

Кандиба І.О.

Чорноморський національний університет імені Петра Могили

Фісун М.Т.

Чорноморський національний університет імені Петра Могили

Горбань Г.В.

Чорноморський національний університет імені Петра Могили

Антіпова К.О.

Чорноморський національний університет імені Петра Могили

ГЕНЕРАЦІЯ СЦЕНАРІЇВ ВСТУПНОЇ КАМПАНІЇ ЗАКЛАДУ ВИЩОЇ ОСВІТИ НА ОСНОВІ КОГНІТИВНОЇ КАРТИ ТА ПРЕДМЕТНО-ОРІЄНТОВАНОЇ МОВИ ПРОГРАМУВАННЯ

У статті досліджено методи передбачення, що базуються на експертному оцінюванні. Проаналізовано сучасні дослідження в галузі передбачення та прогнозування вступної кампанії закладу вищої освіти (ЗВО), методи підвищення ефективності окремих ресурсів вступної кампанії. Запропоновано алгоритм реалізації методів передбачення вступної кампанії ЗВО на основі сценарного аналізу. Проаналізовано методи генерації сценаріїв на основі моделі предметної галузі. Представлено реалізацію інформаційної системи для збору експертних оцінок шляхом застосування методів мозкового штурму та Делфі. Проведено дослідження математичних засобів моделювання, що підтримують можливість роботи з графовими структурами. Розроблено предметно-орієнтовану мову програмування, що реалізує функції засобів математичного моделювання на основі графової бази даних Neo4j. Наведено алгоритм розроблення цієї предметно-орієнтованої мови за допомогою автоматизованих генераторів синтаксичних аналізаторів. Засобами представленої інформаційної системи проведено експертне оцінювання ймовірних подій вступної кампанії. На основі оцінок експертів побудовано когнітивну карту вступної кампанії університету, де вершинами представлено ймовірні події, а ребрами відображено ймовірність виклику певної події іншою. Описано метод зберігання та обробки когнітивної карти у Neo4j за допомогою предметно-орієнтованої мови програмування. Застосовано метод генерації множини сценаріїв на основі когнітивної карти шляхом реалізації спеціалізованого оператора предметно-орієнтованої мови. Застосовано методологію дерева ймовірностей для визначення ймовірності кожного згенерованого сценарію. Розроблено інтерфейс перегляду згенерованих сценаріїв із використанням мови розмітки graphviz. Згенеровано множини сценаріїв для передбачення вступної кампанії ЗВО.

Ключові слова: передбачення, сценарний аналіз, когнітивна карта, графові БД, предметно-орієнтована мова програмування.

Постановка проблеми. Ефективне прийняття рішень у будь-якій предметній галузі потребує проведення ґрунтовних досліджень можливих шляхів розвитку цієї галузі. Існують два основні підходи до розв'язку задачі: застосування методів прогнозування та застосування передбачення [1].

Методи прогнозування ґрунтуються на аналізі ретроспективних даних кількісного характеру, що обмежує їх практичне застосування. Передбачення спрямоване на визначення подій якісно нового характеру [2] та їх характеристик, що, ураховуючи сучасні тенденції розвитку технологій і трансформації предметних галузей, робить методи передбачення більш актуальними.

Здебільшого передбачення являє собою застосування окремих взаємопов'язаних методів у певній послідовності. Ці методи сформовано за допомогою універсального підходу системного аналізу [2].

Ключову роль у передбаченні відіграє сценарний аналіз [3]. Сценарний аналіз – це інструмент, що знайшов застосування у багатьох різних галузях. Сутність сценарного аналізу полягає у формуванні або генерації послідовності подій. У разі передбачення сценарії допомагають визначити результат впливу певної послідовності подій [4].

Існує багато різних методів генерації сценаріїв, що поділені на дві основні групи [5]:

– якісні методи: метод мозкової атаки, метод Делфі, морфологічний аналіз та ін.;

– графові методи: дерево подій, дерево відмов, дерево ймовірностей та прогнозний граф.

