ДИПЛОМНА РОБОТА
БАКАЛАВРА

ДРБ. ІП - 63.00.00.000 ПЗ

Група ІП-19-1К

Цапей Петро

2022

Міністерство освіти і науки України
Івано-Франківський національний технічний університет нафти і газу

Інститут інформаційних технологій

Кафедра інженерії програмного забезпечення

УДК 004.4

ДИПЛОМНА РОБОТА БАКАЛАВРА

Тема: “ Кластерні моделі в соціальних мережах та соціальному програмному забезпеченні ”

(назва згідно з наказом ректора)

спеціальність – 121 “Інженерія програмного забезпечення”

ПОЯСНЮВАЛЬНА ЗАПИСКА

ДРБ.ІП - 63.00.00.000 ПЗ

(позначення)

Консультанти:

з питань нормоконтролю

доц., к.т.н /В.Я. Піх/
(посада) (підпис) (дата)(розшифровка підпису)

Студент

ІП-19-1К_ /П.П.Цапей/
(шифр групи) (підпис) (дата) (розшифровка підпису)

Керівник роботи

к.т.н.,доц. /Р.Б.Вовк/
(посада) (підпис) (дата) (розшифровка підпису)

Допускається до захисту
Завідувач кафедри

д.т.н.,проф. /В.І.Шекета/
(посада) (підпис) (дата) (розшифровка підпису)

2022
(рік)

Івано-Франківський національний технічний університет нафти і газу

Інститут, факультет інформаційних технологій

Кафедра інженерія програмного забезпечення

Освітньо-кваліфікаційний рівень бакалавр

Спеціальність 121 – Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ:
Зав. кафедрою **ІПЗ**
проф. В.І. Шекета
“ ____ ” _____2022 р.

ЗАВДАННЯ

НА ДИПЛОМНУ РОБОТУ БАКАЛАВРА СТУДЕНТОВІ

Цапею Петру Петровичу

(прізвище, ім’я, по-батькові)

1. Тема проекту (роботи) “ Кластерні моделі в соціальних мережах та соціальному програмному забезпеченні”

керівник проекту (роботи) Вовк Роман Богданович, доцент, к.т.н.

затверджені наказом закладу вищої освіти від **“** ___ ” __травня___ 2022 р. № ___

2. Строк подання студентом проекту (роботи) 20 червня 2022 р.

3. Вихідні дані до проекту (роботи) Результати і матеріали отримані під час проходження переддипломної практики

4. Зміст розрахунково - пояснювальної записки (перелік питань, які потрібно розробити)

1. Огляд існуючих програмних рішень в предметній області процесів кластеризації
2. Алгоритмічні та математичні рішення для задач процесу кластеризації
3. Застосування кластерних моделей для вирішення CLUSTER HIRE в соціальних мережах
4. Проведення імітаційного моделювання
5. Виконання процесу тестування розробленого програмного забезпечення

5. Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень)

1. Формування команди (рис. 1.1)
2. Оренда кластера (рис. 1.2)
3. Приклад кластерного найму (рис. 1.3)
4. Графік соціальної мережі (рис. 1.4)
5. Робота SCAN (рис. 1.5)

6. Консультанти розділів проекту (роботи)

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання **25 травня 2022 р.**

Керівник _____
(підпис)

Завдання прийняв до виконання_____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту	Примітка
1	Огляд існуючих програмних рішень в предметній області процесів кластеризації	26.05.2022	виконано
2	Алгоритмічні та математичні рішення для задач процесу кластеризації	29.05.2022	виконано
3	Застосування кластерних моделей для вирішення CLUSTER HIRE в соціальних мережах	30.05.2022	виконано
4	Проведення імітаційного моделювання	04.06.2022	виконано
5	Тестування ефективності розробленого алгоритму та підходу	10.06.2022	виконано
6	Оформлення пояснювальної записки дипломної роботи завідувачем кафедри	20.06.2022	виконано

Студент – дипломник _____
(підпис)

Керівник роботи _____
(підпис)

АНОТАЦІЯ

Метою дипломної роботи є визначення спільної групи експертів, які зможуть ефективно працювати разом, щоб завершити низку проектів. В даний час вирішення проблеми в соціальних мережах досягається за допомогою жадібних евристичних алгоритмів і підходу цілочисельного лінійного програмування. Жадібні алгоритми часто дають швидкі результати, але вони роблять локально оптимальний вибір на кожному кроці і не дають глобальних оптимальних результатів. Недоліками підходу є те, що він вимагає значного обсягу пам'яті для створення змінних і обмежень, а також має дуже великий час обробки для великих мереж. Тоді як було доведено, що алгоритм зваженої структурної кластеризації для мереж дає швидші результати для задачі формування команди, що є окремим випадком проблеми найму кластера. Розширенням цієї проблеми є припущення, що серед експертів існує соціальна мережа, яка містить інформацію про їх минулу співпрацю. Якщо два експерта співпрацювали в минулому, то в майбутньому їм краще працювати в одній команді.

КЛЮЧОВІ СЛОВА: КЛАСТЕРНИЙ АНАЛІЗ, ПРОГРАМНИЙ ПРОЕКТ, ЕВРЕСТИЧНИЙ АЛГОРИТМ, ПРОЦЕС КЛАСТЕРИЗАЦІЇ, АНАЛІЗ ДАНИХ

ANNONANION

The aim of the thesis is to identify a collaborative group of experts who can work effectively together to complete a range of projects. Currently, solving the problem in social networks is achieved using greedy heuristic algorithms and an integer linear programming approach. Greedy algorithms often produce fast results, but they make locally optimal choices at each step and do not produce globally optimal results. Disadvantages of the approach are that it requires a large amount of memory to create variables and constraints, and has a very high processing time for large networks. Whereas the weighted structural clustering algorithm for networks has been proven to produce faster results for the team formation problem, which is a special case of the cluster recruitment problem. An extension of this problem is to assume that there is a social network among experts that contains information about their past collaborations. If two experts have cooperated in the past, it is better for them to work in the same team in the future.

KEYWORDS: CLUSTER ANALYSIS, SOFTWARE PROJECT, HEURSTIC ALGORITHM, CLUSTERIZATION PROCESS, DATA ANALYSIS

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ 6
ВСТУП 7
1 ОГЛЯД ІСНУЮЧИХ ПРОГРАМНИХ РІШЕНЬ В ПРЕДМЕТНІЙ ОБЛАСТІ ПРОЦЕСІВ КЛАСТЕРИЗАЦІЇ 11
1.1. Огляд дослідження проблеми формування команди за допомогою кластеризації 11
1.2. Основні поняття соціальних мереж і процесів кластеризації в них 18
1.3. Огляд літератури та методик по темі дипломної роботи 29
2 АЛГОРИТМІЧНІ ТА МАТЕМАТИЧНІ РІШЕННЯ ДЛЯ ЗАДАЧ ПРОЦЕСУ КЛАСТЕРИЗАЦІЇ 36
2.1. Постановка проблеми дослідження 36
2.2. Пропонована стратегія на основі модифікованого зваженого структурного алгоритму кластеризації для мереж 39
3 ЗАСТОСУВАННЯ КЛАСТЕРНИХ МОДЕЛЕЙ ТА АЛГОРИТМІВ ДЛЯ ВИРІШЕННЯ CLUSTER HIRE В СОЦІАЛЬНИХ МЕРЕЖАХ 50
3.1. Програмні засоби для проведення імітаційного моделювання та тестування 50
3.2. Результати виконаного аналізу даних 55
3.3. Порівняння пропонованого підходу з жадібним алгоритмом 71
3.4. Порівняння з підходом цілочисельного лінійного програмування 75
3.5. Результати використання алгоритму вирішення проблеми найму кластерів у соціальних мережах 76
ВИСНОВКИ 79
СПИСОК ПОСИЛАНЬ НА ДЖЕРЕЛА 80
БІБЛІОГРАФІЧНА ДОВІДКА

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ПО – предметна область

КА – кластерний аналіз

ТЗ – Технічне завдання

ПЗ - Програмне забезпечення

МА – мурашиний алгоритм

ВСТУП

Важливим аспектом будь-якої компанії/організації є набір співробітників, які можуть ефективно працювати разом. Компанії, які здійснюють кілька проектів одночасно, змушені наймати все більше і більше співробітників. В епоху соціальних мереж рекрутери зазвичай покладаються на доступні онлайн-соціальні мережі, такі як LinkedIn для набору співробітників. При прийомі на роботу рекрутери повинні переконатися, що сформована команда експертів буде досить економічно ефективною та ефективною, щоб завершити ці проекти у чутливий до часу спосіб.

Проект – це коли група експертів, кожен з яких має необхідні навички, працює разом у команді для досягнення спільної мети. Проект містить набір завдань або заходів, які виконує група експертів. Крім того, експерт — це той, хто має досвід, отриманий шляхом навчання та інструктажу з однієї або кількох навичок. Для виконання сукупності завдань, що вимагають набору різноманітних навичок, потрібна група експертів. Це досягається шляхом надання навичок різним експертам з додатковими знаннями. Це вимагає від великих організацій ефективно наймати групу експертів з різноманітним набором навичок для виконання сукупності завдань для завершення одного проекту, який називається проблемою формування команди.

Тоді на меті найняти групу експертів, які зможуть отримати всі навички, необхідні для набору проектів, а не для одного проекту. Кожен експерт вимагає грошових витрат (тобто зарплати) для участі в проектах, тому має бути встановлений бюджет для найму експертів. Працездатність експерта є ще одним важливим параметром під час найму експерта для кількох проектів. Кожен експерт має максимальну працездатність, яку він може запропонувати, а це означає, що може лише експерт працювати над певною кількістю проектів одночасно. Мета полягає в тому, щоб залучити значних спеціалістів, які достатньо ефективні для виконання ряду проектів у рамках заданого бюджету та працездатності, що є вигідним для організації та інституту. Наприклад,

великі організації зазвичай працюють над численними проектами. Для виконання цих проектів для отримання прибутку потрібен великий і різноманітний набір навичок. Тому вони наймають групу експертів, які можуть завершити ці проекти. Знаходячи групу експертів для ряду проектів, ми намагаємося знайти експертів, які співпрацювали в минулому. Таким чином, вони можуть добре працювати разом, а найнята група експертів співпрацює. Соціальні мережі можуть гарантувати, що сформована група експертів буде спільною. Соціальні мережі, такі як LinkedIn, можуть надати нам інформацію про експертів, які співпрацюють, що допомагає найняти спільну групу експертів . У нашому дослідженні ми зосереджуємось на проблемі у соціальних мережах. У наших експериментах соціальна мережа моделюється як графік, де експерти взаємопов'язані. Кожен вузол графіка вказує асоційованого експерта. Ребро з'єднує двох експертів, а вага краю представляє їх вартість зв'язку, отриману в результаті їхньої минулої співпраці. Чим більше минулого співробітництва між експертами, тим менше вартість комунікації. Таким чином, минула співпраця між експертами безпосередньо пов'язана з витратами на комунікацію сформованої групи експертів. Наприклад, якщо два експерти працювали разом над вісьмома проектами в минулому, а друга пара експертів працювала разом над одинадцятьма проектами в минулому. Тоді, маючи досвід спільного одинадцяти проектів, пара матиме менші витрати на спілкування між ними, ніж пара з вісьмома проектами. Якщо два експерта раніше не співпрацювали, вартість зв'язку вимірюється сумою ваги, що представляє найкоротший шлях між ними. Офіцер з найму також найматиме експертів, які співпрацювали в минулому, оскільки це нагадує їхню правильну робочу частоту та сумісність. Якщо група експертів може добре працювати разом, це допоможе вчасно завершити проекти. Компанія отримує прибуток, виконуючи проекти. Тому для завершення проектів ми обираємо групу експертів з мінімальними витратами на комунікацію.

Беручи до уваги вартість комунікації між двома експертами, потенціал експерта та виділений бюджет, ми знаходимо групу експертів з найменшими витратами на комунікацію для даного набору проектів. Усі особливості проблеми у соціальних мережах детально описані в останній частині дипломної роботи.

У технологічному світі, що розвивається, у всьому світі розвивається багато компаній, які працюють на феномені колективної експертизи. Колективну експертизу можна визначити як наявність певного рівня знань у групі людей або експертів, які володіють навичками, необхідними для виконання проектів. Компанії виграють від різноманітних проектів або можливостей, доступних на ринку, на який націлена компанія. Для кожного проекту їм потрібно найняти або виділити групу експертів. У зв'язку з цим ефективний розподіл ресурсів є критичним для компаній для досягнення конкретних цілей, таких як завершення проектів у встановлені терміни та отримання доходу. Він враховує минуле співробітництво між експертами під час найму, щоб остаточна група експертів була спільною та могла ефективно працювати один з одним, щоб дотримуватись термінів .

Онлайн-фрілансери, такі як Freelancer зазвичай працюють над кількома проектами. Ці компанії наймають фрілансерів з різними наборами навичок, а фрілансери працюють віддалено над різними проектами. Іншими словами, ці проекти мають необхідні навички, і вони повинні наймати експертів або фрілансерів для виконання цих проектів. Для кожної навичок проекту роботодавці наймають експерта, який може виконати вимоги до навичок. Також існує обмеження бюджету, який виділяється на найм експертів. Отже, загальна сума заробітної плати експертів має бути меншою за визначений бюджет.

Нарешті, є багато досліджень, що головною метою яких є вирішення проблеми наймання кластерів у соціальних мережах з використанням жадібних алгоритмів і цілочисельного лінійного програмування. Існують обмеження для

обох підходів. Жадібні алгоритми часто дають швидкі результати, але вони роблять локально оптимальний вибір на кожному кроці і не дають майже оптимальних або оптимальних результатів. Цілочисельне лінійне програмування не є ідеальним для великих соціальних мереж через свою обчислювальну складність, оскільки вимагає значної пам’яті для створення змінних і обмежень і має дуже великий час обробки для великих мереж.

1 ОГЛЯД ІСНУЮЧИХ ПРОГРАМНИХ РІШЕНЬ В ПРЕДМЕТНІЙ ОБЛАСТІ ПРОЦЕСІВ КЛАСТЕРИЗАЦІЇ

1.1 Огляд дослідження проблеми формування команди за допомогою кластеризації

Проблема формування команди (TFP)

Найм групи експертів, які володіють одним або кількома навичками, необхідними для завершення окремого проекту, відомий як формування команди. Це допомагає великим галузям і організаціям успішно завершити проекти вчасно. Це також допомагає експертам покращувати та відточувати свої навички та покращувати свої позиції в організації завдяки відмінним результатам. Концепція полягає в тому, щоб зібрати людей, які володіють навичками, необхідними для успішного завершення проекту. Для відбору людей для роботи в команді над проектом можна використовувати різні обмеження та фактори.

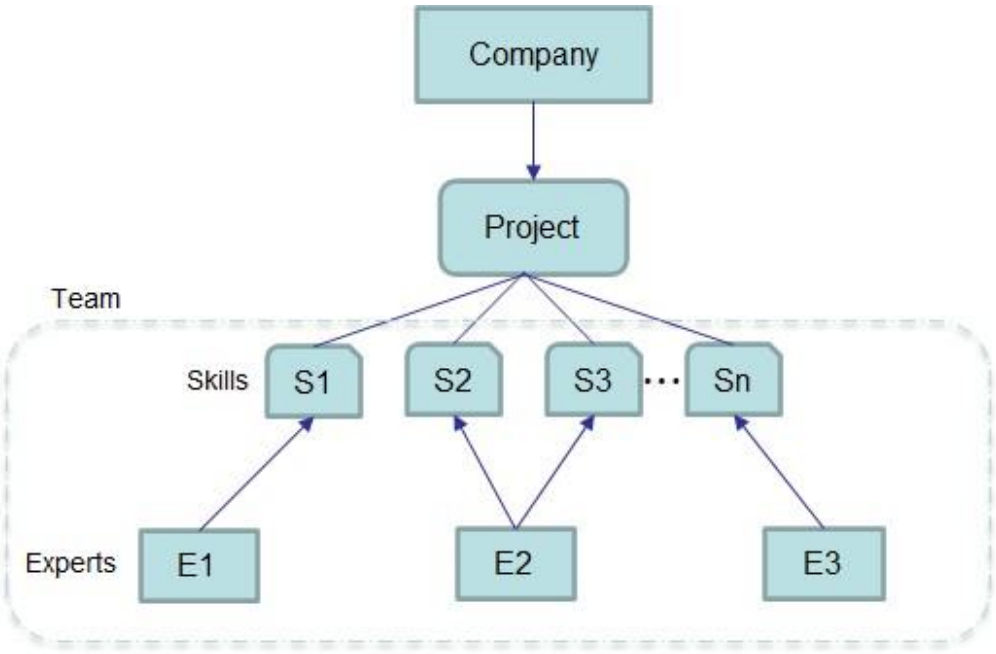


Рисунок 1.1 - Формування команди

Наприклад, подібний досвід членів команди є одним із факторів, які приносять користь команді, роблячи спілкування всередині команди більш ефективним. Це зменшує конфлікти та непорозуміння між членами команди. Таким чином, зменшується кількість часу, необхідного для створення активної робочої групи, оскільки менше потрібно для коригування стилів роботи

Як показано на рисунку 1.1, є один проект, який вимагає певних навичок, і ми наймаємо команду експертів, щоб виконати всі навички, необхідні для успішного завершення цього проекту.

Оренда кластера

Cluster Hire – це процес найму групи експертів (особистих осіб) для виконання вимог до набору проектів у межах бюджету організації, що наймає, та працездатності експертів. Наприклад, якщо компанії потрібно найняти групу експертів для десяти проектів, де кожен експерт володіє заздалегідь визначеним набором навичок, метою тут буде пошук експертів з цими навичками, необхідними для успішного завершення проектів у межах обмежень. Це називається проблемою Cluster Hire.

Існують різні атрибути та обмеження, які впливають на процес відбору експертів для певного набору проектів у задачі Cluster Hire, вони описані нижче:

1) Бюджет

Бюджет – це верхня межа грошової оцінки, яка призначається набору проектів фінансовим відділом компанії. Мета – найняти групу експертів у межах бюджету для виконання ряду проектів. Сума витрат на найм експертів для комплексу проектів не може перевищувати заздалегідь визначеного бюджету.

2) Експерти та їх атрибути

Навички експерта: кожен експерт або особа в соціальній мережі, що працює в компанії, має акредитацію за його/її досвід у певному наборі навичок.

Іншими словами, кожен експерт має набір навичок, які він/вона може запропонувати для проектів. Наша мета – підбирати експертів для конкретних проектів залежно від кваліфікації експертів, які б допомогли у виконанні проектів та одержанні прибутку для компанії.

Вартість найму експерта: вартість найму можна визначити як деяку грошову вигоду, пов'язану з експертом за участь у проектах. Кожен експерт має заздалегідь визначену вартість найму, пов'язану з ним. Вартість найму залишається незмінною незалежно від кількості проектів, призначених експертам, чи кількості навичок, які мають експерти. Необхідно враховувати, що сума витрат на найм відібраних експертів не повинна перевищувати бюджет, призначений для найму.

