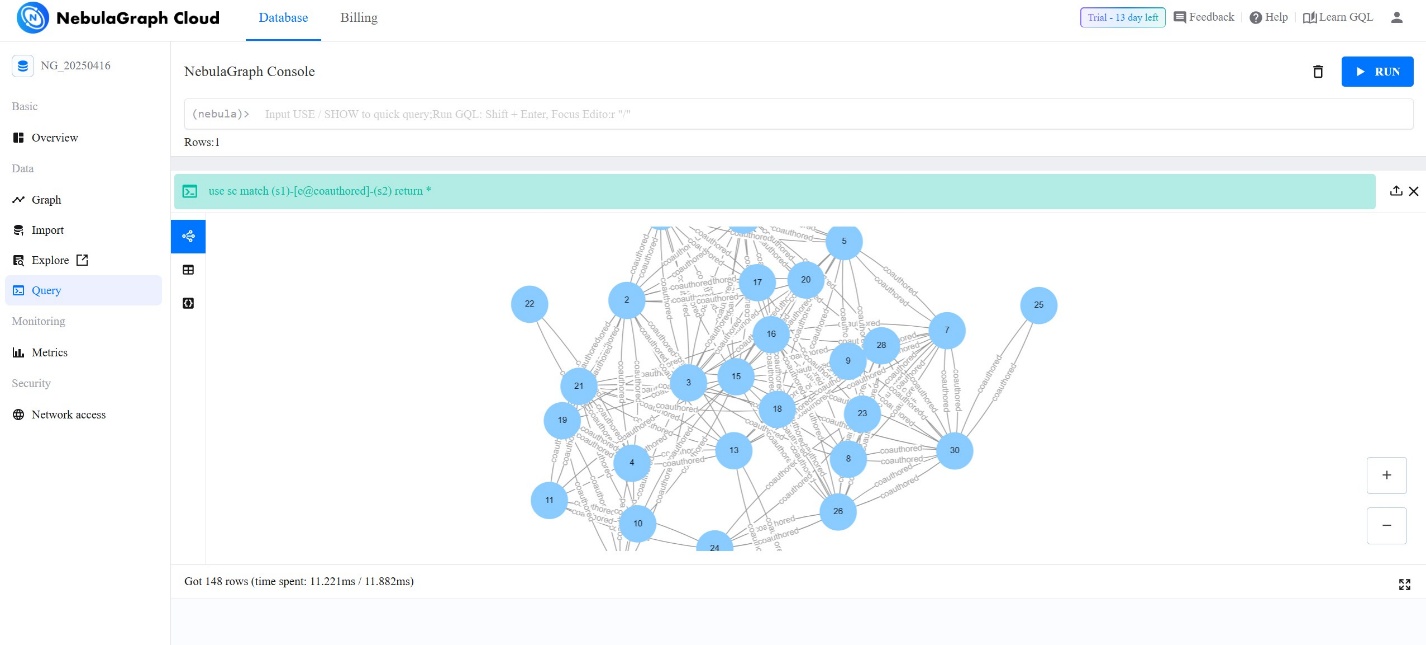


Рисунок . Визуализация графов

1. Исследование связей конкретного ученого

1. Найдите ученого с scientist\_id = 1 через Query Node.

Результат выполнения, представлен на рисунке 6.