Генерація сценаріїв для предметних галузей – складний процес, що вимагає на першому етапі застосування методів отримання якісного характеру, що входять до групи якісних методів генерації, а на другому – застосування складнішого математичного апарату – теорії графів та методів роботи з ними. Предметною сферою для дослідження та досягнення поставленої мети вибрано вступну кампанію університету. Вона залежить від багатьох чинників, у тому числі якісно нового характеру. Прикладом цього є введення карантину, що обмежує можливість проведення різних масових заходів: дня відкритих дверей, наукового пікніку, курсів підготовки до зовнішнього незалежного оцінювання та ін. Зважаючи на цей факт та динамічну зміну умов вступу до університету застосування передбачення для вступної кампанії є актуальним.

Аналіз останніх досліджень і публікацій. Ефективне проведення вступної кампанії закладу вищої освіти є актуальною проблемою, що розглядається в багатьох джерелах [6–8]. Основні напрями досліджень цих робіт: визначення шляхів проведення професійної орієнтації, підвищення ефективності інформування абітурієнтів, дослідження впливу регіонального складника на вступну кампанію.

У роботі [6] проведено дослідження та класифікацію профорієнтаційних робіт кафедри ЗВО, у тому числі формування у абітурієнтів розуміння особливостей кар'єри, що почнеться після закінчення навчання, але в роботі відсутні пропозиції щодо сценаріїв реалізації заходів вступної кампанії.

Важливим складником вступної кампанії закладу вищої освіти є інформаційні ресурси мережі Інтернет. У роботі [7] наведено дослідження зв'язку між частотою пошукових запитів цільової аудиторії та кількістю абітурієнтів закладів вищої освіти. Це дослідження допомагає визначити основні напрями роботи з інформаційними ресурсами, а саме зі сторінкою університету у мережі Інтернет. Підвищення ефективності впливу online-ресурсів ЗВО більш детально розглянуто у роботі [8]. Однак у роботах [7; 8] не пропонується алгоритму побудови сценаріїв для більш ефективного проведення профорієнтаційних робіт.

Способи генерації сценаріїв запропоновано у роботах [5; 9]. Так, у [5] запропоновано застосування методології системного аналізу для гене-

рації та аналізу сценаріїв. У роботі [9] пропонується спосіб опису предметної галузі на основі когнітивних карт. Також у [9] описується можливість генерації сценаріїв на основі когнітивних карт шляхом перебору дуг графу когнітивної карти. Виконаний аналіз показав, що питання передбачення результатів вступної кампанії потребують подальших досліджень.

Постановка завдання. Метою дослідження є розроблення інформаційної системи для генерації множини сценаріїв на основі когнітивної карти вступної кампанії закладу вищої освіти. Створення подібної системи дасть змогу ефективніше застосовувати ресурси університету для проведення вступної кампанії. Для досягнення цілі були поставлені такі завдання:

– дослідження методів якісного експертного оцінювання, що необхідно застосувати для визначення ймовірних дій і подій вступної кампанії та їх характеристик;

– розроблення засобів представлення графових моделей та операцій над ними для можливості обробки когнітивних карт;

– реалізація генерації множини сценаріїв на основі когнітивної карти.

Виклад основного матеріалу дослідження. Сам процес передбачення є чітко визначеною послідовністю взаємозв'язаних кроків. У роботі [2] запропоновано такі етапи передбачення, що спираються на методи якісного аналізу:

- 1) попереднє дослідження проблеми;
- 2) якісний аналіз проблеми;
- 3) генерація сценаріїв;
- 4) оцінка реалістичності сценаріїв.

Першим кроком генерації сценаріїв є визначення концептів когнітивної карти. Для реалізації цього кроку необхідне проведення опитування експертів, що може бути здійснено на основі кількох різних методів. Початок передбачення результатів вступної кампанії ЗВО вимагає визначення всіх можливих дій і подій та їх характеристики. Реалізацію цього кроку можливо здійснити за допомогою застосування методу мозкової атаки, що потребує від експертів висловлювання всіх ідей [10].

Визначивши всі ймовірні дії і події та їх характеристики, експертам необхідно оцінити визначені характеристики. Одним із найбільш розповсюджених методів експертного оцінювання є метод Делфі. Суть методу полягає у проведенні індивідуального анкетування висловлювань експертів, що дає змогу отримати їх оцінки без групового впливу шляхом досягнення консенсусу або стійкості результатів [11; 12].