Працездатність експерта: працездатність означає максимальну кількість проектів, у яких експерт може брати участь у певний момент часу. У випадку, коли експерт має багато навичок, який може задовольнити вимоги до навичок більш ніж одного проекту, ми можемо призначити кілька проектів експерту в межах їхньої працездатності. Зверніть увагу, що експерт може охопити лише одну навичку в конкретному проекті; він не може охоплювати декілька навичок одного проекту. Це розумне припущення, оскільки організаціям може знадобитися наймати більше одного експерта для однакових навичок у проектах. Наприклад, якщо для проекту потрібні три експерти з Java, тоді нам потрібно найняти трьох унікальних експертів з Java для одного проекту, а не одного експерта з Java з працездатністю трьох. Але проектне завдання для експертів має бути в межах їхньої працездатності. Наприклад, якщо експерт має сім навичок, корисних для даного набору проектів, але експерт має працездатність лише для чотирьох проектів. Потім експерту буде виділено чотири різні проекти для чотирьох навичок.

Вартість комунікації експертів: у нашій установці ми маємо соціальну мережу, де експерти взаємопов'язані. Соціальна мережа представлена у вигляді графа, де вузли представляють експертів, а вузли з'єднані ребрами. Вага країв

відображає вартість зв'язку між експертами. Вартість комунікації між експертами залежить від кількості випадків, коли експерти вже мали досвід співпраці або роботи. Ми гарантуємо, що обрана група експертів має мінімальні витрати на комунікацію, що є важливим для хорошої співпраці. Група таких експертів зменшує накладні витрати часу на інтерпретацію досвіду кожного експерта тощо.

3) Проекти та їх атрибути

Навички проекту: кожен проект вимагає заздалегідь визначеного набору навичок, які необхідні для завершення проекту. Організації наймають людей, які володіють відповідним рівнем досвіду в навичках, необхідних для успішного завершення проекту. Ми повинні гарантувати, що ми наймаємо не менше, ніж кількість експертів, необхідних для виконання набору навичок, необхідних для проектів. Наприклад, для проекту потрібні три експерти з машинного навчання та два експерта з програмної інженерії. Наше завдання – знайти п’ять різних експертів для проекту для його успішного завершення.

Прибуток проекту: прибуток кожного проекту генерується після завершення проекту найнятою групою експертів. З кожним проектом пов’язаний грошовий прибуток (заздалегідь визначений дохід), який організація отримує після успішного завершення проекту.

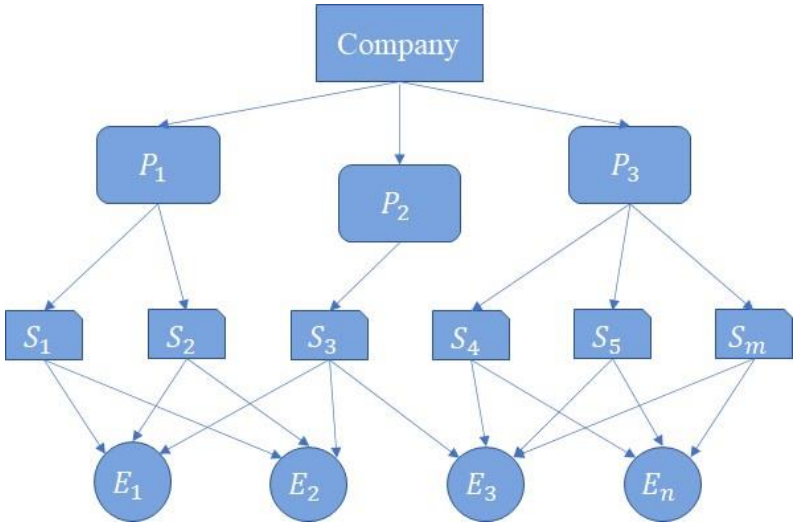


Рисунок 1.2 - Оренда кластера

Таблиця 1.1 - Інформація про експертів

Ідентифікатор експерта	Навички експерта	Вартість найму	Працездатність
$E1$	$S1, S2, S3$	2000	3
$E2$	$S1, S2, S3$	4000	5
$E3$	$S3, S4, S5, Sm$	8000	4
En	$S4, S5, Sm$	5000	3

Рисунок 1.2 ілюструє приклад проблеми Cluster Hire. Є компанія, яка працює над трьома проектами, а саме проект P1, проект P2 і проект P3. Кожен проект вимагає певного набору навичок для його завершення. Мета – найняти групу експертів для цих трьох проектів з обмеженим бюджетом та працездатністю. Бюджет, призначений для найму групи експертів для цих трьох проектів, становить 20 тис. Якщо ми візьмемо проект P1, він потребує по одному експерту для навичок S1 і S2, як показано на малюнку 2. Отже, на основі наявних експертів, показаних у таблиці 1.1, ми наймемо експертів E1 і E2 для завершення проекту P1. Ми можемо найняти E1 та E2, оскільки сума їхньої вартості найму становить 6000, що входить до бюджету. Зверніть увагу, що ми можемо найняти одного експерта для кількох проектів, враховуючи працездатність окремого експерта.

Тепер для проекту P2 нам потрібен лише один експерт для навичок S3, але у нас є три експерти, E1, E2 і E3, які можуть реалізувати цю навичку. Тому ми можемо виділити одного з уже найнятих експертів до проекту P2, що заощадить витрати на найм нового експерта. Так само ми наймаємо експертів для проекту P3. Коли ми застосовуємо проблему Cluster Hire до соціальних мереж, ми також повинні враховувати вартість комунікації експерта під час найму експертів. Отже, нашою метою буде знайти спільну групу експертів з мінімальними витратами на комунікацію, які зможуть виконати три проекти в рамках заданого бюджету та працездатності. Це було б ефективним вирішенням проблеми Cluster Hire у соцмережі.

Різниця між TFP та Cluster Hire

Дано набір n доступних експертів, кожен з яких має певні навички, а також набір m проектів організації, для яких потрібні люди з певними навичками. На малюнку 3 показано приклад проблеми Cluster Hire. У ньому пояснюються основні компоненти проблеми наймання кластерів і зображено можливе рішення проблеми.

У задачі Cluster Hire, коли групу експертів наймають для виконання ряду проектів, враховуються певні фактори. Ці фактори згадуються нижче:

- Необхідний набір навичок проектів
- Бюджет на завершення проектів
- Вартість зв'язку між найнятими експертами
- Навантаженість експертів
- Прибуток, отриманий після успішного завершення кожного проекту

Тоді як проблема формування команди стосується найму ефективної команди експертів для завершення одного проекту. Під час найму для окремого проекту головними факторами, які слід пам’ятати, є необхідний набір навичок проекту, бюджет проекту та вартість спілкування між експертами. Таким чином, під час найму потрібно враховувати набагато менше факторів, що робить проблему формування команди простою порівняно з проблемою найму в кластері.

На рисунку 1.3 ми маємо кілька проектів у правій частині; кожен з них має набір необхідних навичок. Щоб завершити ці проекти, для кожної необхідної навички ми повинні найняти експерта, який зможе реалізувати цю конкретну навичку. Наприклад, проект 1 вимагає двох експертів для навичок $S1$ і одного експерта відповідно для навичок $S2$ і $S3$. Як показано на малюнку 3, два експерта з навичками $S1$ розподіляються до Проекту 1. Один із двох експертів, найнятих для навичок $S1$, працює для Проекту 1 і Проекту 2 одночасно, що заощаджує витрати на найм для організації. Це є хорошим прикладом проблеми Cluster Hire.

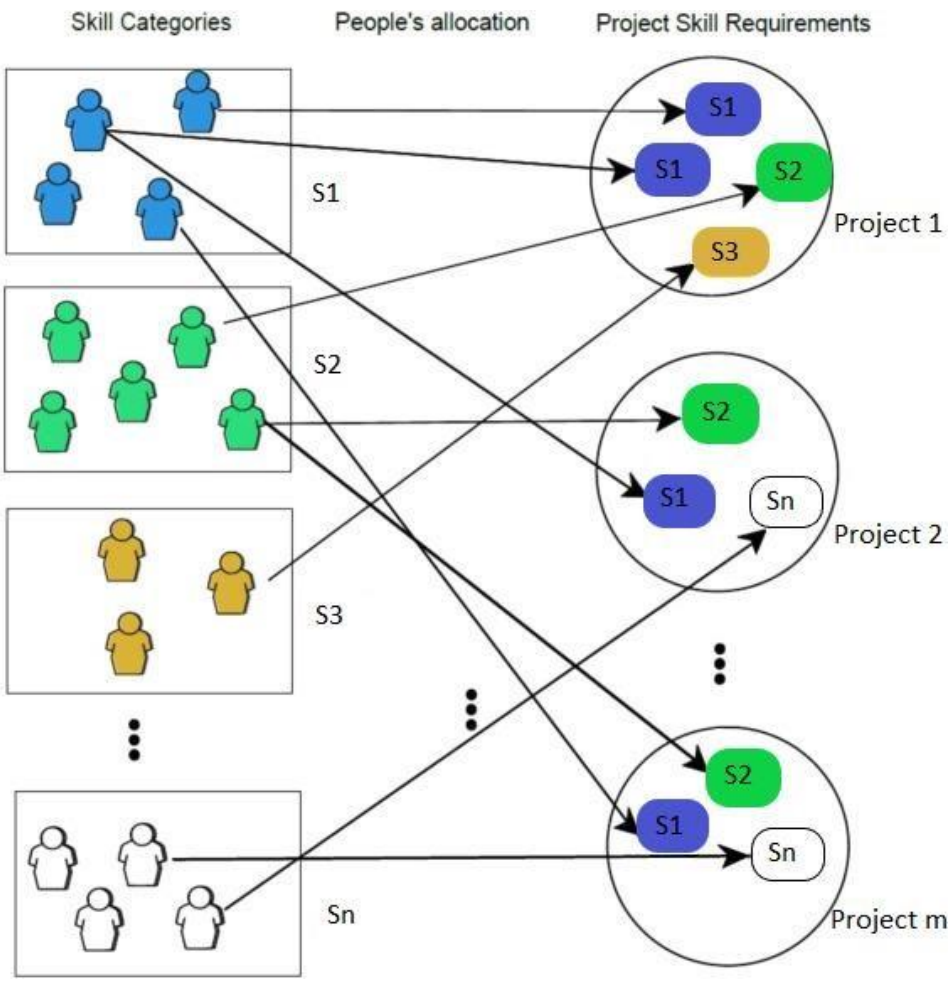


Рисунок 1.3 - Приклад кластерного найму

Однак, якщо ми виберемо лише один проект з малюнка 3 і найняти команду експертів для одного проекту, тоді це називається проблемою формування команди. Наприклад, для проекту 2 потрібен по одному експерту для навичок $S1$, $S2$ та Sn. Тому ми наймаємо команду експертів, не турбуючись про працездатність експертів чи прибуток проекту. Це приклад проблеми формування команди, коли ви наймаєте команду експертів для одного проекту.

Наша дипломна робота спрямована на вирішення проблеми Cluster Hire у соціальних мережах. Наймаючи групу експертів для набору проектів, ми також дбаємо про те, щоб група експертів була спільною, тобто вони мали менші витрати на зв’язок між ними. Інформація про вартість зв'язку зберігається в соціальній мережі, яка представлена у вигляді зваженого графіка. Вага межі

між двома експертами представляє значення вартості зв'язку. Детальніше про соціальні мережі ми поговоримо в наступному розділі.

1.2 Основні поняття соціальних мереж і процесів кластеризації в них

Соціальні мережі

В літературі описано соціальні мережі як найпопулярніший спосіб моделювання взаємодії людей у групі чи спільноті [1]. Соціальні мережі можна представити у вигляді графів, де вершина відповідає експерту/особі в якійсь групі, а ребро являє собою форму зв'язку/асоціації між відповідними двома експертами [1]. Інтереси, притаманні будь-якій спільноті, стимулюють асоціації в соціальних мережах. В [1] описуються соціальні мережі як набір вузлів, пов’язаних між собою набором відношень (ребрів) між ними, і ці соціальні мережі дотримуються складного зразка і утворюють складну систему вершин і зв’язків (ребрів) між ними. Соціальну мережу можна представити у вигляді зваженого або незваженого графіка [1]. Граф соціальної мережі можна представити у вигляді зваженого графа, який позначається G = (V, E, W), де V — множина вершин (акторів), E — множина ребер (відношень), W — множина вершин (акторів). вага ребер (відношень) [1].

Рисунок 1.4, показаний нижче, є прикладом графіка соціальної мережі, де кожен вузол представляє експерта, а ребра, що з’єднують два вузли, мають пов’язану з ними вагу. У цьому випадку вага краю представляє вартість комунікації між двома експертами, і кожен експерт має певні навички, які представлені на рисунку 1.4.

Більше того, автор пише, що соціальні мережі поширені всюди і можуть бути створені з різних дисциплін, таких як соціологія, дружба в Twitter, LinkedIn, протеїнова мережа тощо [1]. Соціальна мережа – це великий графік, на відміну від інших графіків. Незважаючи на подібний вигляд мереж, існують

деякі характерні особливості соціальної мережі, такі як ступінь поширення, кластеризований коефіцієнт і відстань шляху (6 ступенів поділу) [1].

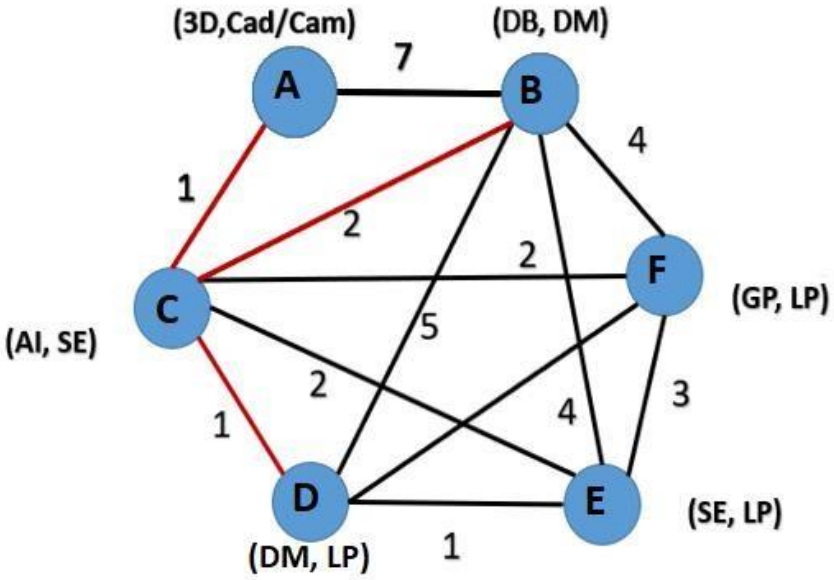


Рисунок 1.4 - Графік соціальної мережі

Для формування соціальних мереж було розроблено кілька реляційних і структурних підходів до соціального аналізу. Проте для демонстрації соціальних моделей, соціальної поведінки та соціальних почуттів були застосовані ідеї культури та культуротворення та класична соціологія. Всі перераховані вище фактори описували соціальні мережі в минулому. Крім того, існує інший підхід, який зосереджується лише на реальних моделях взаємодії та взаємозв'язку, за допомогою яких соціальні групи та спільнота взаємопов'язані. У деяких випадках його визначають як соціальну систему або соціальний організм. У 1929 році Фрідж Карінті був першою людиною в людстві, який ввів концепцію 6 ступенів поділу, що означає, що будь-які два випадково обрані люди в світі знаходяться на відстані 6 кроків один від одного. Пізніше, у 1960 році, Стенлі Мілграм експериментував, щоб знайти середню довжину шляху між цими людьми, і вона виявилася 5,9. Іншою помітною характеристикою

соціальної мережі є поширення ступенів. Визначається як ймовірність кількості з’єднань вузла з іншими вузлами по всій мережі. У 1999 році Albert-Lszl Barabsi та ін. сказав, що деякі вузли мають набагато більше з'єднань, ніж інші, названі концентратором, і вони використовували термін «безмасштабна мережа» [2].

Значення соціальної мережі для ефективного формування команди

Соціальна мережа, яка призначена для формування команди для виконання складних завдань, безпосередньо пов’язана з ефективною організацією команди. В [24] провели декілька експериментів і дійшли висновку, що такі соціальні структури є життєво важливими для виконання завдань та успішного завершення будь-якої проектної роботи. З кожним проектом і кожною командою пов’язаний підграф, де набір вершин представляє експертів, які мають колективні навички, необхідні для виконання поставлених завдань. Вони спільно проводять експерименти для вивчення впливу мережі на реальні дані, показуючи, як соціальні мережі серед членів команди можуть впливати на продуктивність та динаміку команди. В [25] вивчали динаміку формування груп, де розглядали три параметри, а саме членство, зростання та еволюцію, що допомогло їм проаналізувати розвиток великих груп в організації/спільноті. Вони дійшли висновку, що на бажання та бажання особи/експерта приєднатися до певної спільноти/організації впливають численні фактори.

Крім того, вони провели дослідження цих факторів, щоб проаналізувати їх масштабованість. Попереднє дослідження показало кореляцію між бажанням людини приєднатися до спільноти та кількістю людей, з якими вона дружить у цій організації/спільноті. Існує ряд методів, які використовуються для визначення найбільш помітних детермінант, що впливають на таку соціальну групу; ці методи відомі як методи дерева рішень.

Далі розроблено нову методологію, яка вимірює вхід і вихід людей з однієї спільноти в іншу на основі їхнього рівня інтересу до певної/конкретної теми.

Набір даних DBLP використовується для вилучення результатів з експериментів і методологій.

Аналіз соціальних мереж (SNA)

Аналіз соціальних мереж (SNA) можна просто визначити як поглиблений аналіз структури соціальних мереж, тенденції до часу, моделі взаємин із соціальними акторами та наявних даних разом з ними [1]. Оскільки соціальні мережі формуються здебільшого з нашою екологічною структурою, дослідження її первинних показників, таких як центральність близькості, центральність між посередністю, центральність ступеня, діаметр тощо, дасть більш надійні результати, що було б інноваційною зміною у світі [1]. Щоб проаналізувати соціальну мережу, нам потрібно перетворити її на граф з вузлами та ребрами, де вузли є соціальними акторами (можуть бути людиною, організацією чи будь-яким іншим), а ребра — це відносини між ними [1]. Графік може бути як зваженим, так і незваженим (вага в основному визначається на основі подібності двох вузлів, відстані між ними або частоти співпраці) [1]. При цьому вона може бути як спрямованою, так і ненаправленою [1]. Тому аналіз соціальних мереж використовує концепції теорії графів.

Ще однією унікальною характеристикою соціальної мережі є її динамічна поведінка. Однак реальні соціальні мережі нестійкі й не статичні, як-от LinkedIn, Facebook, Twitter, Instagram, Snapchat тощо. Вони швидко розширюються та швидко розвиваються з часом. Ці непередбачувані зміни роблять соціальні мережі складними і дуже важкими для розуміння, а ще більш складними для вирішення.