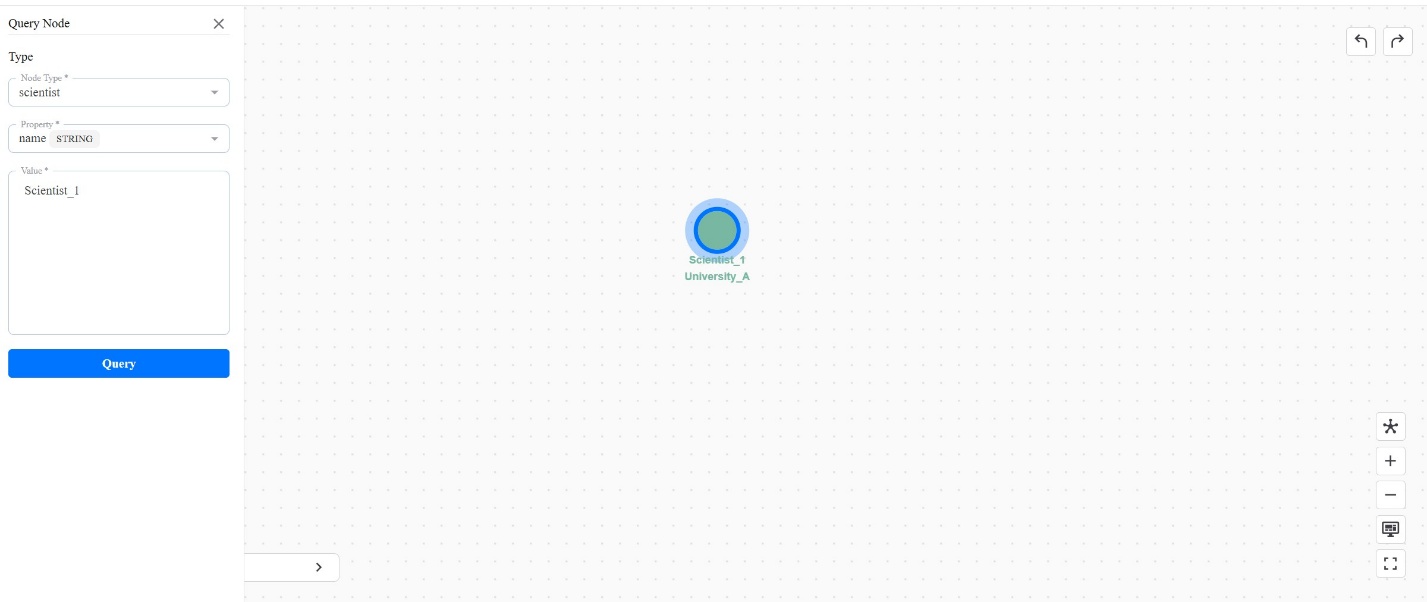


Рисунок 6. Поиск учёного с id 1

2. Используя Explore Relation:

a. Покажите всех его соавторов (ребро coauthored).

Результат выполнения, представлен на рисунке 7.