Алгоритм реалізації передбачення можна представити у вигляді блок-схеми (рис. 1).

Реалізувати опитування експертів можливо шляхом розроблення web-застосунку. Для кожного методу необхідна реалізація окремого інтерфейсу.

Експертною групою виступали викладачі кафедри інженерії програмного забезпечення. Виділено низку подій, що впливають на вступну кампанію ЗВО: відгуки друзів, збільшення кількості вакансій ІТ-компаній, залучення до викладання фахівців із цих компаній, проведення дня відкритих дверей, проведення наукових пікніків у місті, розміщення реклами у соціальних мережах, проведення курсів підготовки до ЗНО, роздача друкованої продукції, зустрічі з викладачами у школах, зустрічі з викладачами у коледжах, залучення учнів шкіл до розважальних заходів у ЗВО, участь у вступній кампанії фахівців з ІТ-компаній, розміщення на телебаченні рекламних матеріалів ЗВО, проведення онлайн-зустрічей, участь викладачів у шкільних олімпіадах, розміщення на сайті ЗВО інформації про якість викладання (сертифікати акредитації, сертифікати міжнародних курсів та ін.), поліпшення умов навчання у ЗВО (проведення ремонту, закупівля нових меблів та обладнання), участь студентських гуртків самореалізації у вступній кампанії ЗВО, консультування у

приймальній комісії, збільшення або зменшення кількості абітурієнтів у поточному році, поява нового актуального напрямку на ринку ІТ (вплив на актуальність галузі у цілому), введення нових курсів, пов'язаних із вимогами ринку ІТ.

Отримані дані використовуються для побудови когнітивної моделі, що в літературі часто має назву когнітивної карти. Когнітивна карта являє собою узагальнену структуру знань у вигляді графічно формалізованого представлення зв'язків між концептами (фактами, подіями, властивостями та ін.) [14]. Відображення когнітивної карти можливо у вигляді графу, де вершини представлені концептами, а ребра відображають вплив концептів один на одного. Граф під час реалізації когнітивної карти може бути знаковим або зваженим, де вага ребра відображає якусь міру впливу у заданій шкалі [15].

Важливу роль відіграє саме структура, на якій будується когнітивна карта. Графи – це математична абстракція сукупності об'єктів двох типів: вершин та ребер. Граф визначається математичним записом як $G(V,S,F)$, де V – множина вершин (вузлів); S – множина ребер (дуг); F – функція відображення множини ребер на множину вершин [16]. Графи можуть бути використані у різних цілях: для відображення алгоритмів, моделювання процесів логістики, економіки, у комунікативних та транспортних задачах.

Існують різні програмні середовища, що підтримують можливість роботи з графовими структурами: Maple, Mathcad, Matlab та ін. Із перерахованих необхідно виділити Maple, що орієнтований на вирішення складних математичних задач теорії графів, візуалізацію даних та моделювання. Він має власну вбудовану мову програмування. Ця мова реалізує синтаксис стандартних математичних операцій: піднесення в степені, добування кореню, тригонометричні операції, розв'язання рівнянь й нерівностей, інтегральне числення і безпосередню роботу з графами.

Робота з графами у Maple здійснюється за допомогою модуля *networks*. Цей модуль містить опис таких операторів для роботи з графами [17]:

1) *addedge* – додавання ребер до графу;