У сучасному світі аналіз соціальних мереж застосовується до різних бізнесів і дисциплін, таких як наука, політика, охорона здоров’я та повсякденне життя [1]. Здебільшого він використовується для підвищення ефективності та ефективності процесів прийняття рішень [1]. Такі програми використовуються в правоохоронних органах, громадських організаціях, охороні здоров’я та

соціальних мережах, таких як Facebook, Instagram, Snapchat і LinkedIn. Деякі програми аналізу соціальних мереж допомагають аналізувати поширення захворювань [3], виявляти терористичні дії [4], спостерігати за динамічними мережами співавторства [5] та багато інших дослідницьких застосувань у реальному світі.

Різні проблеми SNA

Згідно [1], аналіз соціальних мереж займається різними проблемами. Вони такі, як зазначено нижче:

- Проблема формування команди
- Передбачення посилань
- Виявлення спільноти
- Спільна рекомендація
- Виявлення лідерства
- Міграція між громадами
- Сентиментальний аналіз
- Страховий аналіз
- Виявлення шахрайства

Тут ми збираємося обговорити деякі з перерахованих вище проблем. Проблема формування команди (TFP) відноситься до найму групи експертів для виконання призначеного проекту, який вимагає виконання набору навичок, де набір необхідних навичок є підмножиною набору загальної кількості навичок. Прогноз зв’язку передбачає відсутність зв’язків у поточних мережах і нові зв’язки або зв’язки з розпадом у майбутніх мережах [7]. Виявлення спільноти можна визначити як пошук людей з подібними смаками, вибором і перевагами для зв’язування з соціальною мережею, що призводить до утворення віртуальних кластерів або спільнот [8]. Спільна рекомендація визначає користувачів, чиї смаки подібні до смаків даного користувача, і рекомендує товари, які їм сподобалися [9].

У нашому дослідженні ми прагнемо вирішити проблему Cluster Hire (який є розширенням TFP) у соціальних мережах шляхом мінімізації витрат на комунікацію. Найняти групу експертів із певним набором навичок для виконання ряду проектів є складним завданням. Знайти групу експертів у реальному житті з сотень людей є дуже витратним і нездійсненним завданням. Тому ми прагнемо вирішити проблему найму кластерів у соціальних мережах, у складі якої є багато експертів. Ми реалізуємо наше рішення за допомогою модифікованого зваженого структурного алгоритму кластеризації для мереж (MWSCAN), який є алгоритмом кластеризації. У наступному розділі ми обговоримо деякі концепції та підходи кластеризації.

Кластеризація

Групування подібних елементів у групу визначається як кластеризація. За допомогою кластеризації можна сформувати багато груп на основі їхніх характеристик та міркувань. Кластеризація графів — це особливий тип кластеризації, який використовується на великих графах або мережах. Крім того, кластеризація мережі (розбиття графів) є важливим завданням для виявлення базових структур у мережах і допомагає при аналізі соціальних мереж [10].

У кластеризації існують різні категорії. Вони такі, як зазначено нижче:

• Ієрархічна кластеризація

• Кластеризація на основі щільності

• Кластеризація на основі розділів

• Кластеризація на основі сітки

Кластеризація на основі щільності

Кластеризація на основі щільності покладається на вузли на графіку для групування. Кластер утворений вершинами, які сильно пов’язані між собою. Деякі приклади: DBSCAN (Просторова кластеризація додатків із шумом на основі щільності), AHSCAN (Агломеративний ієрархічний структурний

алгоритм кластеризації для мереж), DHSCAN (Алгоритм розділеного ієрархічного структурного кластерування для мереж), SCAN (алгоритм структурного кластерування мереж для мереж). Алгоритм DBSCAN був запропонований для просторової кластеризації даних із шумом. Завдяки своїм унікальним характеристикам цей алгоритм швидко став відомим у різних сферах. Застосування цього алгоритму включає в себе такі галузі науки, як групування просторової мережі цивільної інфраструктури, хімія, спектроскопія, медична діагностика на основі медичних зображень та соціальні науки (дані феромонів) [11]. DBSCAN також використовується для відокремлення тривимірних зображень у дистанційному зондуванні.

Недоліком DBSCAN є те, що він ненадійний, коли обмежувальні елементи двох кластерів розташовані відносно занадто близько. Щоб протистояти цьому, Xiaowei Xu et al. [10] запропонував алгоритм структурної кластеризації мереж (SCAN). Цей алгоритм був здатний створювати щільні кластери, а також ефективно обробляти слабко зв’язані вузли з хабами. Ці хаби були описані як вузли, які одночасно пов’язані між собою з двома або більше кластерами.

Незважена кластеризація графів за допомогою SCAN

В [10] запропоновано новий метод кластерування мережевого графа на основі структурної подібності. Це було використано для групування неорієнтованого графа, а також незваженого графа на основі структурної подібності. Тому автор назвав його Structural Clustering Algorithm for Networks (SCAN). Він виявляє кластери, концентратори та виброси, використовуючи структуру та зв’язність вершин як критерії кластеризації [10]. Цей алгоритм знаходить основний вузол поза мережею. Вузол Core вибирається на основі кількості сусідів у його околиці, які структурно подібні. Пізніше, розглядаючи цей вузол Core як початковий кластер, він створює кластер навколо нього. Цей підхід ділить графік на три частини: кластери, концентратори та викиди, які проілюстровані на наступному малюнку 5. Концентратори — це, по суті, вузли,

які об’єднують багато кластерів — це вузли, які незначно пов’язані з кластером зі слабким зв’язком. На малюнку є один концентратор, який з'єднує два кластери і працює як міст між двома кластерами. Крім того, у нас є один викид, який слабо пов’язаний з одним із кластерів.

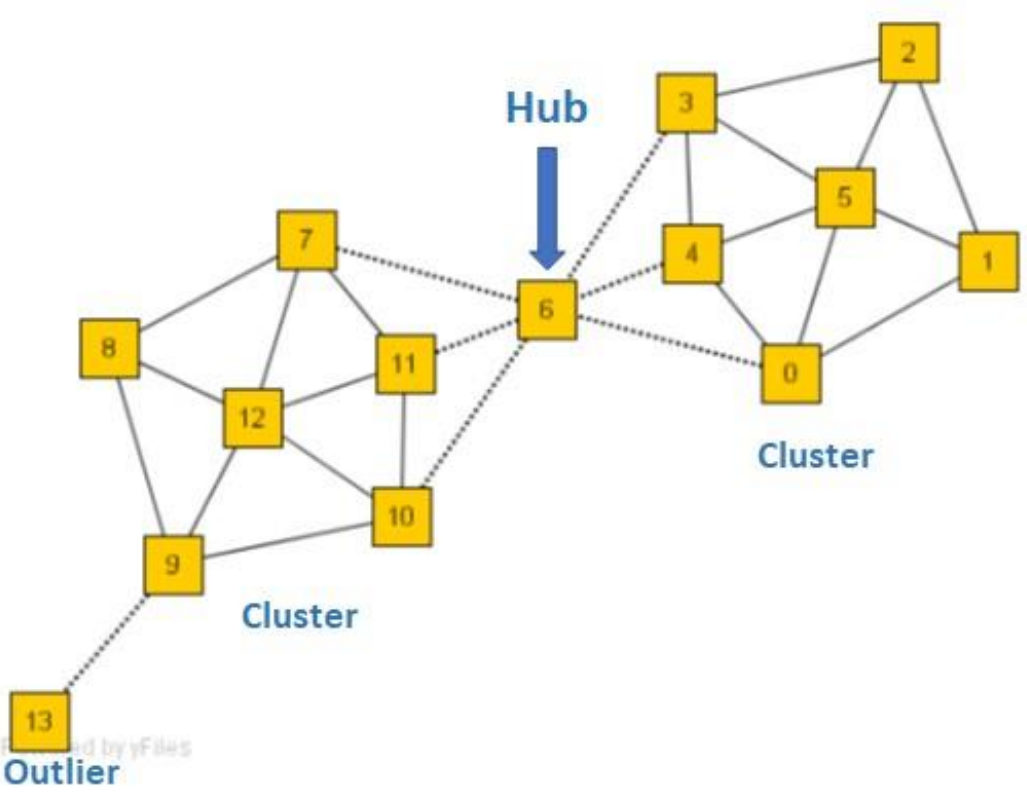


Рисунок 1.5 - Робота SCAN [10].

У цій дослідницькій роботі [10] вони зосередилися на простому незваженому та неорієнтованому графі. Нехай $G(V, E)$ — граф, де V — множина вершин або вузлів, а E — набір ребер, що сполучають дві вершини. Структура вершини визначається її сусідством. Формальні визначення з [10] описані нижче.

Визначення 1. (Вершина структура)

Нехай v, w V , структура v визначається його околицями. Вершинна структура вершини v позначається $\tau(v)$ [10] τ

Визначення 2. (структурна подібність)

Структурна подібність між двома вершинами v і w буде великою, якщо вони мають подібну структуру сусідів, що є частим режимом спільної роботи, і їх структурна подібність позначається $\sigma(v, w)$ [10]

Вони нормалізують кількість спільних сусідів двох вершин на середнє геометричне розміру двох околиць. Тому що якщо ми використовуємо лише кількість спільних сусідів, то вершина 6 на малюнку 5 буде об'єднана в один із кластерів або призведе до злиття двох кластерів [10].

Визначення 3. (ε - Околиці)

Для вершин v і w ми застосовуємо поріг ε до структурної подібності $\sigma(v, w)$, коли призначаємо кластерну належність вершині v [10]. Це називається ε – околиця для вершини v і позначається $N\varepsilon(v)$:

$$N\varepsilon(v) = \{w \in \tau(v) \mid \sigma(v, w) \geq \varepsilon\}$$

Визначення 4. (ядро)

Вершина $v \in V$ називається ядром з посиланням на ε і μ , якщо її околиця містить принаймні μ вершин, де μ — кількість експертів околиць, підключених до основної вершини. Отже, основним експертом є експерт із високою зв'язком [10]. Основна вершина позначається $Core\varepsilon,\mu(v)$, де виконується наступна умова,

$$|N\varepsilon(v)| \geq \mu.$$

Визначення 5. (Пряма доступність структури)

Ми вирощуємо кластери з основної вершини наступним чином. Якщо вершина знаходиться в ε – околиці ядра, вона також повинна бути в тому ж кластері. Вони мають схожу структуру і пов'язані між собою. Ця ідея представлена у наведеній нижче формулі [10]

$$DirREACH\varepsilon,\mu(v, w) = Core\varepsilon,\mu(v) \wedge w) \wedge w$$

Псевдокод алгоритму SCAN [10]

АЛГОРИТМ СКАНУВАННЯ ($G(V, E)$, ε, μ)

// всі вершини в V позначені як некласифіковані; для кожної некласифікованої вершини $v \in V$ робити

// КРОК 1. перевірити, чи є v ядром;

якщо $Core\varepsilon,\mu(v)$, то

// КРОК 2.1. якщо v є ядром, новий кластер розширюється; генерувати новий ClusterID; вставити всі $x \in N\varepsilon(v)$ до черги Q; а $Q \neq 0$ робити y = перша вершина в Q;

$R = \{x \in V \mid DirREACH\varepsilon,\mu(y, x)\}$; для кожного $x \in R$ зробіть, якщо x некласифікований або не є членом, тоді призначте поточний ідентифікатор кластера x; якщо x некласифікований, то вставте x до черги Q; видалити y з Q; інше

// КРОК 2.2. якщо v не є ядром, він позначається як нечлен v як не-член; кінець для.

// КРОК 3. далі класифікує нечлени для кожної нечленної вершини v do

if ($\exists$ x, $y \in \Gamma(v)$ (x.clusterID $\neq$ y.clusterID), то позначте v як концентратор else мітку v як викид; закінчення для закінчення SCAN.

Алгоритм SCAN починається з відвідування кожної вершини один раз, щоб знайти структурно-зв'язані кластери, а потім ідентифікувати концентратори або викиди, які відвідують ізольовані вершини [10]. Для заданих налаштувань параметрів алгоритм SCAN виконує один прохід мережі та знаходить усі підключені кластери. На початку всі вершини позначаються як некласифіковані і заносяться в чергу. Алгоритм SCAN класифікує кожну вершину, як члена кластера, так і не-члена. Для кожної некласифікованої вершини, SCAN перевіряє, чи є ця вершина ядром. Якщо вершина є ядром, новий кластер розширюється з цієї вершини. В іншому випадку вершина позначається як нечленна [10]. Для формування нового кластера SCAN починається з основної вершини і знаходить усі структурно доступні вершини з

основної вершини. SCAN генерує новий ідентифікатор кластера, який буде призначено кожному члену кластера. Щоб знайти членів кластера, він додає всі некласифіковані вершини сусідства основної вершини в чергу. Для кожної вершини в черзі він знаходить усі безпосередньо доступні вершини, які є некласифікованими, і вставляє ці вершини в чергу. Ці кроки повторюються, поки черга не стане порожньою [10]. Решта вершини, які не є членами, можна позначити як концентратори або викиди. Якщо ізольована вершина має два або більше ребер з іншими вершинами, то її можна класифікувати як концентратор. Тоді як інші вершини будуть класифікуватися як викиди [10].

Кластеризація зважених графів за допомогою WSCAN

Зважені графи мають певне число, пов'язане з ребрами, яке зазвичай називають вагою ребра [3]. Метод кластеризації, розглянутий вище в розділі 2., націлений на незважені графіки. Однак, якщо надано зважений графік, алгоритм SCAN просто ігноруватиме ваги ребер і виконуватиме кластеризацію на основі структурних властивостей графа. У деяких випадках це може бути прийнятним, але іноді зовсім неприпустимо [4]. Щоб подолати цю проблему, автор [1] та [3] запропонував новий алгоритм під назвою Weighted Structural Clustering Algorithm for Networks (WSCAN) як рішення для виконання кластеризації у зважених графах на основі структурної подібності. Алгоритм представив дещо модифіковану формулу для структурної подібності, яка визначається нижче.

Визначення 6. (Розширена структурна подібність)

Нехай $G(V, E)$ — граф, де V — множина вершин або вузлів, а E — набір ребер, що сполучають дві вершини. Нехай $v, w \in V$ і функція нижче показують структурну подібність двох вершин у графі, де $w(v, w)$ представляє вагу ребра між експертами/вершинами v і w.

Структурна подібність позначається $\sigma(v, w)|\tau(v) \cap \tau(w)|$

$$\sigma(v,w) = \frac{}{\sqrt{|(\tau(v)||\tau(w))|}\ w(v,w)}$$

У наведеному вище рівнянні розширена структурна подібність двох вершин буде великою, якщо вони мають подібну структуру сусідів з низькою вартістю зв’язку між ними, що означає, що вони часто працюють разом. При розрахунку структурної подібності між ними враховується вага краю. Алгоритм WSCAN використовує всі інші формули, які використовуються в алгоритмі SCAN, за винятком розширеної формули структурної подібності. Алгоритм WSCAN виконує ті самі кроки, які описані в псевдокоді SCAN в розділі з модифікованою формулою для структурної подібності.

1.3 Огляд літератури та методик по темі дипломної роботи

Функції вартості комунікації команди в соціальних мережах

Проблема формування команди була вперше представлена спільноті інтелекту та управління даними Лаппас та ін. [6]. Для оцінки вартості комунікацій команди було запропоновано дві функції. Перша функція вимірювала вартість комунікації команди за діаметром підграфа, створеного найнятою командою; діаметр вимірюється найбільшим найкоротшим шляхом між будь-якими двома вузлами підграфа. Друга функція розрахувала вартість зв'язку команди за вартістю мінімального поясного дерева (MST) на підграфі. Однак обидва підходи мали недоліки, оскільки перший підхід вимірює лише вартість комунікації між двома експертами, які найбільше віддалені один від одного в команді. Тоді як другий підхід також не вимірює вартість комунікації всіх експертів, які працюють у команді.

Тому, щоб протистояти цим недолікам, Kargar et al. [2] випустили дві нові функції вартості комунікації команди на основі проведених ними експериментів.

Ці дві функції описані нижче:

• Перша функція називається функцією sumofdistance, яка обчислює вартість комунікації команди на основі підсумовування найкоротшої відстані між усіма експертами, які володіють необхідними навичками для даного проекту, або іншими словами, які є частиною команди.

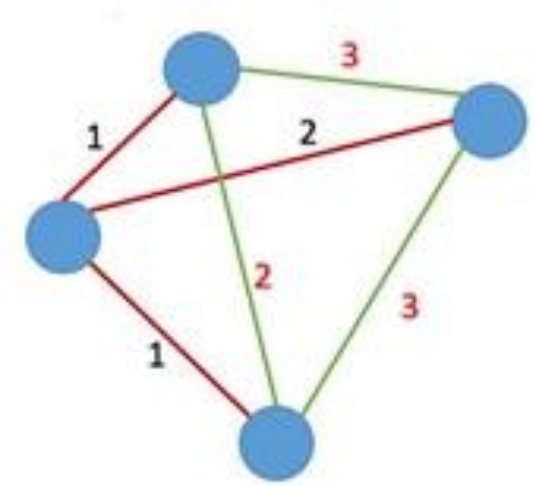


Рисунок 1.6 - Зважений графік команди

На рисунку 1.6, якщо ми обчислимо вартість комунікації команди, то це буде сума всієї ваги/відстані між усіма вершинами.

Вартість спілкування в команді = 1 + 1 + 2 + 2 + 3 + 3 = 12

• Друга функція зосереджена на розрахунку вартості комунікації в команді, якщо в команді є лідер. У цій установці команда експертів має призначеного керівника, який відповідає за координацію та моніторинг проекту. Кожен експерт у команді має спілкуватися з лідером; тому вартість комунікації лідера є важливою при розрахунку вартості комунікації команди. Ця функція враховує комунікаційні витрати лідера разом з усіма іншими експертами команди. Вона відома як функція відстані лідера, яка визначає суму найкоротшого шляху між кожним експертом і лідером у проекті для обчислення вартості комунікації команди.

Ми використовували функцію sumofdistance для розрахунку вартості зв'язку між вибраною групою експертів, оскільки в нашій установці немає лідера, а sumofdistance враховує всіх експертів у найнятій групі експертів.

Проблема формування команди онлайн у соціальних мережах

В [5] провели дослідження та експериментували з онлайн-версією проблеми формування команди. Вони використовували соціальну мережу експертів, яка володіє набором навичок для виконання кількох проектів і має сумісність і попередню співпрацю один з одним, отримані з їх попереднього досвіду роботи. Вони розглянули послідовність завдань, які надходять в режимі онлайн, кожне завдання вимагає певного набору навичок. Основна ідея статті полягає у формуванні кількох команд для кількох проектів; кожен проект і пов'язаний з ним набір завдань з'являються в режимі онлайн. Метою було сформувати команду, маючи на увазі три атрибути, які обговорюються нижче:

• Охоплюються навички, необхідні для завершення проекту

• Сформована команда є спільною

• Працездатність особистості не перевантажена

У минулому для розв'язання задачі TFP використовувалися різні евристичні та наближені алгоритми з зазначеними вище цілями з подальшим аналізом результатів та висновками.