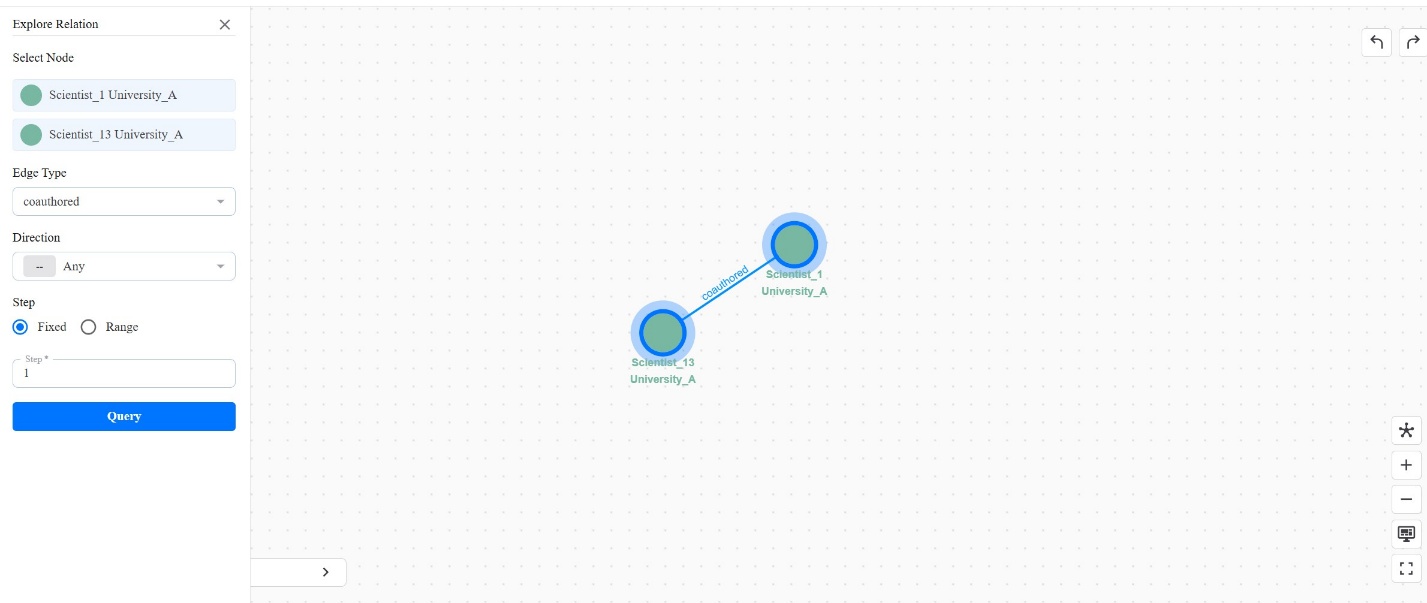


Рисунок . Все соавторы учёного с id 1

b. Определите, сколько совместных публикаций у него с каждым соавтором.

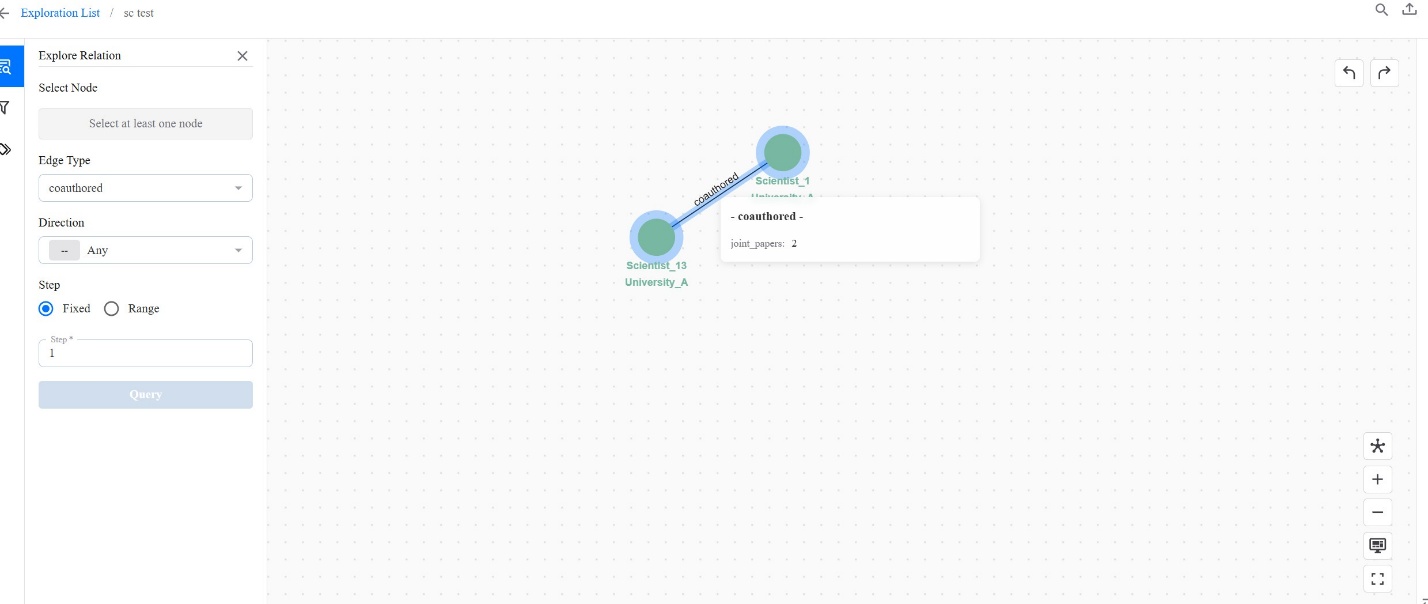
Результат выполнения, представлен на рисунке 8. 

Рисунок . Количество публикация с соавтором

3. Визуализируйте граф и ответьте:

a. С каким университетом чаще всего сотрудничает этот ученый? University\_A

2. Анализ публикаций ученого

1. Найдите публикации ученого scientist\_id = 5 через Query Node (тип scientist).

Результат выполнения, представлен на рисунке 9.