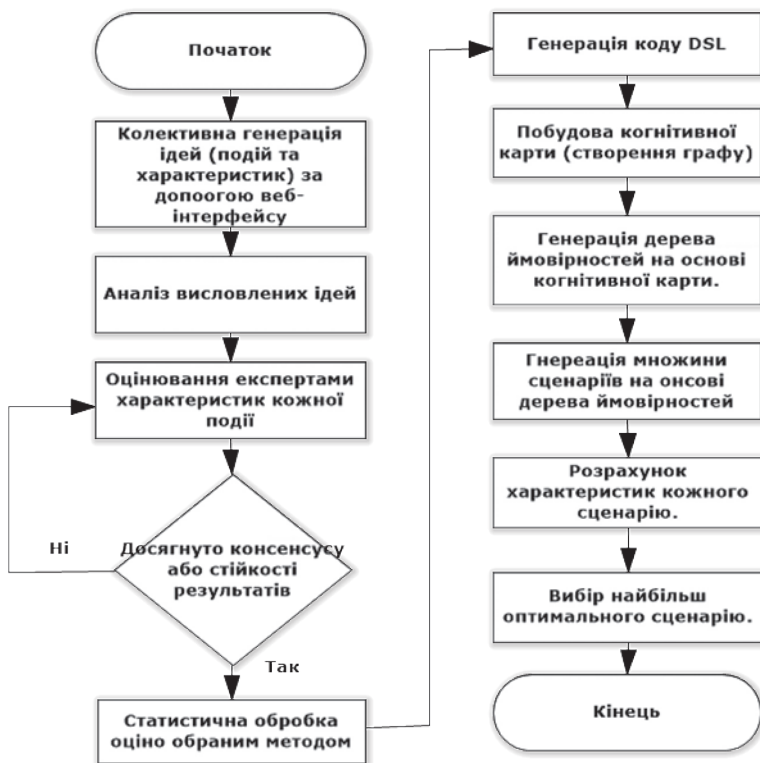


Рис. 1. Блок-схема алгоритму передбачення вступної кампанії

- 2) *addvertex* – додавання вершин до графу;
- 3) *adjacency* – побудова матриці суміжності;
- 4) *complete* – створення повного графу;
- 5) *delete* – видалення вершин та ребер із графу;
- 6) *draw* – побудова креслення графу;
- 7) *duplicate* – створення копії графу;
- 8) *ends* – повернення імен вершин;
- 9) *eweight* – знаходження ваги ребер;
- 10) *flow* – знаходження максимального потоку в мережі;
- 11) *head* – знаходження початкових вершин;
- 12) *incidence* – побудова матриці інцидентності;
- 13) *isplanar* – перевірка графа на планарність;
- 14) *new* – створення нового (порожнього) графу;
- 15) *petersen* – створення графу Петерсена;
- 16) *random* – створення випадкового графу;
- 17) *shortpathstree* – побудова дерева найкоротших шляхів;
- 18) *show* – відображення повної інформації про граф;
- 19) *tail* – відображення кінцевих вершин;
- 20) *void* – створення графу без ребер;
- 21) *vweight* – знаходження ваги вершин.

Процес передбачення, що виконується шляхом якісного аналізу системи, базується на сценарному аналізі [2]. Під сценарієм розуміють сукупність імовірних, взаємопов'язаних подій. Кожна подія має власний набір характеристик, що впливає на загальні характеристики сценарію.

Когнітивна карта є основою для представлення взаємозв'язків елементів сценарію та генерації множини сценаріїв [18]. Отже, під час реалізації передбачення на основі когнітивної моделі графа кожна вершина графа може мати власний набір характеристик. Програмні засоби, що реалізують математичну обробку графової моделі, не підтримують можливість зберігання набору даних у вершинах [17; 19; 20].

На ринку ІТ-продуктів уже пропонується низка систем керування базами даних (СКБД), що реалізовані на основі графової моделі. За даними порталу *db-engines.com*, лідером серед таких програмних продуктів станом на лютий 2021 р. є СКБД Neo4j. Ця СКБД повністю реалізує графову модель зберігання та обробки даних. Вона працює значно швидше із запитом до кількох пов'язаних різнотипних об'єктів, аніж реляційні СКБД, що дає змогу швидко отримати всі можливі шляхи на графі, котрі відображатимуть когнітивну модель. СКБД Neo4j має вбудовану мову маніпуляції даними Cypher. На жаль, ця мова не підтримує оператори математичної обробки графової моделі:

генерацію матриць суміжності та інцидентності, відображення початкових та кінцевих вершин, без підключення додаткового модулю не підтримується знаходження максимального потоку на графі та ін., але мова має всі необхідні оператори для маніпуляції даними, у тому числі обчислення, як і в реляційних БД, деяких агрегатних функцій: *count*, *avg*, *max*, *min*, *sum*, *stDev*. Окрім того, мова має спеціалізовані оператори для виконання операцій над графами: пошук найкоротшого шляху, визначення всіх шляхів та ін. [10].