Методика кластеризації TFP у соцмережі

Проблема формування команди (TFP) у соціальній мережі популярна і широко досліджується багатьма дослідниками та авторами. Було запропоновано багато методів, і було проведено безліч експериментів для вирішення TFP, щоб виявити та ідентифікувати експертів, які могли б виконати всі завдання, необхідні для проекту. Вони також пам'ятали, що сформована команда найбільш спільна, щоб скоротити час, необхідний для виконання цього проекту. З цією метою в роботі [7] запропонували техніку кластеризації, яка

групує набір експертів у вигляді кластера, де точки кластера представляють експертів з необхідними навичками для проекту. Вони прийняли зважений алгоритм SCAN (WSCAN), який є розширеною/розширеною версією SCAN, тобто алгоритм структурної кластеризації для мереж, для вирішення проблеми формування команди (TFP) з мінімальними витратами на зв'язок. Вартість комунікації розраховується на основі минулої співпраці та досвіду роботи (раніше працював у кількох проектах) між двома найнятими експертами. Їхня техніка ґрунтується на основній ідеї ідентифікації кластерів, концентраторів та викидів у мережі/графі експертів, де кожен вузол представляє експерта, а ваги краю представляють вартість зв'язку. По-перше, для вирішення проблеми розглядався пул експертів, які володіють навичками, необхідними для виконання конкретного проекту. Далі вони шукали стрижня, тобто експерта з високими зв'язками серед усіх експертів. Потім кластер розширюється від ядра до вузлів сусідства, тобто від вузлів щільно зв'язаних до вузлів слабо зв'язаних. Розширення кластера здійснюється до досягнення порогового діапазону вартості зв'язку. Порівняння результатів зваженого сканування для TFP виконується за допомогою жадібного, генетичного, випадкового та точного алгоритму, і WSCAN дає швидші результати порівняно з усіма іншими алгоритмами.

Оренда кластера в соціальній мережі за допомогою Greedy Algorithm

Проблему Cluster Hire у соціальних мережах вперше досліджували Meet та інші. Їхнє дослідження було двооб'єктивним, коли вони наймають конкретну групу людей, які можуть отримати максимальний прибуток і мати мінімальні витрати на спілкування на основі їхньої минулої співпраці та досвіду роботи. Зокрема, вони мали на меті вирішити проблему Cluster Hire у соціальній мережі за допомогою жадібного алгоритму. Їхня робота була заснована на припущенні, що існує соціальна мережа серед експертів, де кожен експерт пов'язаний з іншим експертом через попередню співпрацю в одному або кількох проектах.

Крім того, кожен експерт у сформованій установці пов’язаний із зарплатою, тобто вартістю найму. Також експерти мають свою працездатність. Найм здійснюється з урахуванням того, що сума заробітної плати всіх найнятих експертів, які працюють над проектами, не перевищуватиме бюджет, виділений на конкретний набір проектів. Крім того, враховано, що жоден спеціаліст у сформованому колективі не перевантажений або працює більше свого ліміту чи працездатності.

Для вирішення двооб'єктивної проблеми було запропоновано два жадібних алгоритми, а саме, стратегія вибору експертів і стратегія вибору проекту, щоб досягти всіх вищезгаданих атрибутів під час відбору команди. Обидва ці алгоритми використовували функції оцінки, які описані нижче [23].

Таблиця 1.2 - Позначення, що використовуються для жадібного алгоритму

E	Набір експертів
P	Набір проектів
$C(e)$	Вартість експерта e
$PF(p)$	прибуток від завершення проекту p
$Cap(e)$	здатність експерта e
$Dist(e, e')$	вартість зв'язку між двома експертами e і e'
λ	Параметр компромісу між прибутком і вартістю зв’язку
ε	Підсумкова група експертів

Стратегія «Вибір експерта» обирає експерта на кожній ітерації на основі його оцінки, і цей експерт додається до останньої групи експертів. Ось функція оцінки, яку вони використовували для призначення балів кожній парі експертів і проектів, що позначається $scep$.

$$sc_e^p \leftarrow \lambda . \frac{PF(p).\min\{Skill(e,p), Cap(e)\}}{C(e)} + (1-\lambda).\frac{a}{\sum_{e' \in \varepsilon} Dist(e,e')}$$

Де $Skill(e, p)$ представляє кількість навичок у проекті p, які може охопити експерт e, а a — це константа нормалізації. Інформацію про всі позначення можна знайти в таблиці 2 вище. Стратегія вибору експерта працює шляхом оцінки кожного експерта в соціальній мережі, який може охопити одну або кілька необхідних навичок, необхідних для успішного завершення проектів, таким чином, що вона охоплює численні навички високоприбуткового проекту, а вартість спілкування між експертами залишається низькою. Між кількістю навичок, які експерт e може охопити в p, і потужністю e, ми вибираємо мінімальне значення, щоб переконатися, що ми не перевищуємо можливості експерта. Вартість експерта також враховується під час оцінки, щоб переконатися, що найм здійснюється в рамках бюджету. На кожній ітерації вони призначають бали парі експертів/проектів і обирають експерта з найвищим балом.

Оскільки є незначні шанси, що найнятий експерт коштує дешево і може охопити багато навичок разом із мінімальною співпрацею, вони ввели λ як параметр компромісу між прибутком і вартістю зв'язку, який мав значення від 0 до 1. Цей компроміс корисно для визначення та розрахунку того, чи приділяти більше ваги витратам на зв'язок чи прибутку.

Говорячи про другу стратегію, стратегія вибору проекту вибирає проект на кожній ітерації, призначаючи бали неохопленим проектам, використовуючи їх функцію оцінки. Мета полягає в тому, щоб завершити проекти з максимальною маржею прибутку, набір експертів, найнятих для виконання проектів, коштує менше або в межах бюджету та може ефективно спілкуватися з досвідом минулої співпраці та досвіду роботи. На кожній ітерації спочатку вони знаходять команди для кожного неохопленого проекту, а потім використовують

функцію підрахунку балів для оцінки цих проектів. У кожній ітерації обирається проект з найвищим балом.

На кожній ітерації та для будь-якого неохопленого проекту вони знаходять набір експертів, щоб охопити цей проект. Щоб знайти цей набір експертів для проекту p, вони починають з порожнього набору Ep. Після цього вони вибирають експерта для додавання до Ep, який максимізує наступну функцію оцінки min $\{Skill(e, p), Cap(e)\}$ a:

$$sc_e \leftarrow \lambda . \frac{\min\{Skill(e,p), Cap(e)\}}{C(e)} + (1-\lambda) . \frac{a}{\sum_{e' \in E_p} Dist(e,e')}$$

Після того, як у нас буде набір експертів Ep для всіх розкритих проектів, наступним кроком буде оцінка проектів за допомогою наступної функції оцінки, і проект з найвищим балом буде доданий до пулу вибраного проекту на кожній ітерації.

$$\lambda . \frac{PF(p)}{\sum_{e \,\epsilon\, E_p} C(e)} + (1 - \lambda) . \frac{b}{\sum_{e \,\epsilon\, E_p} \sum_{e' \in \varepsilon} Dist(e,e')}$$

Усі позначення, використані в рівняннях, описані в таблиці 1.2 вище a і b, що використовуються в рівняннях, є просто нормалізаційними константами. Далі вони провели свої експерименти на реальному наборі даних і довели ефективність свого підходу з використанням жадібного алгоритму в порівнянні з випадковим алгоритмом, який використовується в соціальних мережах.

2 АЛГОРИТМІЧНІ ТА МАТЕМАТИЧНІ РІШЕННЯ ДЛЯ ЗАДАЧ ПРОЦЕСУ КЛАСТЕРИЗАЦІЇ

2.1 Постановка проблеми дослідження

Нехай $E = \{e1, e2, \dots, en\}$ позначимо набір n експертів; $S = \{s1, s2, \dots, sm\}$ позначають набір навичок m (Усі символи, використані в дипломній роботі, узагальнено в таблиці 3 нижче). Кожен експерт e володіє набором навичок, що позначається $ES(e)$. Кожен експерт e вимагає зарплату, тобто вартість найму для участі в проектах, що позначається $C(e)$. Вартість найму кожного експерта заздалегідь визначена, яка залишається незмінною незалежно від кількості проектів, призначених експертам, або кількості навичок, які мають експерти. Набір заданих проектів позначається $P = \{p1, p2, \dots, pk\}$. Кожен проект також складається з набору необхідних навичок, які повинні бути охоплені експертами для завершення проекту. Набір навичок, необхідних для проекту p, позначається $PS(p)$. Кожен експерт e має працездатність, що позначається $Cap(e)$, що є максимальною кількістю проектів, в яких експерт може брати участь одночасно. Працездатність полягає в тому, щоб ми не перевантажували експерта занадто великою кількістю проектів. Один експерт може брати участь у кількох проектах, але не може виконувати більше однієї навички для одного проекту. Нам також надається бюджет B, щоб найняти групу експертів для виконання ряду проектів.

Визначення 1. Група експертів: враховуючи все вищесказане, група експертів E може завершити підмножину проектів $\mathcal{P}$ P, якщо виконується наступне:

$\mathcal{P}$ і , експерт e в $\mathscr{E}$ призначений для виконання необхідної навички s для p.

2. , e не охоплює більше проектів, ніж $Cap(e)$.

1. , B, що означає, що сума вартості найму кожного експерта e в $\mathscr{E}$ знаходиться в межах бюджету B.

2. Загальна вартість комунікації $CC(\mathscr{E})$ мінімальна, а це означає, що вони мають високу співпрацю між експертами.

Таблиця 2.1 - Символи, використані в дипломній роботі

символи	Позначення
E	Набір n експертів
S	Набір навичок m
P	Набір k проектів
$ES(e)$	Набір навичок, яким володіє експерт e
$PS(p)$	Набір необхідних навичок за проектом p
$C(e)$	Вартість найму експерта e
$PF(p)$	Прибуток від виконання проекту p
$Cap(e)$	Здатність експерта e
ei	ith експерт
G	Граф соціальної мережі з n вузлів
$Dist(ei, ej)$	Вартість відстані/зв'язку між ei та ej у G
$CC(\mathscr{E})$	Вартість спілкування між групою експертів $\mathscr{E}$
B	Загальний бюджет для найму групи експертів для набору проектів

Існує соціальна мережа, змодельована як неорієнтований зважений граф G, де експерти з'єднані один з одним. Кожен експерт ei позначається як вузол графа G. У цій роботі терміни вузол і експерт використовуються як синоніми. Два експерта в графі з'єднані один з одним через ребро, якщо вони раніше співпрацювали, тобто працювали над тими ж проектами в минулому. Вага краю означає вартість зв'язку між цими двома експертами. Чим менша вага краю, тим міцніша співпраця двох експертів. Якщо між двома експертами немає прямої співпраці, вартість зв'язку між ними визначається вагою найкоротшого

шляху між двома експертами на графіку G. Якщо експерт не пов'язаний з жодним іншим експертом на графіку, ми розглядаємо вартість комунікації для даного експерта як нескінченність.

Ми прагнемо найняти групу експертів, де витрати на спілкування між ними мінімальні. Вартість комунікації між групою експертів розраховується як сума відстаней між кожною парою експертів у групі, яка відома як функція сумової відстані. На малюнку 7 ми маємо графік експертів з лівого боку, а ми відображаємо найману групу експертів для необхідного набору навичок з правого боку. Графік експертів на малюнку 7 представляє експертів у вигляді вузлів, зважені межі між експертами представляють вартість комунікації, а також наведені навички кожного експерта.

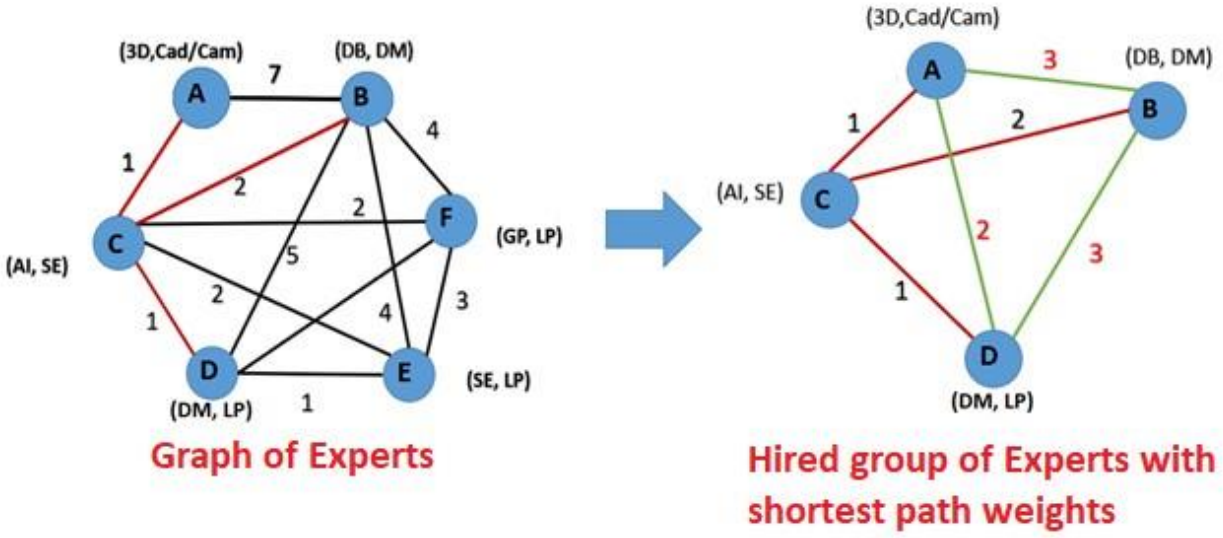


Рисунок 2.1 - Найкоротший шлях серед експертів

У групі найнятих експертів ми використовуємо вагу найкоротшого шляху між експертами як вартість зв'язку між експертами, коли в графі експертів немає переваги між експертами. Наприклад, експерти A і D не мають сполучного ребра в графі експертів на малюнку . Отже, ми знаходимо найкоротший шлях між A і D, який лежить від A до C і C до D. Складаємо

підсумок відповідні вагові коефіцієнти краю (1+1=2), щоб знайти значення вартості зв’язку між A та D. Тому ми відображаємо зважену межу між експертами A та D з вагою «2» на малюнку . Найкоротший шлях також може бути використовується для мінімізації вартості зв’язку між експертами, підключеними один до одного на графіку. Наприклад, на малюнку експерти A і B мають зважене ребро між ними з вагою «7» у графіку експертів, але існує найкоротший шлях між A та B через C, і вага цього найкоротшого шляху – це сума ваг між A і C, а потім C і B (1+2=3). Тому ми змінюємо вагу ребра між A і B від «7» до «3» на графіку праворуч на малюнку. Таким чином, ми знаходимо найкоротший шлях між кожним експертом у найнятій групі експертів. Ці найкоротші шляхи між експертами відображаються зеленим кольором, а ваги найкоротших шляхів відображаються червоним кольором на рисунку 2.1.

Вартість зв'язку між найманою групою експертів розраховується за сумовідстанним підходом. Ми найняли групу експертів із вагою краю між кожною парою експертів у групі на малюнку . Отже, ми виконуємо підсумовування відстаней між кожною парою експертів, щоб знайти вартість групового зв’язку. Наприклад,

- Необхідний набір навичок = {DB, AI, 3D, LP}
- Наймана група експертів = {B, C, A, D}
- Вартість групового спілкування = (1+2+3+2+1+3) = 12

2.2 Пропонована стратегія на основі модифікованого зваженого структурного алгоритму кластеризації для мереж

Ми обговоримо деякі важливі терміни, пов’язані з модифікованим зваженим структурним алгоритмом кластеризації для мереж (MWSCAN).

1. Графік експертів: його можна визначити як G(V, E, W), де вершини V: відноситься до набору експертів (E).

Ребра E: набір ребер між експертами, що представляють зв'язки між експертами. Коефіцієнти ваг W: набір вагових коефіцієнтів між експертами, що представляють вартість зв'язку між експертами.

2. Вартість зв’язку: її можна визначити як відстань/вагу між двома експертами ei і ej на графіку G. Його позначають $Dist(ei, ej)$. Коли між експертами не існує прямого зв’язку, вага найкоротшого шляху між двома експертами розглядається як вартість зв’язку між ними.

2. Сума відстаней. Вартість спілкування групи експертів – це сума відстаней між кожною парою експертів у групі, яка визначається як

$$xj{=}i{+}1\ Dist(ei, ej)$$

Тут x — кількість експертів у групі експертів[21].

4. Структура вершин: нехай ei (множина експертів), вершинна структура ei визначається його околицями, що позначається τ [21].

Тут (ei, ej) представляє межу між експертами ei і ej, яке належить до множини ребер E в графіку G.

5. Розширена структурна подібність: структурна подібність двох вершин ei і ej буде великою, якщо вони мають подібну структуру сусідів. Це означає, що вони мають високу частоту спільної роботи, а вартість зв’язку між ними буде низькою. Структурна подібність між експертами ei і ej позначається $\sigma(ei, ej)$.

$$\sigma(e_i, e_j) = \frac{\left|\tau(e_i) \cap \tau(e_j)\right|}{\sqrt{\left|\left(\tau(e_i)||\tau(e_j)\right)\right|}\ Dist(e_i, e_j)}$$

Тут $\sigma(ei, ej)$ обернено пропорційна вартості зв’язку. Якщо $\sigma(ei, ej)$ висока, то це означає, що вартість спілкування між експертами ei і ej низька σ

$$(e_i, e_j) \propto \frac{1}{Dist^{(e_i, e_j)}}$$

Припустимо, якщо ei і ej мають високу вартість зв'язку між ними, то частота спільної роботи низька і навпаки.

5. Околиці (ε): ми застосовуємо поріг ε до структурної подібності, коли призначаємо членство в кластері експерту ei і, застосовуючи поріг ε, можемо визначити ε сусідство експерта, яке позначається як $N\varepsilon(e_i)$.

$$N\varepsilon(e_i) = \{e_j \in \tau(e_i) \mid \sigma(e_i, e_j) \geq \varepsilon\}$$

Експерт ядра: вершина $ei\ E$ називається ядром з посиланням на ε і μ , якщо її околиця містить принаймні μ вершин. Основний експерт позначається $Core(\epsilon,\mu)(ei)$, якщо він задовольняє наступній умові:

$$|N|\varepsilon = \mu$$

Де μ – кількість експертів сусідства, підключених до основної вершини (експерта високої зв’язки).

Алгоритм MWSCAN

Алгоритм MWSCAN — це наше рішення проблеми Cluster Hire у соціальних мережах. Цей алгоритм отримує набір n експертів і набір k проектів, графік соціальної мережі G і призначений бюджет B як вхідні дані. У наших умовах кожен експерт має навички, працездатність та вартість найму, і кожен проект має свій набір необхідних навичок. Результатом алгоритму буде група експертів $\mathscr{E}$, які зможуть охопити необхідні навички підмножини проектів $\mathcal{P}$, мінімізуючи витрати на зв’язок між групою експертів $\mathscr{E}$. Найм експертів здійснюється в межах призначеного бюджету, тому вартість найму групи експертів не перевищує встановлений бюджет B.