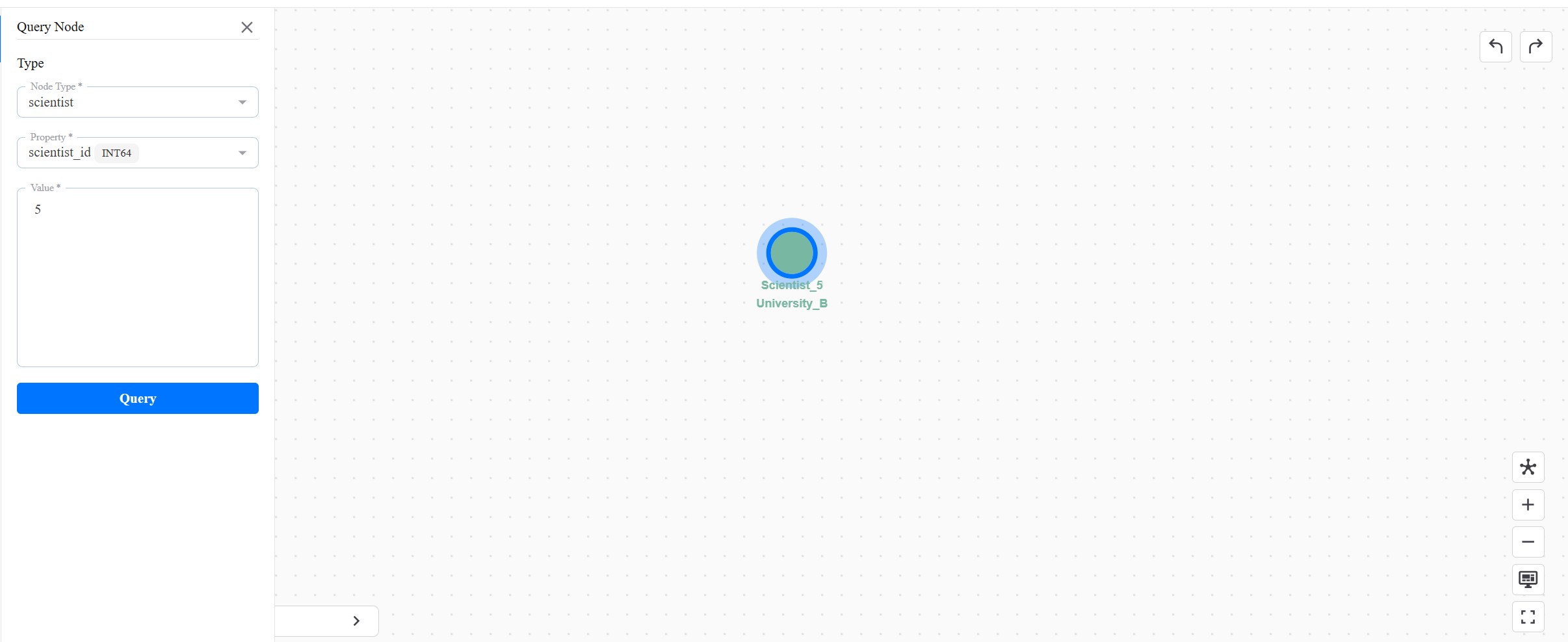


Рисунок 9. Поиск учёного с id 5

2. Используя Explore Relation:

a. Покажите связи wrote между ученым и его публикациями.

Результат выполнения, представлен на рисунке 10.