Вузли у СКБД Neo4j є аналогією рядку таблиці у реляційних базах даних, де мітки виконують роль полів, а аналогом самої таблиці є тип вузла. Загального представлення таблиці не існує, що дає змогу встановлювати різні характеристики однотипним об'єктам.

Синтаксис мови Cypher сильно відрізняється від вбудованих мов у середовищах Maple, Mathcad, Matlab. Цей факт є підставою для створення предметно-орієнтованої мови роботи з графовою БД Neo4j, що використовуватиме оператори роботи з графами розповсюджених засобів математичного моделювання графів, наприклад Maple.

Предметно-орієнтованими мовами (ПОМ) називають мови програмування, що використовуються в певних предметних сферах або для спрощення роботи фахівців у певній галузі без навичок програмування [22]. До цього типу мов програмування можна віднести більшість мов моделювання (GPSS, Matlab, Maple та ін.), мови обробки даних (SQL, Cypher), мови для розроблення експертних систем (CLIPS, OWL) тощо.

Транслятори ПОМ включають у себе такі компоненти, як лексичний, синтаксичний та семантичний аналізатори. Розроблення перерахованих аналізаторів є відносно складним процесом, але його можна спростити шляхом використання спеціалізованих автоматичних генераторів лексичного та синтаксичного аналізаторів [23].

Реалізація ПОМ для когнітивного моделювання потребує використання операторів маніпуляції даними аналогічних операторам БД та операторів математичної обробки графів:

- 1) *add vertex* та *add edge* – для створення вершин та ребер у графі;
- 2) *delete* – для видалення вершин або ребер у графі;
- 3) *ends* – для перегляду списку наявних вершин та їхніх міток;
- 4) *head* – для перегляду початкових вершин;
- 5) *tail* – для відображення кінцевих вершин;
- 6) *adjacency* – перегляд матриці суміжності;

- 7) *incedence* – відображення матриці інцедентності;
- 8) *ShortPath* – пошук найкоротшого шляху;
- 9) *AllPath* – пошук усіх шляхів між двома вершинами;
- 10) *GetAllScenario* – генерація всіх можливих сценаріїв.

Представлені математичні операції з графами є аналогами операцій пакету Maple [17]. Застосування автоматичних генераторів аналізаторів потребує опису вхідної ПОМ за допомогою метамови форм Бекуса – Наура (БНФ) [24; 26]:

<Операція над графом> ::= <операція додавання> | <операція видалення> | <пошук найкоротшого шляху> | <пошук всіх шляхів> | <відображення всіх вершин> | <відображення початкових вершин> | <відображення кінцевих вершин> | <відображення матриці інцедентності> | <відображення матриці суміжності> | <генерація сценаріїв>;

<операція додавання> ::= **add** <додати сутність> <додати сутність> ::= <додати вершину> | <додати ребро>

<додати вершину> ::= **vertex** <назва> : <тип> (<перелік міток>)

<перелік міток> ::= <назва> : <значення> | <назва> : <значення> , <перелік міток>

<додати ребро> ::= **edge** <тип> (<назва> , <назва>) : <властивості ребра>

<властивості ребра> ::= <назва> : <значення> | <назва> : <значення> , <перелік міток>

<операція видалення> ::= **delete** <видалити сутність>; <видалити сутність> ::= <видалити вершину> | <видалити ребро>

<видалити ребро> ::= **edge** (<назва> , <назва>)

<видалити вершину> ::= **vertex** <параметри видалення вершини>

<параметри видалення вершини> ::= **ID** : <номер> | <назва> | <перелік міток>

<пошук найкоротшого шляху> ::= **ShortPath** (<назва> , <назва>)

<пошук всіх шляхів> ::= **AllPath** (<назва> , <назва>)

<генерація сценаріїв> ::= **GetAllScenario** (<назва> , <назва>) : <перелік операцій над мітками> | <перелік операцій над мітками> ::= <операція над міткою> | <операція над міткою> , <перелік операцій над полями>

<операція над міткою> ::= <операція> (<назва>)

<операція> ::= **summ** | **multiplication** | **probability**;