Алгоритм: Cluster Hire у соціальній мережі за допомогою MWSCAN

Вхідні дані: графік соціальної мережі G(V, E, W), набір експертів n E = $\{e1, e2, \dots, en\}$, набір k проектів P = $\{p1, p2, \dots, i$ має кожен експерт навички $ES(ei)$, працездатність $Cap(ei)$ та вартість найму $C(ei)$, пов'язана з цим. Кожен проект pi має набір необхідних навичок $PS(pi)$. У нас також є бюджет B, призначений для набору проектів P.

Вихід: група експертів $\mathscr{E} \subseteq E$ для підмножини проектів $\mathcal{P} \subseteq P$, де $CC(\mathscr{E})$ мінімальна.

1. Знайдіть пул експертів (PoE) від E

2. Усі вершини некласифіковані

3. Для кожної некласифікованої вершини $ei \in E$;

3.1 Підрахуйте кількість експертів у ε-околиці експерта ei

4. Експерт ei з найбільшою кількістю підключень у $N\varepsilon(ei)$ буде основним експертом

4.1 Якщо більше одного експерта мають однакову кількість експертів у $N\varepsilon(ei)$, то

4.1.1 Експерт ei, який охоплює більшу кількість навичок для P в межах $Cap(ei)$, буде основним експертом

4.1.2 Якщо експерти охоплюють однакову кількість навичок, то

4.1.2.1 Вибрати основного експерта випадковим чином з експертів з найбільшою кількістю зв'язків у $N\varepsilon(ei)$

5. Загальна_вартість += $C(ei)$

6. Якщо Загальна_вартість ≤ B, то

6.1 Створіть новий кластер від основного експерта ei

6.2 Вставте основного експерта ei до кінцевої групи експертів $\mathscr{E}$

6.3 Вилучіть навички, отримані ei, із необхідного набору навичок на основі $Cap(ei)$

7. Для кожного $ej \in N\varepsilon(ei)$;

7.1 Отримайте $Dist(ei, ej)$ від основного експерта ei Кінець для

8. У той час як (необхідний набір навичок не порожній ТА Total_cost ≤ B) зробіть

8.1 Виберіть експерта ej з мінімальним значенням $Dist(ei, ej)$

8.2 Якщо більше ніж один експерт має однаковий $Dist(ei, ej)$, то

8.2.1 Виберіть експерта ej, який охоплює більшу кількість навичок для P в межах $Cap(ei)$

8.2.2 Якщо експерти охоплюють однакову кількість навичок, то

8.2.2.1 вибрати експерта випадковим чином з експертів з мінімальним значенням з $Dist(ei, ej)$

8.3 Якщо ej охоплює будь-які навички з необхідного набору навичок у межах $Cap(ej)$, тоді8.3.1 Total_cost += $C(ej)$

8.3.2 If Total_cost ≤ B, then

8.3.2.1 Add expert ej into final group of experts $\mathscr{E}$

8.3.2.2 Remove skills fulfilled by ej from the required skillset based on $Cap(ej)$

9. Повернути групу експертів $\mathscr{E}$

10. Кінець

Алгоритм починається з пошуку пулу експертів (PoE) із заданого набору експертів E шляхом видалення всіх експертів, які не мають жодних навичок, із необхідного набору навичок проектів. У наведеному графі G всі вершини некласифіковані на початку. Ми починаємо зі сканування кожної вершини та підрахунку кількості вершин сусідства, які існують у ε-околиці. Експерт з найбільшою кількістю експертів у своєму ε-околиці буде нашим основним експертом. У випадку, якщо більше одного експерта мають однакову кількість зв'язків у своєму районі, ми вибираємо основного експерта, який охоплює більше навичок у необхідному наборі навичок у межах їхньої працездатності. Коли ми знайдемо основного експерта, ми починаємо формувати кластер навколо основного експерта з порогом ε. Усі експерти, присутні в ε-околиці основного експерта, будуть частиною кластеру. Ми також додаємо основного експерта до останньої групи експертів $\mathscr{E}$ і вилучаємо навички, якими володіє

основний експерт, із необхідного набору навичок. До загальної вартості додаємо вартість найму основного експерта.

Для решти необхідного набору навичок ми починаємо сканувати експертів у ε-околиці основного експерта та отримуємо від основного експерта *відстань/вартість* зв'язку сусіда. Спочатку ми обираємо сусіднього експерта з мінімумом основного експерта та перевіряємо, чи надає обраний експерт будь-які навички в решті необхідного набору навичок. Якщо вони забезпечують будь-які навички в необхідному наборі навичок, ми додаємо вартість найму експерта до загальної вартості. Як тільки ми переконаємося, що загальна вартість знаходиться в межах бюджету, ми згодом вилучаємо відповідні навички експерта з необхідного набору навичок, пам'ятаючи про працездатність експерта. Вибраний експерт додається до остаточної групи експертів $\mathscr{E}$. Ми дотримуємося того ж процесу, поки не знайдемо експертів з усіма необхідними навичками або поки призначений бюджет не буде вичерпано. У підсумку ми отримаємо остаточну групу експертів $\mathscr{E}$, яку наймають для виконання заданого набору проектів з мінімальними витратами на комунікацію між групою експертів.

Приклад: Ми обговоримо невеликий приклад проблеми Cluster Hire у соціальних мережах та її вирішення за допомогою запропонованого нами підходу MWSCAN.

Таблиця 2.2 - Набір проектів

Проекти	Необхідні навички:	Необхідна кількість експертів	Прибуток Бюджет
P1	Java: 2	50000	50000
P2	Мережа: 1 Звіт: 1	35000	

Таблиця 2.3 - Набір експертів

Ідентифікатор експерта	Експертні навички	Вартість/Можливість	заробітна плата
A	Java, дані	5000	5
B	Звіт, аналітика, мережа	7000	2
C	Безпека, великі дані	8000	3
D	Звіт, дані, хмара	4500	4
E	Java, C++, дані	10000	4
F	SE, Алгоритм, PM	7500	3
G	Хмара, тестування	6000	5
H	БД, Python, Безпека	3500	1

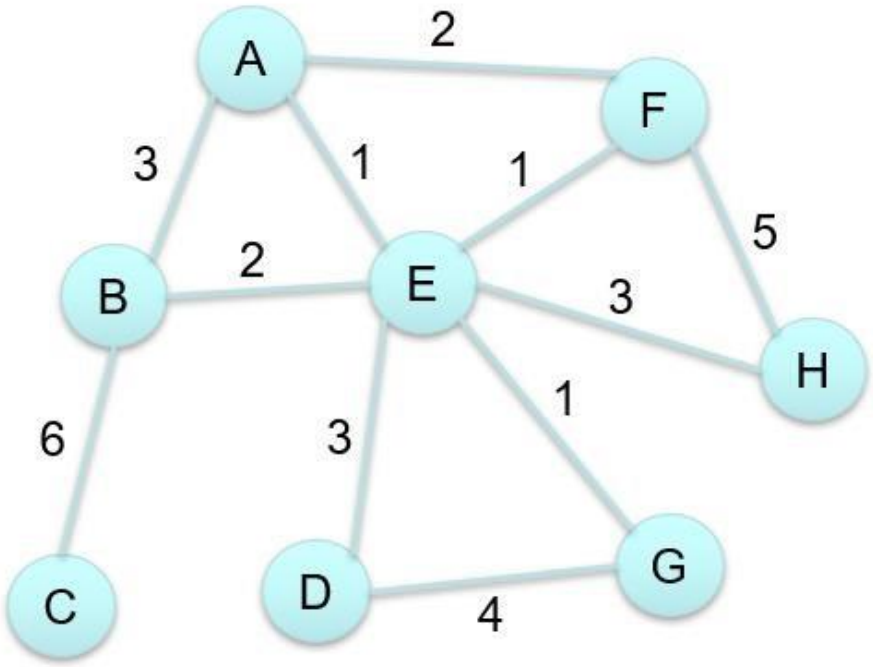


Рисунок 2.2 - Графік експертів

Проблема: Для даного набору проектів і набору експертів у таблицях 4 і 5 ми повинні знайти групу експертів, які зможуть виконати вимоги проектів у межах бюджету 50 000, зберігаючи низьку вартість комунікації. Будь ласка, зверніть увагу, що на малюнку 8 наведено графік експертів, що представляє значення вартості зв’язку між експертами як ваги краю.

Рішення:

Крок 1: Знайдіть пул експертів (PoE), видаливши всіх експертів, які не можуть виконати будь-які необхідні навички для даних проектів. Таким чином, ми виключаємо експертів C і G з малюнка 8, щоб отримати пул експертів (PoE). Пул експертів для даного сценарію представлений на рисунку 2.3 нижче.

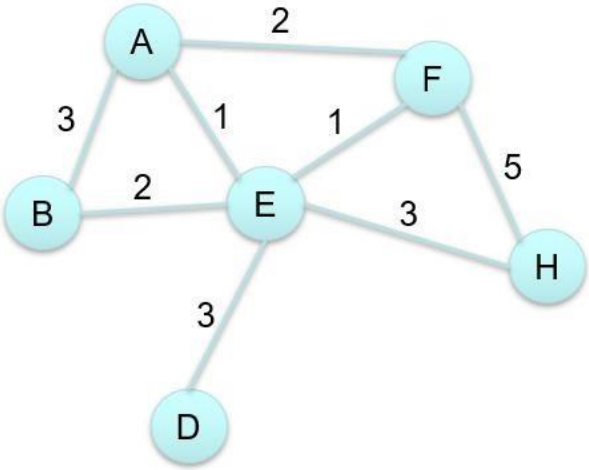


Рисунок 2.3 - Пул експертів (PoE)

Крок 2. Наступним кроком є пошук основного експерта з пулу експертів. Основним експертом буде експерт, який має максимальну кількість експертів у своєму *ε*- околиці. У нашому сценарії експерт E має найбільшу кількість експертів (5) у своєму *ε*- околиці. Таким чином, експерт E є основним експертом із набору експертів.

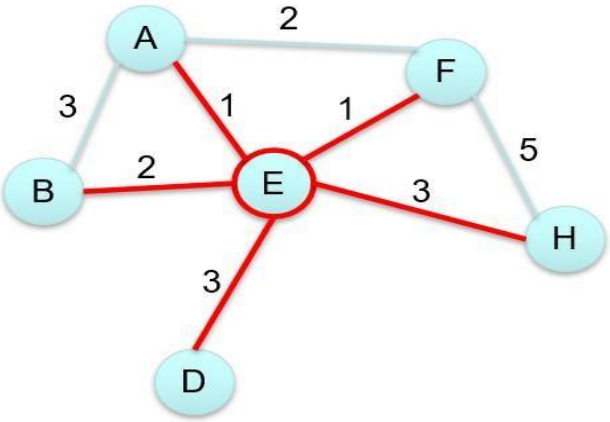


Рисунок 2.4 - Основний експерт E

Крок 3. Переконавшись, що вартість найму експерта E знаходиться в межах призначеного бюджету в 50 000, ми додаємо основного експерта до останньої групи експертів. Ми також видаляємо відповідні навички, отримані основним експертом, а оновлені необхідні навички для проектів представлені в таблиці 6 нижче. Навички, якими володіє найнятий експерт на кожному кроці, виділені у таблицях.

Залишок бюджету: 50000 – Вартість експерта E = 50000 – 10000 = 40000

Група експертів: {E}

Таблиця 2.3 - Крок 3 – Навики, що залишилися в проектах

Проекти	Необхідні навички	Необхідна кількість експертів	прибуток
P1	Java: Network:	1 1	50000
P2	Report: Data: Algorithm:	1 1 1	35000

Крок 4. Для решти необхідних навичок ми шукаємо експертів по сусідству з основним експертом, вибираючи спочатку експерта з найменшими витратами на спілкування. Тут експерти A та F мають вартість комунікації 1. У цьому випадку ми вибираємо експерта, який може реалізувати більшу кількість навичок у необхідних навичках проектів, пам’ятаючи про їхню працездатність. У цьому прикладі експерт A може охоплювати дві навички (P1-Java, P2-Data) і мати працездатність 5. Таким чином, він може виконувати дві навички для двох проектів. Тоді як Експерт F може охопити лише один навик (алгоритм P2), і це в межах його працездатності (3). Тому ми наймаємо експерта A і додаємо його до кінцевої групи експертів для відповідних вимог проекту.

Залишок бюджету: 40000 – Вартість експерта A = 40000 – 5000 = 35000

Група експертів: {E, A}

Таблиця 2.4 - Крок 4 – Навики, що залишилися в проектах

Проекти	Необхідні навички: необхідна кількість експертів	прибуток
P1	Java: 0 Мережа: 1	50000
P2	Звіт: 1 Дані: 0 Алгоритм: 1	35000

Крок 5: Ми будемо вибирати всіх наступних експертів основного експерта, поки не досягли всіх необхідних навичок проектів, або не вибираємо бюджет, призначений для найбільших експертів для даного набору проектів. Оскільки спочатку ми виберемо експерта з мінімальною комунікацією, ми виберемо експерта F, який відповідає одному навику (алгоритму) у проекті P2. Таким чином, експерт F буде доданий до групи експертів після перевірки бюджету.

Залишок бюджету: 35000 – Вартість експерта F = 35000 – 10000 = 25000

Група експертів: {E, A, F}

Таблиця 2.5 - Крок 5

Проекти	Необхідні навички: необхідна кількість експертів	прибуток
P1	Java: 0	50000
P2	Мережа: 1	35000

Крок 6: Потім ми обираємо наступного експерта з мінімальними витратами на зв’язок з основним експертом, щоб виконати інші необхідні навички для набору проектів. Експерт B вибирається, оскільки він має найменші витрати на зв’язок з основним експертом. Експерт B може охопити дві навички (P1-Мережа, P2-Report) з необхідного набору навичок для проектів.

Залишок бюджету: 25000 – Вартість експерта B = 25000 – 7000 = 18000

Група експертів: {E, A, F, B}

Таблиця 2.6: Крок 6 – Навики, що залишилися в проектах

Проекти	Необхідні навички: обов'язкові кількість експертів	прибуток
P1	Java: 0 Мережа: 0	50000
P2	Звіт: 0 Дані: 0 Алгоритм: 0	35000

Після кроку 6 усі вимоги до даних проектів будуть виконані, як показано в таблиці 9. Отже, наша остаточна група експертів для цих двох проектів буде:

Остаточна група експертів: {E, A, F, B}

Нам вдалося найняти вищезгадану групу експертів для даного набору проектів, у межах даного бюджету в 50 000, зберігаючи мінімальну вартість комунікації між експертами. Це один із прикладів вирішення проблеми найму кластерів у соціальних мережах за допомогою MWSCAN. У наступному розділі ми обговоримо деталі впровадження з результатами та аналізом.

3 ЗАСТОСУВАННЯ КЛАСТЕРНИХ МОДЕЛЕЙ ТА АЛГОРИТМІВ ДЛЯ ВИРІШЕННЯ CLUSTER HIRE В СОЦІАЛЬНИХ МЕРЕЖАХ

У цьому розділі ми оцінимо ефективність запропонованого нами підходу до вирішення проблеми Cluster Hire в соціальних мережах.

3.1 Програмні засоби для проведення імітаційного моделювання та тестування

Усі експерименти проводилися на комп'ютері зі специфікаціями пристрою з процесором Intel Core i76700 @ 3,40 ГГц на 64-розрядній операційній системі Windows з 8 ГБ оперативної пам'яті. Ми реалізували наш алгоритм у JDK 14.0.1 (windows-64 bit) за допомогою IntelliJ IDEA 2020.1.3.

Для тестування нашого алгоритму в реальній соціальній мережі ми використовуємо набір даних DBLP, який є однією з мереж експертів. За даними dblp.org, DBLP — це бібліографічний веб-сайт з інформатики, який містить набір даних із понад 5,4 мільйонів публікацій, журналів і конференційних робіт з інформатики. У нашій установці ми моделюємо набір даних DBLP як графік соціальної мережі, де вузли представляють авторів/експертів. Коли два експерти публікують будь-яку роботу разом, вони матимуть зв'язок/межу між ними. З ним пов'язана вага, яка називається вартістю зв'язку. Якщо два експерта не пов'язані безпосередньо, то ми використовуємо вагу найкоротшого шляху між ними як вартість зв'язку. Набір даних містить інформацію про 50 тисяч експертів.

Набір даних експертів зберігається у форматі JSON, де кожен експерт має чотири атрибути, а саме: ідентифікатор експерта, навички, вартість/зарплата та потужність. Ідентифікатор експерта – це унікальний ідентифікатор для кожного експерта. Кожен експерт має набір навичок, які він може запропонувати. Навички експертів витягуються з назв наукових статей/публікацій експертів у

мережі DBLP. Вартість найму експерта – це заробітна плата або грошова оцінка, призначена експерту. Вартість найму експерта випадковим чином встановлюється до 100 000 у нашому наборі даних. Кожен експерт також має працездатність, яка вказує на кількість проектів, над якими експерт може працювати одночасно. Працездатність експерта випадковим чином встановлюється від 1 до 5. Ми використовуємо різну кількість проектів для отримання результатів для експериментів, таких як 3, 5, 7, 10 і 14 проектів. Для кожного проекту ми призначаємо від 3 до 7 навичок, які необхідно виконати, щоб завершити проект. Ми повинні призначити по одному експерту для кожної необхідної навики проекту, а це означає, що кількість експертів, призначених для проекту, буде дорівнювати кількості навичок, необхідних для проекту. Один експерт не може охопити кілька навичок в одному проекті, тобто кожен експерт буде виконувати унікальні навички в даному проекті. Кожен проект має визначену величину прибутку, яка формується після завершення проекту. Ми встановлюємо пріоритет для проектів, виходячи з величини їх прибутку. У деяких випадках ми не можемо охопити всі проекти певного набору проектів через бюджетні обмеження; потім ми наймаємо експертів для проектів з більшим прибутком, а потім для проектів з нижчим прибутком. У наших експериментах ми також використовуємо різні значення бюджету для аналізу результатів, отриманих нашим алгоритмом.

Запропонований алгоритм має на вході декілька файлів наборів даних. Нижче наведено список файлів вхідних наборів даних до алгоритму MWSCAN для задачі Cluster Hire у соціальних мережах,

1. Файл, що містить інформацію про експертів, де ми маємо набір даних, що складається з 50 тисяч експертів/вузлів. На рисунку 3.1 ми маємо інформацію про двох експертів, і ви можете ідентифікувати експертів за їхніми ідентифікаторами експертів, які 1223 і 1224, виділеними червоним кольором. Для кожного ідентифікатора експерта ми маємо три атрибути: навички,

зарплата та потенціал. Ця інформація про експертів також наведена в таблиці 3.1.