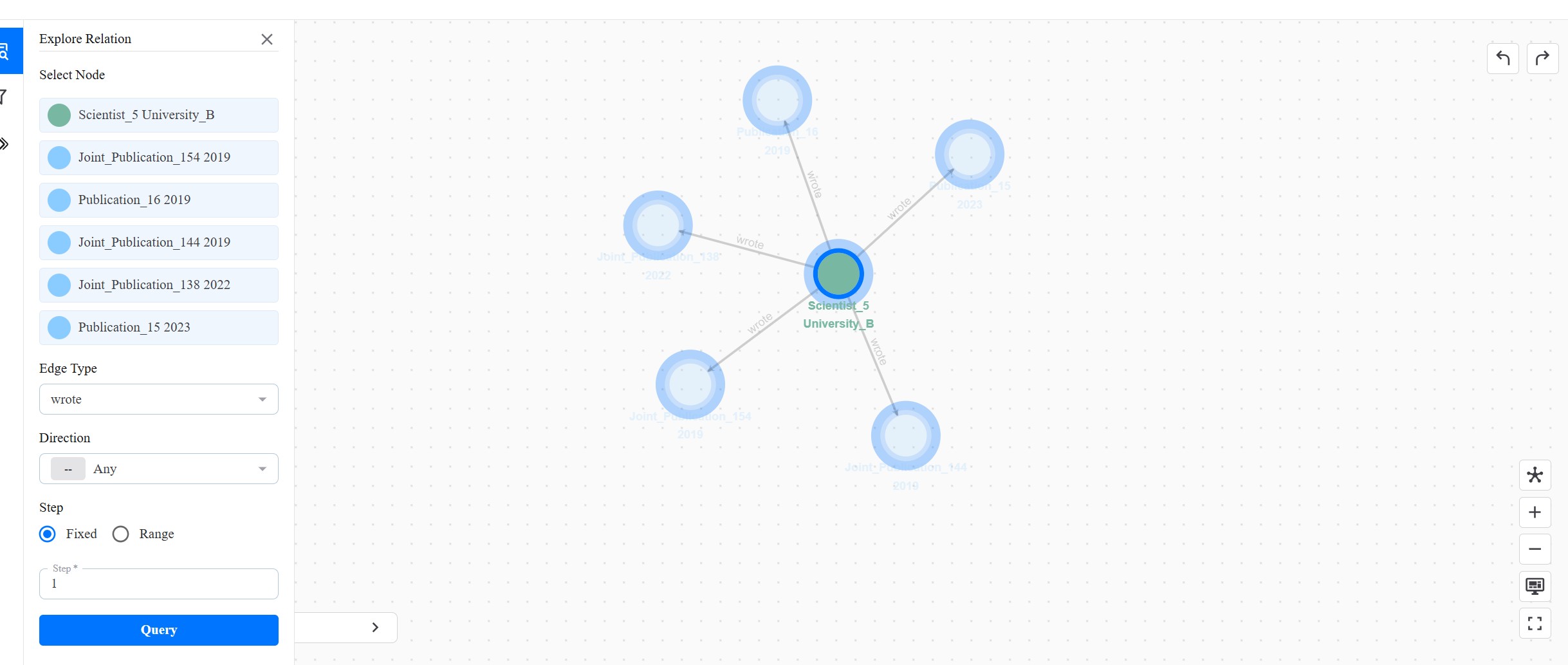


Рисунок . Связи wrote учёного с id 5

b. Отфильтруйте публикации за последние 3 года (самая молодая публикация в датасете – 2023 года).

Результат выполнения, представлен на рисунке 11.