<відображення всіх вершин> ::= **Ends**

<відображення початкових вершин> ::= **Head**

<відображення кінцевих вершин> ::= **Tail**

<відображення матриці інцедентності> ::= **Incedence**

< відображення матриці суміжності> ::= **Adjacency**

<Назва> ::= [A-Z][A-Za-z0-9]+

<Тип> ::= [A-Z][A-Za-z]+

<Значення> ::= <цифра> | <рядок>

<цифра> ::= ^[-]?[0-9]*[.]?[0-9]+

<рядок> ::= « [^»]+ «;

<номер> ::= [0-9]+;

Наведена граматику містить оператор для виконання побудови сценаріїв, що виконує перебір усіх шляхів на графі аналогічно операції побудови сценаріїв на основі дерева ймовірностей. Цей підхід застосовується як один із методів ситуаційного управління [26]. При цьому підході всю когнітивну карту можна відобразити у вигляді дерева ймовірностей, тоді кожен сценарій утворюється шляхом перебору дуг (рис. 2).

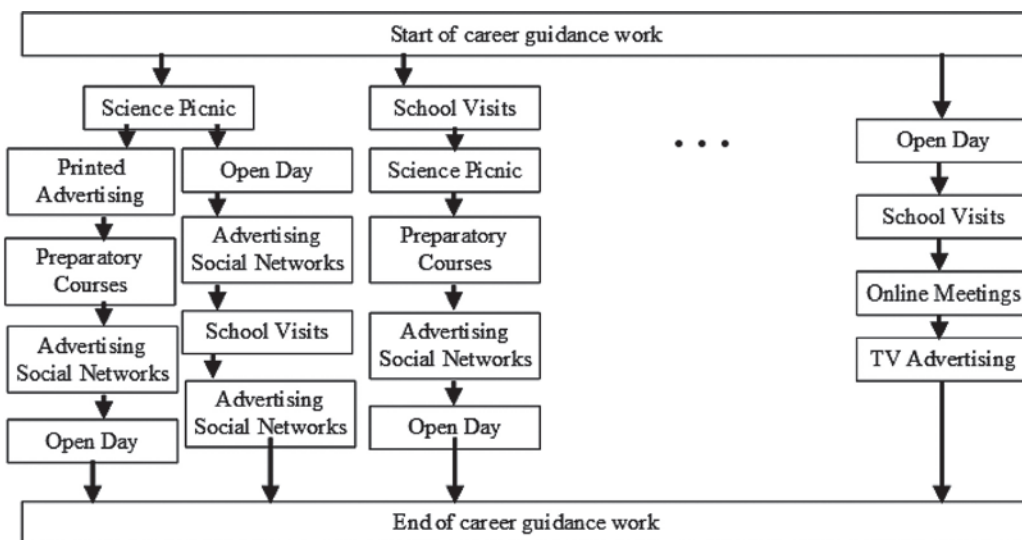


Рис. 2. Відображення процесу формування множини сценаріїв

Важливим складником генерації сценаріїв є розрахунок характеристик кожного сценарію. Цю функцію реалізовано за принципом агрегатних функцій: count, avg, max, min, sum.

За наявності формального опису ПОМ можливо розпочати процес автоматичної генерації синтаксичного аналізатора. Існує декілька доступних інструментів автоматичної генерації аналізаторів: Pyparsing, Parglare, PLY, ANTLR, Unicc.

Серед доступних інструментів варто виділити Unicc (Universal Compiler-Compiler). Він розроблений на мові C і підтримує можливість роботи з різними операційними системами та мовами програмування загального призначення. Під час розроблення ПОМ для графової БД параметрами можуть виступати рядки, що містять текст кирилицею. На відміну від аналогів Unicc підтримує можливість роботи з кирилицею. Згенерований синтаксичний аналізатор працює за методом LALR(1) (Lookahead Left to Right) [26].

Транслятор для вхідної ПОМ розроблено на мові загального призначення Python. Ця мова займає одну з лідируючих позицій у рейтингах вакансій на ринку праці, що сприяє її розвитку та вдосконаленню. Окрім того, Python підтримує декілька різних парадигм програмування та велику кількість додаткових інструментів, наприклад бібліотеки для різних СКБД, вебфреймворки (Flask, Django тощо), бібліотеки для роботи з апаратним забезпеченням та ін. Перелічені переваги роблять python привабливим інструментом для розроблення ПОМ.