Таблиця 3.1 - Файл JSON експертів

Expert ID	Навички	Заробітна плата/Вартість	Ємність
1223	"the", "мережі", "класифікація", "ітерація", "через", "вибір", консенсус","для","ефективні","гетерогенні інші","спам","і"," висновок","виявлення","паттерн rn","відкриття","соціальний","огляд","особливість"	74412	2
1224	"невизначений", "mpp", "topk"," дублікат", "підтримка"	48134	2

Навички експерта відображаються у вигляді масиву навичок для кожного ідентифікатора експерта. Тоді ми маємо значення зарплати та потужності для кожного ідентифікатора експерта



Рисунок 3.1 - Файл JSON експертів

Файл, що містить інформацію про зв'язки/грани між експертами в соціальній мережі експертів. Соціальна мережа експертів – це місце, де вузли представляють експертів, а експерти пов’язані з ребрами, які зважені. Якщо два експерта пов’язані в соціальній мережі, це означає, що у них є зв’язок і вони співпрацювали в минулому. Якщо існує зв’язок між двома експертами, тоді між експертами призначається значення 1,0. Значення 1,0 не відображає вагу між експертами, але це просто означає, що у них є зв’язок і вони співпрацювали в минулому. На рисунку, якщо ми подивимося на перший рядок, виділений червоним, він відображає всі зв’язки експерта ID «1» з іншими експертами в соціальній мережі. Ми виділили кожне з’єднання окремо червоним кольором для ідентифікатора експерта «1». Наприклад, перший елемент у виділеному

розділі на малюнку 12, «3072#1.0» означає, що ідентифікатору експерта «3072» присвоєно значення 1,0, що означає, що існує зв’язок між експертними ідентифікаторами «1» і «3072». Таким чином, він має список ідентифікаторів експертів, таких як 3072, 2, 4, 5, 14, 2062, 19471 і 16 зі значенням 1,0, що означає, що ці експерти мають зв’язок з ідентифікатором експерта «1». Таким чином, файл зберігає інформацію про всіх експертів та їхні зв’язки в соціальній мережі експертів. Цей файл допомагає нам підрахувати кількість підключень для будь-якого експерта в наборі даних.

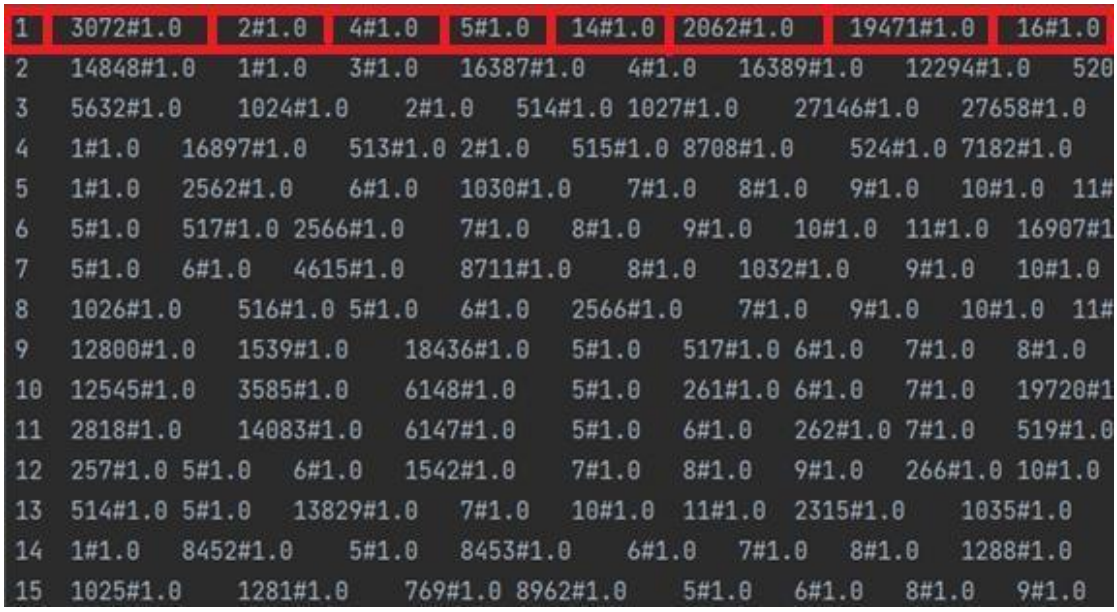


Рисунок 3.2 - Файл зв'язків у соціальній мережі експертів

Файл, що містить інформацію про вартість комунікації між експертами в соціальній мережі експертів. Як обговорювалося раніше, у мережі експертів експерти з’єднані один з одним за допомогою зваженого краю. Вартість комунікації може бути визначена як вага/відстань між експертами, а вартість комунікації базується на минулій співпраці між експертами. Якщо два експерта не пов’язані безпосередньо, то ми використовуємо значення найкоротшого шляху між ними як вартість зв’язку. Нижче значення вартості комунікації означає більше співпраці експертів. Цей файл створюється за допомогою підходу з двома стрибками, запропонованого, як обговорено. Покриття з 2

стрибками допомагає підрахувати найкоротшу відстань шляху між вузлами у великих мережах; інакше це може зайняти дуже багато часу. На рисунку 3.3 ми відображаємо значення вартості зв’язку між різними ідентифікаторами експертів у соціальній мережі експертів. Наприклад, якщо ми виберемо елемент, виділений червоним кольором на малюнку, а саме «6#148.0#14», це означає, що вартість зв’язку між експертними ідентифікаторами «6» і «14» дорівнює 148,0. У цьому форматі інформація про вартість зв’язку зберігається між усіма експертами соціальної мережі експертів.

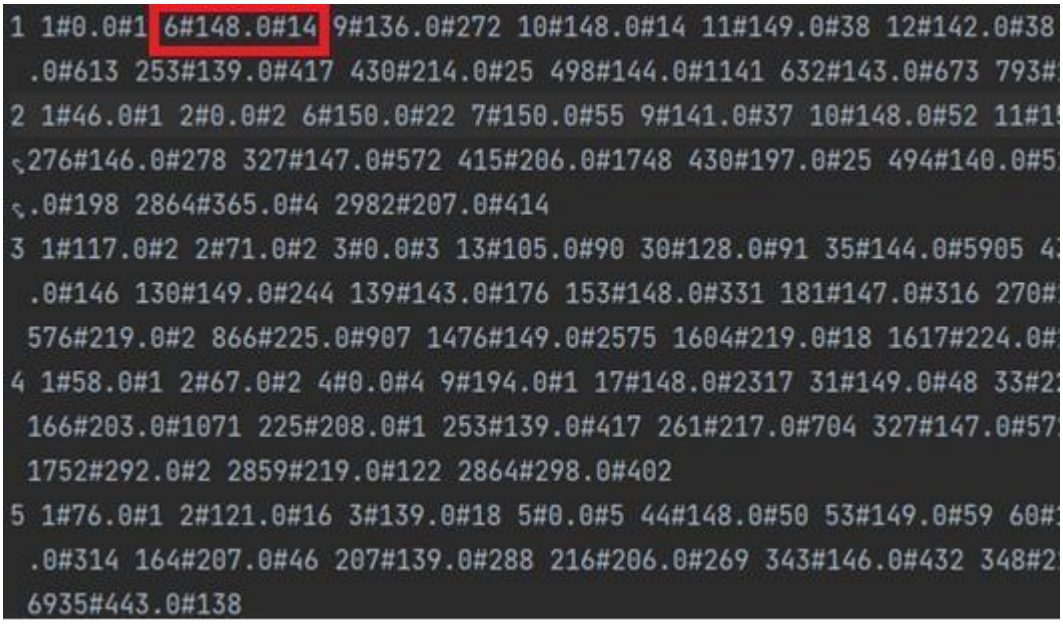


Рисунок 3.3 - Файл вартості зв'язку

Файл з інформацією про вимоги до проекту, наприклад, необхідні навички кожного проекту, прибуток від кожного проекту та бюджет, призначений для даного набору проектів. На малюнку нижче ми маємо групу з трьох проектів, для яких нам потрібно найняти групу експертів. Бюджет, призначений для найму для даної групи проектів, становить 500 тис. Кожному проекту призначається прибуток і набір необхідних навичок, який зберігається в парі ключ і значення. Наприклад, якщо ми подивимося на перший проект на малюнку 14, який виділено червоним кольором. Прибуток від цих проектів становить 500 000, а набір необхідних навичок – {“сервер”, “апаратура”,

“оцінка”, «майнінг», «обчислювання»}. Інформація про інші два проекти зберігається в тому ж форматі. Зверніть увагу, що ми використовуємо вартість прибутку кожного проекту лише для встановлення пріоритету між проектами. Наприклад, якщо ми не можемо завершити всі проекти з даного бюджету, ми віддаємо пріоритет проектам з більшим прибутком.



Рисунок 3.4 - Файл вимог проекту

Основною метою нашої дипломної роботи є вивчення та аналіз впливу алгоритму MWSCAN при застосуванні до проблеми Cluster Hire у соціальних мережах, яка є NP-складною проблемою. Ми оцінюємо продуктивність алгоритму MWSCAN з точки зору кількості завершених проектів, прибутку, отриманого від завершених проектів, і часу виконання алгоритму для різних значень бюджету та кількості проектів. Крім того, ми порівнюємо наші результати з попереднім жадібним алгоритмом, запропонованим Sagarika et al. для вирішення проблеми кластеру Hire у соціальних мережах.

3.2 Результати виконаного аналізу даних

У цьому розділі ми обговоримо результати, отримані при виконанні різних експериментів для даного набору даних із застосуванням запропонованого алгоритму MWSCAN. Набір даних 50 тисяч експертів і соціальна мережа

експертів залишаться однаковими для всіх експериментів, і ми змінюємо кількість проектів у групі для наших експериментів. Ми будемо проводити експерименти з різною кількістю проектів у групі, як показано в таблиці нижче. Для кожного експерименту ми будемо аналізувати вихід на основі кількості завершених проектів, часу виконання алгоритму, а також переконатися, що витрати на найм експертів знаходяться в межах заданого бюджету.

Таблиця 3.2 - Випадки для експериментів

Номер експерименту	Кількість проектів	Бюджет
1	3	500k
2	5	500k
3	7	500k
4	10	500k
5	14	500k

У експерименті 1 ми маємо набір із трьох проектів, для виконання яких ми хочемо найняти групу експертів, щоб отримати навички, необхідні для проектів. Для даного набору з трьох проектів ми встановили значення бюджету в 500 тис. У таблиці 3.3 нижче показано інформацію про вимоги до проекту для експерименту 1. У нас є вся інформація про проекти, як-от необхідний набір навичок, бюджет, призначений для групи проектів, та прибуток від проектів.

Таблиця 3.3 - Вимоги до проекту для експерименту 1

Проекти	Необхідні навички	прибуток	Бюджет
Проект 1	сервер, апаратне забезпечення, оцінка, майнінг, обчислення	500k	500k
Проект 2	сервер, мережа, орієнтований на дані, хмара	250k	
Проект 3	сервер, карта, графік, майнінг, хмара, обчислення	400k	

Для даної групи проектів ми можемо найняти групу експертів з мінімальними витратами на комунікацію для виконання всіх трьох проектів. Вартість найманої групи експертів знаходиться в межах даного бюджету, і ми наймаємо експертів, які співпрацюють, тобто мають мінімальні витрати на комунікацію між собою. У таблиці 3.4 нижче ми відображаємо призначеного експерта для кожного необхідного навику для цих трьох проектів.

Таблиця 3.4 - Проектні завдання для експерименту 1

Проекти	Необхідні навички	Призначений ідентифікатор експерта	Прибуток
Проект 1	сервер	151	500k
	апаратне забезпечення	123	
	оцінити	219	
	видобуток	18	
	обчислення	451	
Проект 2	сервер	151	250k
	мережі	28505	
	орієнтований на дані	86	
	апаратне забезпечення	123	
Проект 3	сервер	151	400k
	карта	1456	
	графік	1	
	видобуток	18	
	хмара	19	
	обчислення	451	

Як показано на рисунку 3.5, витрати на найм групи експертів становлять лише 34 764 особи, що входить до даного бюджету, тому ми змогли завершити всі три проекти із заданого набору проектів. Тривалість даного експерименту

становить 59 секунд. Результат для експерименту 1 показаний на рисунку 3.5, де ми бачимо список найнятих експертів разом з їх інформацією, як-от їхній ідентифікатор експерта, призначені навички, потужність та вартість/зарплата.

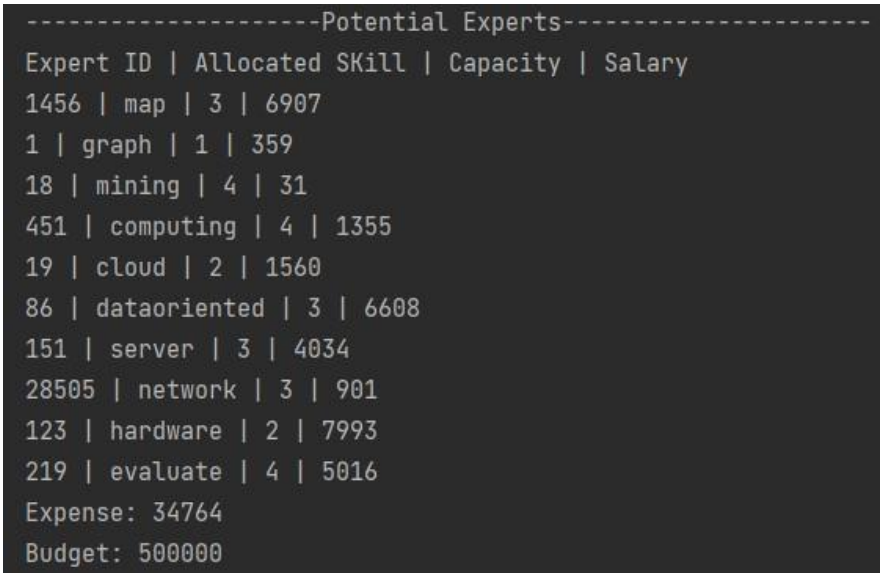


Рисунок 3.5 - Результат експерименту 1

Для експерименту 2 ми збільшуємо кількість проектів до п’яти проектів, але зберігаємо значення бюджету незмінним. У таблиці 3.5 нижче представлена інформація про вимоги до проекту для другого експерименту. У нас є більше проектів і більше навичок, які потрібно охопити, наймаючи експертів, ніж експеримент 1.

Таблиця 3.5 - Вимоги до проекту для експерименту 2

Проекти	Необхідні навички	Бюджет	прибуток
Проект 1	сервер, апаратне забезпечення, оцінка, майнінг, обчислення	500k	500k
Проект 2	сервер, мережа, орієнтований на дані, хмара	250k	
Проект 3	сервер, карта, графік, майнінг, хмара, обчислення	400k	
Проект 4	аналітика, база даних, обчислення, тестування	275k	
Проект 5	алгебра, обчислення, карта	200k	

На рисунку 3.6 зображено результат експерименту 2. У ньому перераховано групу експертів, які володіють навичками, необхідними для цих п’яти проектів. Знову ж таки, ці найняті експерти знаходяться в межах даного бюджету, і вони найняті таким чином, що група експертів співпрацює з мінімальними витратами на комунікацію між найнятими експертами. Витрати на найм експертів – 54291, що в межах бюджету 500 тис. Час виконання цього експерименту становить 14 хвилин 13 секунд, що значно більше, ніж у попередньому випадку.

Ми помічаємо, що наймається більша кількість експертів, які виконують усі необхідні навички за розумну вартість. Крім того, ми відображаємо індивідуальне завдання експерта для кожного навику в даній групі проектів у таблиці 3.5.

```
--------------------Potential Experts---------------------
Expert ID | Allocated SKill | Capacity | Salary
1 | graph | 1 | 359
451 | computing | 4 | 1355
5898 | algebra | 4 | 9335
1456 | map | 3 | 6907
272 | analytics | 4 | 1883
18 | mining | 4 | 31
19 | cloud | 2 | 1560
7187 | testing | 4 | 8089
86 | dataoriented | 3 | 6608
151 | server | 3 | 4034
28505 | network | 3 | 901
219 | evaluate | 4 | 5016
123 | hardware | 2 | 7993
447 | database | 4 | 220
Expense: 54291
Budget: 500000
```

Рисунок 3.6 - Результат експерименту 2

Таблиця 3.6. - Проектні завдання для експерименту 2

Проекти	Необхідні навички	Призначений ідентифікатор експерта	Прибуток
Проект 1	сервер	151	500k
	апаратне забезпечення	123	
	оцінити	219	
	видобуток	18	
	обчислення	451	
Проект 2	сервер	151	250k
	мережі	28505	
	орієнтований на дані	86	
	апаратне забезпечення	123	
	сервер	151	400k
Проект 3	карта	1456	
	графік	1	
	видобуток	18	
	хмара	19	
	обчислення	451	
Проект 4	аналітика	272	275k
	бази даних	447	
	обчислення	451	
	тестування	7187	
Проект 5	алгебра	5898	200k
	обчислення	451	
	карта	1456	

В експерименті 3 кількість проектів збільшено до групи з семи проектів. Як показано в таблиці 3.7, бюджет встановлено на рівні 500 000. Ми маємо набір необхідних навичок і величину прибутку для кожного проекту в групі, як показано в таблиці.

Таблиця 3.7 - Вимоги до проекту для експерименту 3

Проекти	Необхідні навички	Прибуток	Бюджет
Проект 1	сервер, апаратне забезпечення, оцінка, майнінг, обчислення	500k	500k
Проект 2	сервер, мережа, орієнтований на дані, хмара	250k	
Проект 3	сервер, карта, графік, майнінг, хмара, обчислення	400k	
Проект 4	аналітика, база даних, обчислення, тестування	275k	
Проект 5	алгебра, обчислення, карта	200k	
Проект 6	хмара, інфраструктура, дизайн, база даних	300k	
Проект 7	алгоритм, тестування, дизайн	350k	

Після виконання нашого алгоритму для експерименту 3 ми змогли найняти експертів для всіх семи проектів у межах даного бюджету. Групу експертів наймають із мінімальними витратами на зв’язок між ними. Як показано на малюнку, витрати на найм експертів становлять 56172, що входить до даного бюджету. Тривалість даного експерименту становила 18 хвилин 28 секунд. У таблиці наведено індивідуальне експертне завдання для кожної необхідної навички в даній групі проектів. Ми показуємо ідентифікатор експерта, призначений для кожного необхідного навику з нашого результату.

```
--------------------Potential Experts--------------------
Expert ID | Allocated SKill | Capacity | Salary
1 | graph | 1 | 359
451 | computing | 4 | 1355
5898 | algebra | 4 | 9335
17167 | design | 3 | 552
1456 | map | 3 | 6907
272 | analytics | 4 | 1883
18 | mining | 4 | 31
7187 | testing | 4 | 8089
2900 | infrastructure | 2 | 1188
86 | dataoriented | 3 | 6608
151 | server | 3 | 4034
1497 | algorithm | 3 | 244
28505 | network | 3 | 901
7098 | cloud | 3 | 1997
219 | evaluate | 4 | 5016
123 | hardware | 2 | 7993
447 | database | 4 | 220
Expense: 56712
Budget: 500000
```

Рисунок 3.7 - Результат експерименту 3

Таблиця 3.8 - Проектні завдання до експерименту 3

Проекти	Необхідні навички	Призначений ідентифікатор експерта	Прибуток
Проект 1	сервер	151	500k
	апаратне забезпечення	123	
	оцінити	219	
	видобуток	18	
	обчислення	451	
Проект 2	сервер	151	250k
	мережі	28505	
	орієнтований на дані	86	
	апаратне забезпечення	123	

Проект 3	сервер	151	400k
	карта	1456	
	графік	1	
	видобуток	18	
	хмара	7098	
	обчислення	451	
Проект 4	аналітика	272	275k
	бази даних	447	
	обчислення	451	
	тестування	7187	
Проект 5	алгебра	5898	200k
	обчислення	451	
	карта	1456	
Проект 6	хмара	7098	300k
	інфраструктури	2900	
	дизайн	17167	
	бази даних	447	
Проект 7	алгоритм	1497	350k
	тестування	7187	
	дизайн	17167	

У експерименті 4 ми наймаємо групу експертів для десяти проектів. Ми зберігаємо значення бюджету незмінним на рівні 500 000, як показано в Таблиці 3.9. Мета полягає в тому, щоб найняти групу експертів для всіх десяти проектів з мінімальними витратами на комунікацію в межах даного бюджету. Ми також будемо відстежувати час виконання нашого алгоритму.