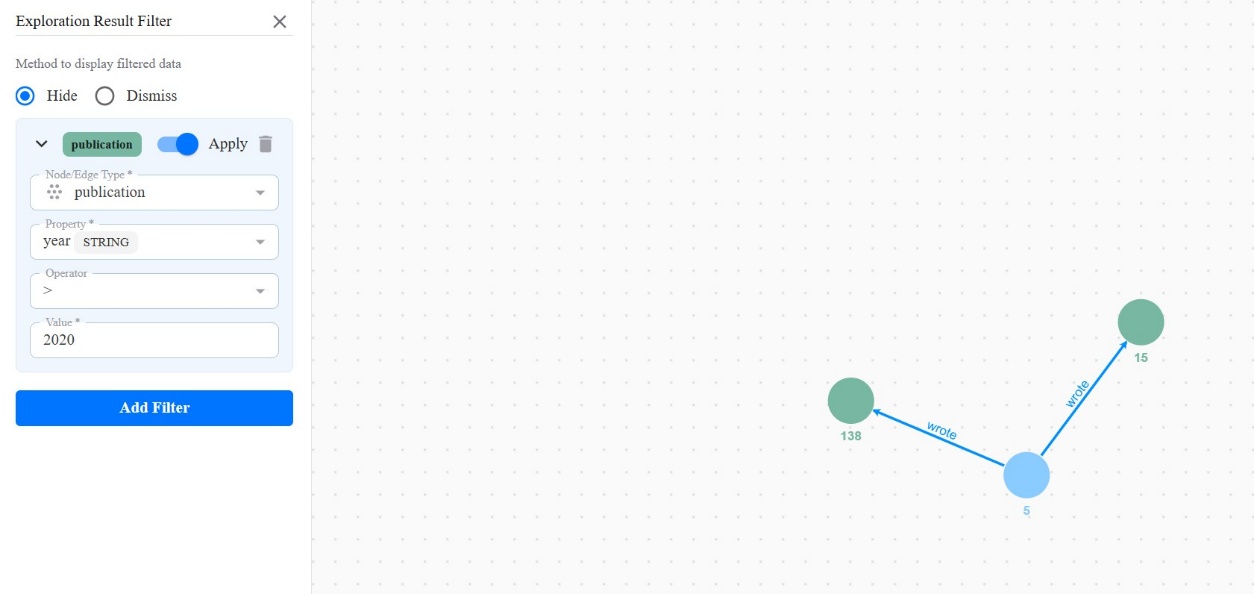


Рисунок . Все статьи учёного за последние 3 года

3. Определите:

a. Какой год был самым продуктивным для этого ученого? Самый продуктивный – 2019 год

3. Поиск пути между учеными

1. Найдите кратчайший путь между учеными scientist\_id = 1 и scientist\_id = 10 через соавторства:

a. Укажите тип ребра coauthored.

b. Ограничьте длину пути до 6 шагов.

Результат выполнения, представлен на рисунке 12.

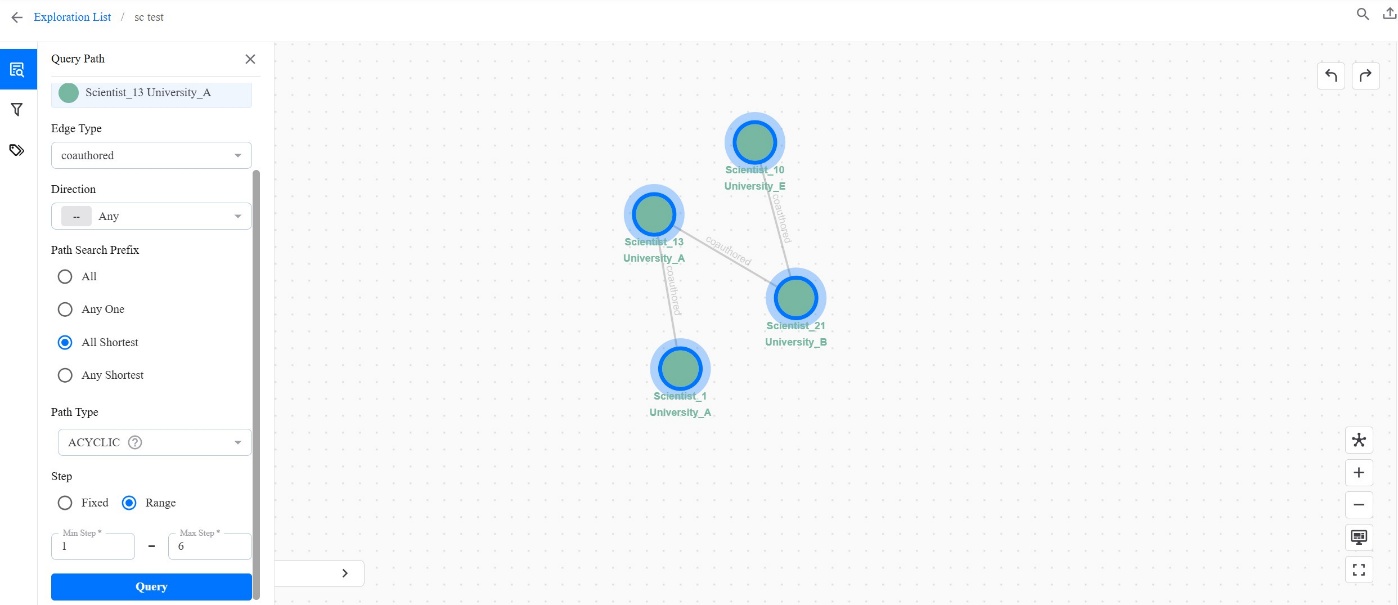


Рисунок . Кратчайший путь

2. Проанализируйте результат:

a. Сколько промежуточных ученых в пути? В пути 2 учёных.

b. Есть ли в цепочке ученые из одного университета? Таких учёных нет.

# Вывод

В ходе выполнения лабораторной работы были получены навыки взаимодействия с NebulaGraph в web-версии, изучен язык запросов и получены навыки взаимодействия с NebulaGraph Explorer.