Результатом генерації Unicc є код на мові Python, що включає у себе декілька класів (рис. 3). Функціонування транслятора ПОМ вимагає наявності семантичного аналізатора, що безпосередньо перетворюватиме вхідний код на код вихідною мовою. Вихідною мовою виступає Cypher, що виконує маніпуляцію з графовою БД.

Підключення до СКБД Neo4j виконується за допомогою бібліотеки ru2neo. Цей модуль доступний у репозиторіях менеджера керування пакунками pip. Виконання підключення здійснюється шляхом уведення даних користувача та адреси призначення до конструктора об'єкта Graph.

Створення вершини мовою Cypher має такий вигляд: `Create(PrintedAdvertising:event {Teachers_hours_according_to_the_plan:0,actual_hours:2,financial_expeses:500, increasing_percentage_entrants:2, average_score_entrants:175, vertexName: «PrintedAdvertising», Probability:0.8, title: «Роздача друкованої продукції» })`, цей же запит ПОМ виглядатиме так: `add vertex PrintedAdvertising:event`

(`Teachers_hours_according_to_the_plan:0, actual_hours:2, financial_expeses:500, increasing_percentage_entrants:1, average_score_entrants:175, Probability:0.8, title: «Роздача друкованої продукції»`); що можна вважати рівноцінними. Але під час створення ребер мова Cypher вимагає виконання запити вибірки даних із параметрами вершин, наприклад поєднати вершини з іменами `PrintedAdvertising` та `TVAdvertising` можливо, виконавши запит:

`MATCH (n),(m) WHERE n.NameVertex=«PrintedAdvertising» AND m.NameVertex=«TVAdvertising» CREATE (a)-[r: Call_Probability]{Probability:0.75}->(n) RETURN type(m)`

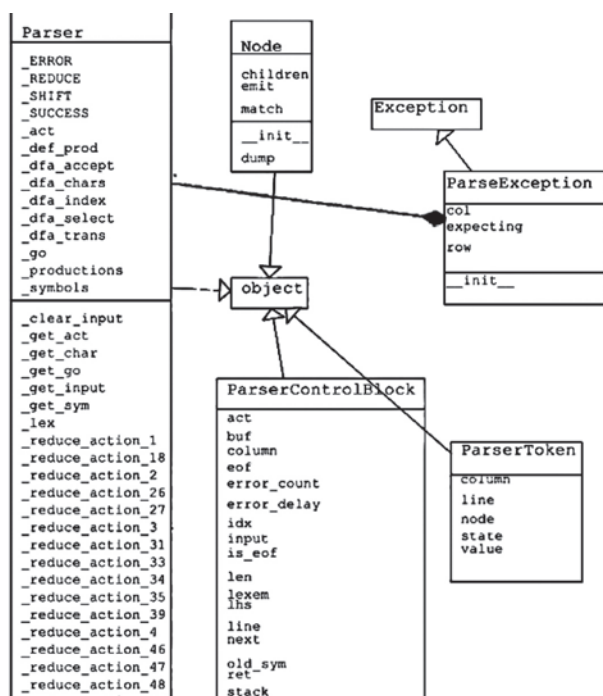


Рис. 3. Діаграма класів згенерованого аналізатора ПОМ

Аналогічний запит ПОМ виглядатиме так: `add edge Probability (PrintedAdvertising, TVAdvertising): Call_Probability:0.65`.

На базі ПОМ створено когнітивну карту вступної кампанії ЗВО. Фізично модель являє собою граф у БД Neo4j, де кожна вершина виступає ймовірною подією у вступній кампанії, а кожне ребро відображає ймовірність виклику однієї події іншою (рис. 4).

Обсяг коду запити ПОМ менше, ніж Cypher, що гарантує зменшення часу, необхідного на написання коду та зменшення ймовірності припуститися помилки під час створення запиту [19].

Інтерфейс для відображення згенерованих сценаріїв являє собою таблицю, у комірках якої зберігаються оцінки відносної ймовірності та назви

ймовірних подій. Кожен рядок цієї таблиці являє собою опис окремого сценарію разом із його характеристиками, наприклад відображення фактичної кількості годин викладачів, витрачених на реалізацію сценарію, та ймовірність сценарію (рис. 5).