Таблиця 3.9 - Вимоги до проекту для експерименту 4

Проекти	Необхідні навички	Profit	Budget
Проект 1	сервер, апаратне забезпечення, оцінка, майнінг, обчислення	500k	500k
Проект 2	сервер, мережа, орієнтований на дані, хмара	250k	
Проект 3	сервер, карта, графік, майнінг, хмара, обчислення	400k	
Проект 4	аналітика, база даних, обчислення, тестування	275k	
Проект 5	алгебра, обчислення, карта	200k	
Проект 6	хмара, інфраструктура, дизайн, база даних	300k	
Проект 7	алгоритм, тестування, дизайн	350k	
Проект 8	база даних, майнінг, хмара	375k	
Проект 9	алгоритм, обчислення, тестування, дані	180k	
Проект 10	хмара, база даних, мережа, алгоритм	420k	

Після запуску алгоритму MWSCAN для наведених десяти проектів ми змогли успішно найняти групу експертів для всіх 10 проектів у межах даного бюджету. На рисунку 3.8 показані результати експерименту 4 з інформацією про всіх найнятих експертів та загальну вартість найму експертів. Витрати на найм експертів – 62284, що входить до бюджету. Час виконання цього конкретного експерименту становить 36 хвилин, оскільки ми збільшуємо кількість проектів до десяти. Але враховуючи NP-складну природу проблеми та великий набір даних із 50 тис. вузлів, очікується, що час виконання буде збільшуватися зі збільшенням кількості проектів. У Таблиці ми призначаємо ідентифікатори експертів кожному необхідному навику в даній групі проектів разом із значенням їх прибутку.

```
--------------------Potential Experts----------------------
Expert ID | Allocated SKill | Capacity | Salary
1 | graph | 1 | 359
451 | computing | 4 | 1355
1381 | cloud | 4 | 6052
5898 | algebra | 4 | 9335
17167 | design | 3 | 552
1456 | map | 3 | 6907
272 | analytics | 4 | 1883
18 | database | 4 | 31
3346 | data | 3 | 22
7315 | mining | 4 | 155
7187 | testing | 4 | 8089
19 | computing | 2 | 1560
2900 | infrastructure | 2 | 1188
86 | dataoriented | 3 | 6608
151 | server | 3 | 4034
1497 | algorithm | 3 | 244
28505 | network | 3 | 901
123 | hardware | 2 | 7993
219 | evaluate | 4 | 5016
Expense: 62284
Budget: 500000
```

Рисунок 3.8 - Результат експерименту 4

Таблиця 3.10 - Проектні завдання для експерименту 4

Проекти	Необхідні навички	Призначений ідентифікатор експерта	Прибуток
Проект 1	сервер	151	500k
	апаратне забезпечення	123	
	оцінити	219	
	видобуток	18	
	обчислення	451	
Проект 2	сервер	151	250k

	мережі	28505	
	орієнтований на дані	86	
	апаратне забезпечення	123	
Проект 3	сервер	151	400k
	карта	1456	
	графік	1	
	видобуток	18	
	хмара	19	
	обчислення	451	
Проект 4	аналітика	272	275k
	бази даних	447	
	обчислення	451	
	тестування	7187	
Проект 5	алгебра	5898	200k
	обчислення	451	
	карта	1456	
Проект 6	хмара	7098	300k
	інфраструктури	2900	
	дизайн	17167	
	бази даних	447	
Проект 7	алгоритм	1497	350k
	тестування	7187	
	дизайн	17167	
Проект 8	хмара	7098	375k
	інфраструктури	2900	

	дизайн	17167	
	бази даних	447	
Проект 9	алгоритм	1497	180k
	тестування	7187	
	дизайн	17167	
Проект 10	хмара	7098	420k
	інфраструктури	2900	
	дизайн	17167	
	бази даних	447	

У експерименті 5 ми збільшуємо кількість проектів до чотирнадцяти, щоб побачити, як працює наш алгоритм для великої кількості проектів. Як показано в таблиці 3.11, ми маємо групу з чотирнадцяти проектів з бюджетом, призначеним для групи проектів, прибутком кожного проекту та набором необхідних навичок для кожного проекту.

Таблиця 3.11 - Вимоги до проекту для експерименту 5

Проекти	Необхідні навички	Profit	Budget
Проект 1	сервер, апаратне забезпечення, оцінка, майнінг, обчислення	500k	500k
Проект 2	сервер, мережа, орієнтований на дані, хмара	250k	
Проект 3	сервер, карта, графік, майнінг, хмара, обчислення	400k	
Проект 4	аналітика, база даних, обчислення, тестування	275k	
Проект 5	алгебра, обчислення, карта	200k	
Проект 6	хмара, інфраструктура, дизайн, база даних	300k	
Проект 7	алгоритм, тестування, дизайн	350k	
Проект 8	база даних, майнінг, хмара	375k	
Проект 9	алгоритм, обчислення, тестування, дані	180k	
Проект 10	хмара, база даних, мережа, алгоритм	420k	

Проект 11	обчислення, оптимізація, аналіз, упра вління	450k	
Проект 12	оптимізація, програмне забезпечення, тестування	390k	
Проект 13	oracle, hadoop, база даних, мережа	475k	
Проект 14	програмне забезпечення, програмування, оракул, інженерія	225k	

Після запуску алгоритму MWSCAN для експерименту 5 ми все ще можемо найняти групу експертів для даної групи з чотирнадцяти проектів у межах заданого бюджету та з мінімальними витратами на зв’язок. Як показано на рисунку 3.9, витрати на найм експертів становлять 114 827, що входить до призначеного бюджету в 500 000. Проте час виконання експерименту 5 становив 59 хвилин, що було значною мірою. Високий час виконання очікується через збільшення кількості проектів, а також з огляду на велику мережу експертів. У таблиці 3.12 ми показуємо ідентифікатори експертів, призначені кожному необхідному навику в заданому наборі проектів.

```
--------------------Potential Experts--------------------
Expert ID | Allocated SKill | Capacity | Salary
10561 | software | 2 | 26104
1 | graph | 1 | 359
451 | computing | 4 | 1355
3657 | management | 3 | 234
5898 | algebra | 4 | 9335
17167 | design | 3 | 552
272 | analytics | 4 | 1883
18 | database | 4 | 31
3346 | data | 3 | 22
7187 | testing | 4 | 8089
7315 | mining | 4 | 155
19 | computing | 2 | 1560
2900 | infrastructure | 2 | 1188
86 | dataoriented | 3 | 6608
151 | server | 3 | 4034
1497 | algorithm | 3 | 244
28505 | network | 3 | 901
603 | analysis | 1 | 239
219 | evaluate | 4 | 5016
1698 | oracle | 4 | 8067
35 | engineering | 2 | 9065
1381 | cloud | 4 | 6052
747 | hadoop | 1 | 2380
16302 | optimization | 2 | 867
37423 | programming | 4 | 3370
1456 | map | 3 | 6907
7098 | cloud | 3 | 1997
123 | hardware | 2 | 7993
447 | database | 4 | 220
Expense: 114827
Budget: 500000
```

Рисунок 3.9 - Результат експерименту 5

Таблиця 3.12 - Проектні завдання для експерименту 5

Проекти	Необхідні навички	Присвоєний ідентифікатор експерта	прибуток
Проекти 1	сервер	151	500k
	апаратне забезпечення	123	
	оцінити	219	
	видобуток	18	
	обчислення	451	
Проект 2	сервер	151	250k
	мережі	28505	
	орієнтований на дані	86	
	апаратне забезпечення	123	
Проект 3	сервер	151	400k
	карта	1456	
	графік	1	
	видобуток	18	
	хмара	19	
	обчислення	451	
Проект 4	аналітика	272	275k
	бази даних	447	
	обчислення	451	
	тестування	7187	
Проект 5	алгебра	5898	200k
	обчислення	451	
	карта	1456	
Проект 6	хмара: 7098	7098	300k
	інфраструктури	2900	
	дизайн	17167	
	бази даних	447	

Проект 7	алгоритм	1497	350k
	тестування	7187	
	дизайн	17167	
Проект 8	хмара	7098	375k
	інфраструктури	2900	
	дизайн	17167	
	бази даних	447	
Проект 9	алгоритм	1497	180k
	тестування	7187	
	дизайн	17167	
Проекти10	хмара	7098	420k
	інфраструктури	2900	
	дизайн	17167	
	бази даних	447	
Проект 11	обчислення	19	450k
	оптимізація	16302	
	аналіз	603	
	управління	3657	
Проект 12	оптимізація	16302	390k
	програмне забезпечення	10561	
	тестування	7187	
Проект 13	oracle	1698	475k
	hadoop	787	
	бази даних	18	
	мережі	28505	
Проект 14	програмне забезпечення	10561	225k
	програмування	37423	
	oracle	1698	
	інженерія	35	

Для всіх п’яти експериментів запропонований алгоритм успішно вирішив проблему найму кластерів у соціальних мережах. Результати були отримані з дотриманням усіх обмежень проблеми, як-от найм експертів у межах визначеного бюджету та їх працездатності, завершення всіх навичок даних проектів та мінімізація витрат на комунікацію між найнятою групою експертів. Тепер ми порівняємо наші результати з минулими підходами, які використовувалися для вирішення проблеми найму кластерів у соціальних мережах.

3.3 Порівняння пропонованого підходу з жадібним алгоритмом

Метою нашої дипломної роботи є вивчення та аналіз впливу алгоритму MWSCAN на розв’язання задачі Cluster Hire у соціальних мережах. Раніше для вирішення проблеми Cluster Hire в соціальних мережах пропонувалися жадібні алгоритми та підхід ILP. Хоча жадібні алгоритми дали швидші результати для задачі найму кластерів у соціальних мережах, це не було оптимальним або майже оптимальним рішенням проблеми. Оскільки жадібні алгоритми роблять локально оптимальний вибір на кожному етапі, їм не вдається створити глобальний оптимальний результат. Ми впровадили алгоритм MWSCAN для вирішення проблеми Cluster Hire у соціальних мережах. Ми провели різноманітні експерименти для алгоритму MWSCAN у великій мережі з 50 тисяч експертів і порівняли наші результати з жадібним алгоритмом, реалізованим Sagarika et al.

Кількість завершених проектів у порівнянні з бюджетом

Щоб виміряти ефективність алгоритму MWSCAN, ми порівнюємо кількість проектів, завершених у межах заданого бюджету, як для алгоритму MWSCAN, так і для жадібного алгоритму. Ми проводили експерименти для k кількості проектів, де $k = \{10,14\}$. У цьому налаштуванні база даних 50 тисяч

експертів і графік експертів залишаються незмінними, і ми просто змінюємо значення бюджету, призначеного для найму експертів. Потім ми аналізуємо кількість проектів, які алгоритми можуть виконати в межах заданого бюджету. Рисунок 20 представляє графік кількості завершених проектів та бюджету для 10 кількості проектів як для MWSCAN, так і для жадібного алгоритму. Як показано на графіку, алгоритм MWSCAN зміг виконати більшу кількість проектів за даний бюджет порівняно з жадібним алгоритмом. Це означає, що MWSCAN може виявитися більш економічно ефективним для організацій, оскільки вони можуть виконати більшу кількість проектів за обмежений бюджет, ніж рішення, надане жадібними алгоритмами.

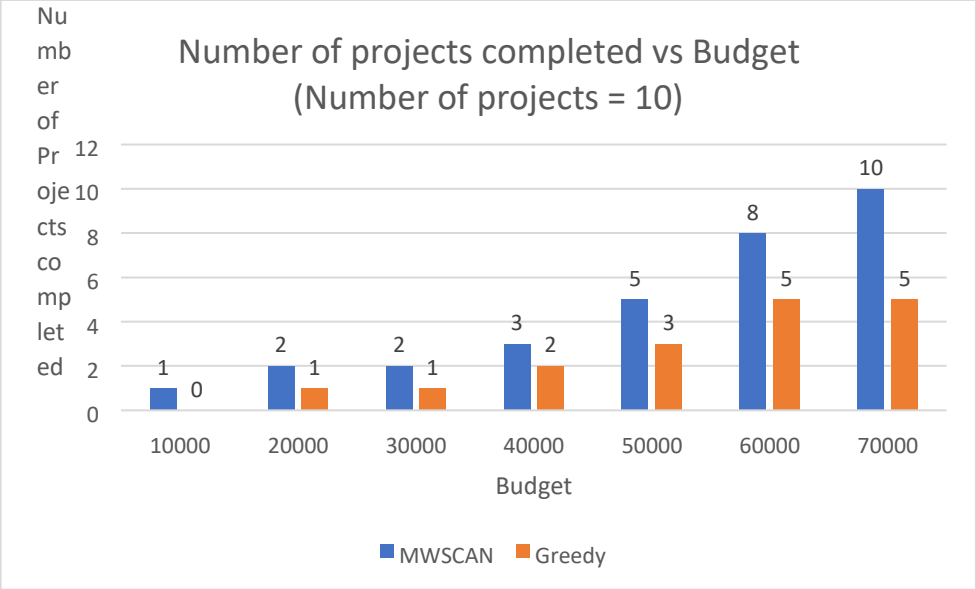


Рисунок 3.10 - Кількість завершених проектів у порівнянні з бюджетом (кількість проектів = 10)

Для другого випадку експерименту ми змінюємо кількість проектів у групі. Ми аналізуємо кількість завершених проектів для різних значень бюджету. На рисунку ми показуємо результати щодо кількості завершених проектів у порівнянні з бюджетом як для MWSCAN, так і для жадібного алгоритму. Знову ж таки, MWSCAN виконує більше проектів у порівнянні з жадібним алгоритмом для даного бюджету. Результати, отримані MWSCAN, є

економічно ефективними та вигідними для організацій. Оскільки MWSCAN завершує більше проектів у межах обмеженого бюджету, це приносить більше прибутку для компанії. У наступному розділі ми обговоримо порівняння прибутку між двома підходами.

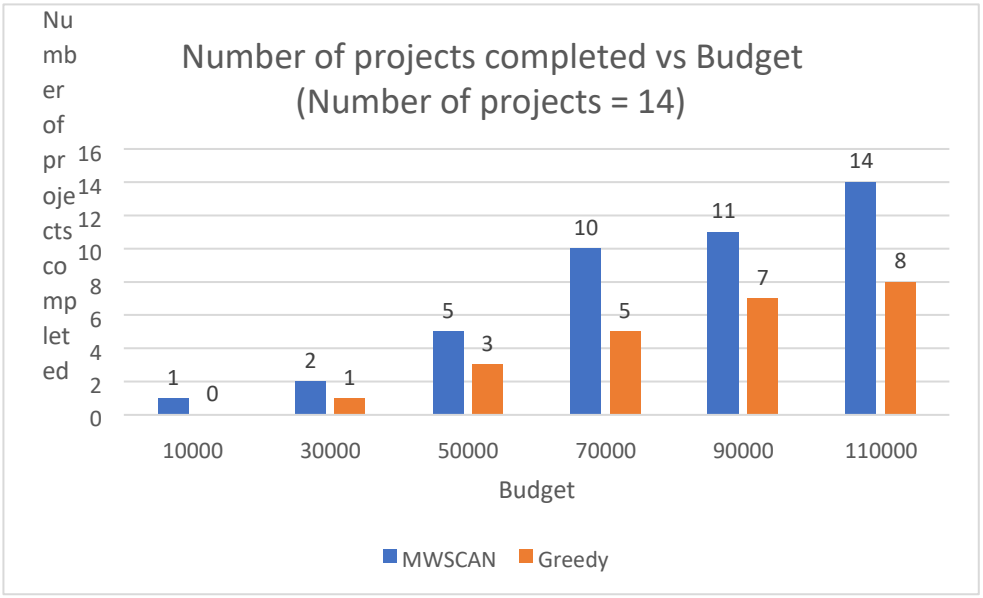


Рисунок 3.11 - Кількість завершених проектів у порівнянні з бюджетом (кількість проектів = 14)

Загальний прибуток проти бюджету

У цьому розділі ми порівнюємо прибуток, отриманий за допомогою алгоритму MWSCAN, із жадібним алгоритмом для різної вартості бюджету. Для цих експериментів ми залишаємо експертів і графік набору даних експертів однаковими. Змінюємо кількість проектів і проводимо експерименти для k кількості проектів, де $k = \{10,14\}$. На рисунку наведено графік співвідношення загального прибутку та бюджету для 10 проектів. На графіку показано порівняння між MWSCAN і жадібним алгоритмом. Як видно з графіка, загальний прибуток, отриманий за алгоритмом MWSCAN, вищий за жадібний алгоритм.

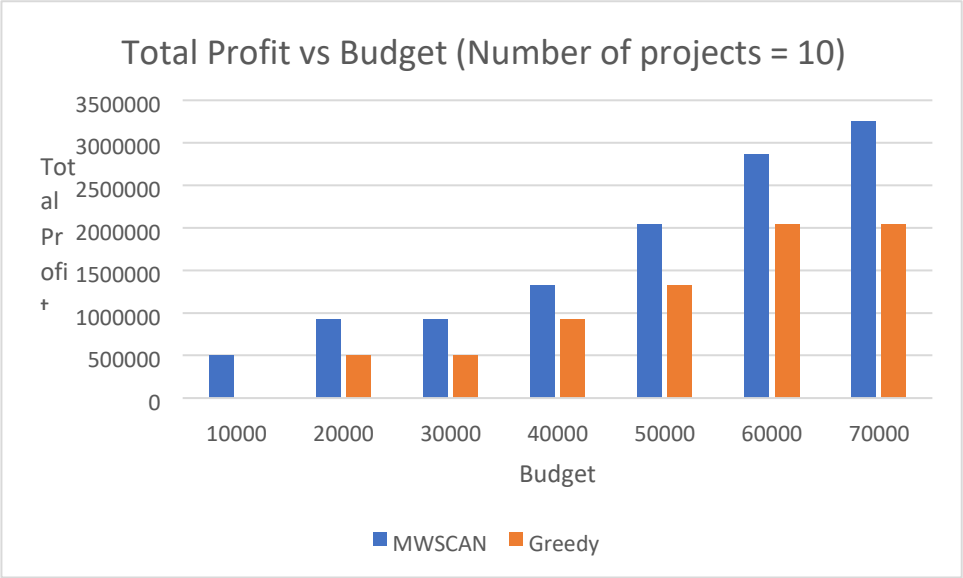


Рисунок 3.12 - Загальний прибуток проти бюджету (кількість проектів = 10)

Для другого випадку ми маємо групу з 14 проектів, і ми аналізуємо загальний прибуток і бюджет як для MWSCAN, так і для жадібного алгоритму. На малюнку нижче показано результати порівняння прибутку між двома алгоритмами. З малюнка видно, що алгоритм MWSCAN генерує більший загальний прибуток порівняно з жадібним алгоритмом для різних значень бюджету. З наших експериментів можна сказати, що алгоритм MWSCAN більш вигідний для організацій порівняно з жадібним алгоритмом.

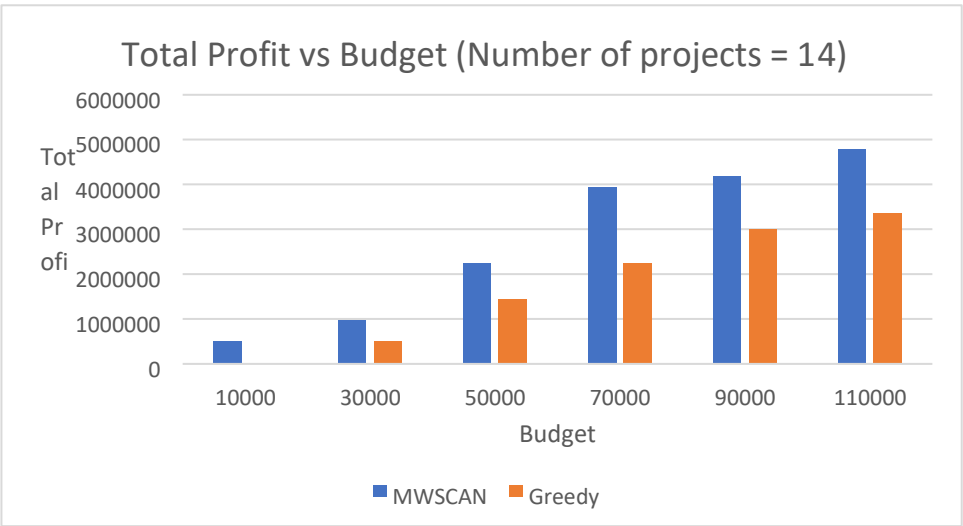


Рисунок 3.13 - Загальний прибуток проти бюджету (кількість проектів)

3.4 Порівняння з підходом цілочисельного лінійного програмування

Поряд із жадібним алгоритмом, підхід цілочисельного лінійного програмування (ILP) також був реалізований у минулому для вирішення проблеми наймання кластерів у соціальних мережах. Sagarika et. ін. запропонував підхід ILP, який зміг генерувати майже оптимальні рішення порівняно з жадібними евристичними рішеннями для проблеми найму кластерів у соціальних мережах. Недоліки підходу ILP полягають у тому, що йому потрібен величезний обсяг пам'яті для створення змінних і обмежень, а час виконання підходу також дуже високий для великих соціальних мереж. ILP проходить через усі можливі найкращі цільові функції; тому час виконання для ILP, як правило, дуже високий у порівнянні з жадібними рішеннями.

Ми запускаємо алгоритм ILP для невеликого набору даних, який складається з 50 експертів, 26 навичок і 5 проектів, який використовується в дослідницькій роботі. Ми реалізували підхід ILP в IDE Spyder (Python 3.8) за допомогою Anaconda Navigator. Наші експерименти проводилися на комп'ютері зі специфікаціями пристрою з процесором Intel Core i7-6700 @ 3,40 ГГц, на 64-розрядній операційній системі Windows з 8 ГБ оперативної пам'яті.

Через обчислювальну складність підходу ILP він не міг дати результатів у нашій системі. Ядро аварійно завершує роботу, оскільки запитує необґрунтований обсяг пам'яті та завершує роботу програми, як показано на малюнку. Програма ILP працює майже 4 години і все одно не може дати жодних результатів через обмеження пам'яті.

Як зазначено в дослідницькій роботі, мінімальна системна вимога становить 32 ГБ ОЗП для заданого розміру входу. Оскільки в нашій системі обмежена оперативна пам'ять, ми не змогли запустити підхід ILP та виконати порівняння з алгоритмом MWSCAN навіть для невеликого набору даних із 50 експертів.



Рисунок 3.14 - Результат ILP (50 експертів)

Але на тій же системі алгоритм MWSCAN зміг своєчасно генерувати майже оптимальне рішення проблеми Cluster Hire у соціальних мережах для великого набору даних із 50 тис. експертів. Оскільки алгоритм ILP не може генерувати результати, ми порівняли результати алгоритму MWSCAN з жадібним алгоритмом.

3.5 Результати використання алгоритму вирішення проблеми найму кластерів у соціальних мережах

У цьому розділі ми обговоримо ефективність запропонованого нами алгоритму вирішення проблеми найму кластерів у соціальних мережах з метою відбору групи експертів у межах заданого бюджету та працездатності з мінімальними витратами на комунікацію між групою експертів. Ми запропонували алгоритм MWSCAN для вирішення проблеми Cluster Hire у соціальних мережах. Ми провели різні експерименти, щоб оцінити ефективність нашого підходу. Ми проводимо експерименти з п’ятьма наборами проектів розмірів 3, 5, 7, 10 та, 14. Результати показують, що вирішення проблеми найму кластерів у соціальних мережах за допомогою алгоритму MWSCAN дає надійні та ефективні результати. Він задовольняє всі вимоги щодо бюджету, вартості зв’язку та навичок для вирішення проблеми.

Щоб оцінити ефективність нашого підходу, ми порівняли наші результати з минулим жадібним підходом. Ми провели порівняння на основі кількості завершених проектів і бюджету для двох наборів проектів розміром 10 та 14. Також ми провели порівняння на основі загального отриманого прибутку та бюджету як для алгоритму MWSCAN, так і для жадібного алгоритму. На основі результатів порівнянь MWSCAN видає більш ефективні результати в порівнянні з жадібним алгоритмом. Перш за все, алгоритм MWSCAN завершив більшу кількість проектів у даному бюджеті, ніж жадібний алгоритм. Крім того, MWSCAN приніс більше прибутку для даного бюджету в порівнянні з жадібним алгоритмом. З результатів порівняння стає зрозуміло, що організації отримають більше прибутку, якщо використовуватимуть алгоритм MWSCAN для вирішення проблеми найму кластерів у соціальних мережах, а не використовують попередні жадібні алгоритми. Тому можна сказати, що MWSCAN виробляє більш економне та вигідне рішення проблеми Cluster Hire у соціальних мережах порівняно з жадібним алгоритмом.

Ми також провели експерименти, щоб порівняти час виконання обох підходів для різної кількості проектів. Ми змінюємо кількість навичок, необхідних для кожного проекту, від 3 до 7, і вимірюємо час виконання, щоб отримати середній час виконання для заданої кількості проектів. На малюнку показано середнє порівняння часу виконання між запропонованим нами підходом MWSCAN та попереднім жадібним алгоритмом. Як видно на малюнку, алгоритм MWSCAN генерує результати майже за той самий час, що і жадібний алгоритм для трьох проектів у групі. Але в міру збільшення кількості проектів у наших експериментах середня тривалість роботи алгоритму MWSCAN значно збільшується в порівнянні з жадібним алгоритмом. Це можна розглядати як одне з обмежень алгоритму MWSCAN у порівнянні з жадібним алгоритмом. Але, оскільки ми проводимо експерименти з великим набором даних із 50 тис. вузлів, зрозуміло, що час виконання трохи вище. Крім того, алгоритм MWSCAN здійснює пошук у всій соціальній мережі експертів, щоб

знайти майже оптимальне рішення проблеми Cluster Hire в соціальних мережах. Для порівняння, жадібні алгоритми не генерують майже оптимальне рішення і роблять локально оптимальний вибір на кожному кроці.

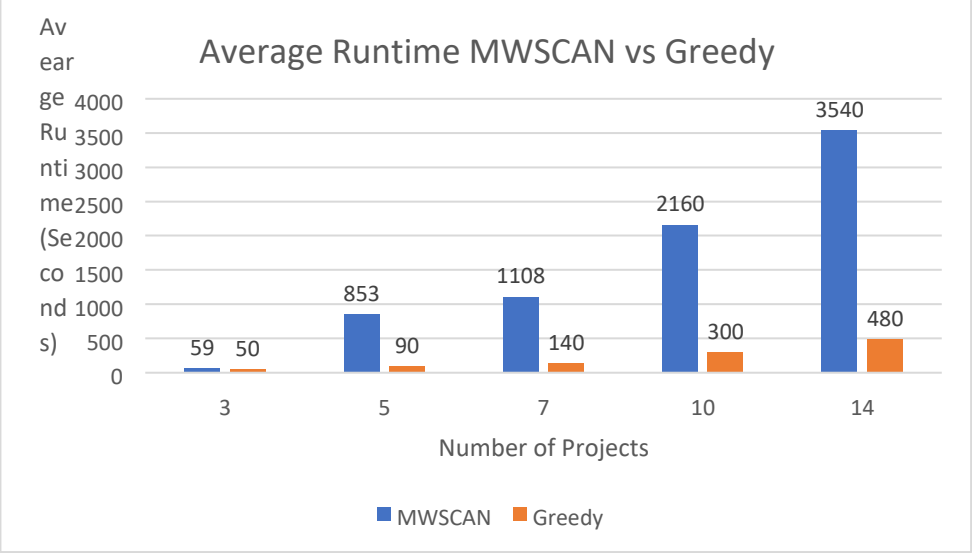


Рисунок 3.15 - Середня тривалість виконання проти кількості проектів

Наша дипломна робота була спрямована на вивчення та аналіз впливу алгоритму MWSCAN на вирішення проблеми Cluster Hire у соціальних мережах. Проблема Cluster Hire у соціальних мережах носить NP-складний характер; важко знайти оптимальне/близько до оптимального рішення через усі обмеження та цілі. Жадібні рішення, які використовувалися в минулому, не дали майже оптимальних результатів і не дали реального вирішення проблеми. З результатів порівняння можна сказати, що алгоритм MWSCAN міг завершити більшу кількість проектів і отримати більший прибуток за дану величину бюджету в порівнянні з жадібним алгоритмом. Таким чином, можна зробити висновок, що алгоритм MWSCAN створив ефективне, вигідне та майже оптимальне рішення проблеми Cluster Hire у соціальних мережах за розумні терміни.

ВИСНОВКИ

В дипломній роботі розглянуто кластерні моделі в соціальних мережах та соціальному програмному забезпеченні, а також запропоновано алгоритм для вирішення проблеми наймання кластерів у соціальних мережах. Мета роботи полягає в тому, щоб найняти групу експертів у межах заданого бюджету та працездатності, мінімізуючи витрати на комунікацію між найнятою групою експертів. Найм групи експертів у межах заданого бюджету та працездатності експертів при мінімізації витрат на комунікацію перетворює проблему в NP-складну проблему. Таким чином вирішили проблему найму кластерів у соціальних мережах за допомогою алгоритму кластеризації. Наші експерименти з набором даних показують, що алгоритм може ефективно вибрати групу експертів з мінімальними витратами на зв'язок для заданого набору проектів у межах заданих обмежень.

Крім того, наше дослідження походить від більш практичного та реалістичного сценарію найму, який можна було б застосувати майже до всіх типів проектів. Це підкреслює економію часу та грошей під час найму спільної групи експертів, що допомагає працювати ефективно та результативно. Ми розглядаємо можливість найняти групу експертів з різними навичками для різних проектів, щоб вони могли виконувати кілька проектів для однієї організації, враховуючи, що вони мають необхідні навички та працездатність. Це допоможе фірмам, що розробляють програмне забезпечення заощадити на витратах на найм для комплексних проектів.

СПИСОК ПОСИЛАНЬ НА ДЖЕРЕЛА

1. Bailey, Stefanie, and Peter Marsden (1999) ‘Interpretation and interview context’, Social Networks,21: 287-309.
2. Barabasi, Albert-Laszlo and Albert, Reka (1999) ‘Emergence of scaling in random networks’, Science, 286(5439):509-512.
3. Barabasi, Albert-Laszlo (2002) Linked. NY: Perseus.
4. Barnes, J.A. (1954) ‘Class and committees in a Norwegian island parish’, Human Relations, 7:39-58.
5. Barnes, J.A. (1972) Social Networks. Reading, MA: Addison-Wesley.
6. Batagelj Vladimir and Mrvar Andrej (2007) Pajek: Package for Large Network Analysis. University of Ljubljana, Slovenia. http://vlado.fmf.uni-lj.si/pub/networks/pajek/
7. Bearman, Peter, Moody, James and Stovel, Katherine (2004) ‘Chains of affection.’ American Journal of Sociology,110:44-91.
8. Bearman, Peter S. and Parigi, Paolo (2004) ‘Cloning headless frogs and other important matters: Conversation topics and network structure’, Social Forces, 83(2):535-557.
9. Bernard, H. Russell and Killworth, Peter (1977) ‘Informant accuracy in social network data II’, Human Communication Research, 4:3-18.
10. Bian, Yanjie (1997) ‘Bringing strong ties back in’ American Sociological Review, 62(3):366- 385.
11. Blau, Peter M. (1994) Structural Context of Opportunities. Chicago: University of Chicago Press.
12. Boase, Jeffrey and Wellman, Barry (2006) ‘Personal relationships: On and off the Internet’, in Anita Vangelisti and Dan Perlman (eds.), CambridgeHandbook of Personal Relationships.Cambridge: Cambridge University Press. pp. 709-723.
13. Boorman, Scott, and White, Harrison (1976) ‘Social structure from multiple networks II: Role structures’, American Journal of Sociology, 81:1384-1446.

14. Borgatti, Stephen, Everett, Martin and Freeman, Linton (2002) UCINET 6.0 for Windows, , MA: Analytic Technologies. http://www.analytictech.com/
15. Borgatti, Stephen, Mehra, Ajay, Brass, Daniel and Labianca, Giuseppe (2009) 'Network analysis in the social sciences', Science, 323(5916):892–895.
16. Bott, Elizabeth (1957) Family and Social Network. London: Tavistock.
17. Breiger, Ronald (1974) 'The duality of persons and groups', Social Forces,53:181-190.
18. Brewer, Devon D. (2000) 'Forgetting in the recall-based elicitation of personal and social networks', Social Networks,22:29-43.
19. Brewer, Devon and Webster, Cynthia. (1999) 'Forgetting of friends and its effects on measuring friendship networks', Social Networks, 21:361-373.
20. Buchanan, Mark (2002) Nexus: Small Worlds and the Groundbreaking Theory of Networks. New York: Norton.
21. Burt, Ronald. (2004) 'Structural holes and good ideas', American Journal of Sociology,110(2):349–399.
22. Burt, Ronald (1998) 'The gender of social capital', Rationality and Society,10:5-46.
23. Burt, Ronald (1984) 'Network items and the General Social Survey', Social Networks, :293-339.
24. Burt, Ronald S. (1992) StructuralHoles. Cambridge, MA: Harvard University Press.
25. Burt, Ronald (1997) 'The contingent value of social capital', Administrative Science Quarterly,42:339-65.
26. Burt, Ronald (2005) Brokerage and Closure: An Introduction to Social Capital. Oxford: Oxford University Press.
27. Butts, Carter (2008) 'Network: A package for managing relational data in R', Journal of Statistical Software, 24(2).
28. Carley, Kathleen M. (2006) 'A dynamic network approach to the assessment of terrorist groups and the impact of alternative courses of action', In Visualising

Network Information(pp. KN1-1 – KN1-10). Meeting Proceedings RTO-MP-IST-063, Keynote 1. Neuilly-sur-Seine, France. http://www.rto.nato.int/abstracts.asp

29. Casciaro, Tiziana, Carley, Kathleen M. and Krackhardt, David (1999) 'Positive affectivity and accuracy in social network perception', Motivation and Emotion, 23:285-306.
30. Chang, Mariko Lin (2005) 'With a little help from my friends (and my financial planner)', Social Forces, 83(4):1469-97.
31. Chen, Katherine K. (1999) 'The networks methodology of proxy-reporting: How well do respondents proxy-report different kinds of information on others, and how do respondents err in making these proxy-reports?', Qualifying paper, Department of Sociology, Harvard University, Cambridge, MA.
32. Cohen, Reuven, Erez, Keren, ben-Avraham, Daniel and Havlin, Shlomo (2000) 'Resilience of the Internet to random breakdowns', Physical Review of Letters,85:4626-28.
33. Cohen, Reuven, Erez, Keren, ben-Avraham, Daniel and Havlin, Shlomo (2001) 'Breakdown of the Internet under intentional attack', Physical Review of Letters,86:3682-85.
34. Cohen, Sheldon, Brissette, Ian, Skoner, David P., and Doyle, William J. (2001) 'Social integration and health: The case of the common cold', Journal of Social Structure, http://www.library.cmu.edu:7850/JoSS/cohen/cohen.html
35. Cohen, Sheldon, Doyle, William J., Skoner, David P., Rabin, Bruce S. and Gwaltney, Jack M. (1997) 'Social ties and susceptibility to the common cold', Journal of the American Medical Association,277:1940-44.
36. Coleman, James S. (1988) 'Social capital in the creation of human capital', American Journal of Sociology,94:S95-S120.

37.Coser, Rose Laub (1975) ‘The complexity of roles as a seedbed of individual autonomy’, The Idea of Social Structure: Papers in Honor of Robert K. Merton, New York: Harcourt Brace Jovanovich.

38.Campbell, Karen and Lee, Barret (1992) ‘Sources of personal neighbor networks: social

39.integration, need, or time?’, Social Forces, 70(4):1077-1100.

40.Domínguez, Silvia and Maya-Jariego, I. (2008) ‘Acculturation of host individuals: immigrants and personal networks’, American Journal Community Psychology, 42:309-27.

41.Emirbayer, Mustafa (1997) ‘Manifesto for a relational sociology’, American Journal of Sociology, 103(2):281-317.

42.Erickson, Bonnie (1996) ‘Culture, class and connections’, American Journal of Sociology,102:217-251.

43.Feld, Scott (1981) ‘The focused organization of social ties”, American Journal of Sociology, 86:1015-35.

44.Fernandez, Roberto M. and Harris, David (1992) ‘Social isolation and the underclass’, in Adele Harrell and George Peterson (eds.), Drugs, Crime, and Social Isolation. Washington DC: The Urban Institute. pp. 257-293.

БІБЛІОГРАФІЧНА ДОВІДКА

Тема дипломної роботи бакалавра: " Кластерні моделі в соціальних мережах та соціальному програмному забезпеченні "

Обсяг пояснювальної записки: 83 аркуші

Дата закінчення дипломної роботи 13 червня 2021р.

Підпис студента _____