Відображення окремого сценарію включно з усіма характеристиками ймовірних подій вступної кампанії, що входять до цього сценарію, вимагає використання додаткових засобів. Таким засобом є бібліотека graphviz. Вона підтримує реалізацію

однойменної мови розмітки для відображення графів. Використавши цю бібліотеку, можливо відобразити не лише загальну структуру графу, а й властивості його вершин (рис. 6).

Отже, у запропонованому варіанті передбачення враховано 22 ймовірні події та 85 взаємозв'язків між подіями, що можуть вплинути на проведення вступної кампанії ЗВО. Використавши генерацію на основі створеної когнітивної моделі оператором:



Рис. 4. Відображення графу засобами Neo4j

Scenario view			
Scenario			
	summ(actual_hours)	Probability	Event
19	150	0.0090	Участь викладачів ЗВО у шкільних олімпіадах
20	200	0.0090	Відвідування шкіл
21	138	0.0090	Розміщення реклами на телебаченні
22	24	0.0010	Рекомендації приймальної комісії
23	18	0.0900	Покращення умов навчання (ремонт, оновлення ма
24	44	0.0010	Участь викладачів ЗВО у шкільних олімпіадах
25	32	0.0090	Роздача друкованої продукції
26	54	0.0001	Відвідування технікумів
27	64	0.0001	Відвідування технікумів
28	94	0.0010	Вілвілвання шкіл

Export to excel

Рис. 5. Інтерфейс користувача для відображення множини згенерованих сценаріїв

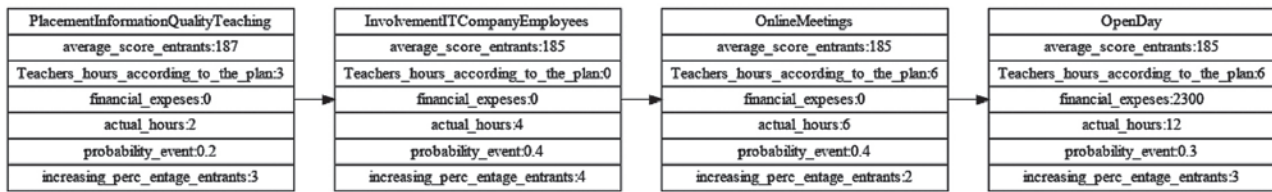


Рис. 6. Відображення вузлів із характеристиками

$GetAllScenario(Begin, End): \quad summ(actual_hours),$

де *Begin* – подія початку (Start of career guidance work), *End* – подія завершення сценарію (End of career guidance work). *Summ ()* функція, що сумує певний показник подій у сценарії, фактична кількість годин, що затрачені викладачами на певну подію, у результаті на основі когнітивної моделі отримуємо 2 315 можливих сценаріїв вступної кампанії.

Висновки. Проведено дослідження методів передбачення. Застосовано передбачення на основі сценарного аналізу для визначення всіх

імовірних шляхів проведення даного процесу. Розроблено інформаційну систему для реалізації методів експертних оцінок. Реалізовано предметно орієнтовану мову програмування для побудови когнітивної карти у середовищі Neo4j. Застосовано метод генерації сценаріїв на основі перебору дуг графу когнітивної карти. Згенеровано 2 315 сценаріїв вступної кампанії ЗВО за допомогою розробленої системи.

У подальшому планується інтеграція до розробленої системи підсистеми багатокритеріальної оптимізації, що дасть змогу спростити процес вибору на основі кількох критеріїв.

Список літератури:

1. Грабовецький Б.Є. Методи експертних оцінок: теорія, методологія, напрямки використання : монографія. Вінниця : ВНТУ, 2010. С. 171.
2. Згуровський М.З. Сценарний аналіз як системна методологія предвидення. *Системні дослідження та інформаційні технології*. 2002. № 1. С. 7–38.
3. Лебідь О.Ю. Деякі аспекти застосування когнітивного моделювання в державному управлінні. *Державне управління: удосконалення та розвиток*. 2015. № 11. URL: <http://www.dy.nayka.com.ua/?op=1&z=922>.
4. Ляліна Н.С., Бобро М.К. Сценарне прогнозування як інструмент антикризового управління в умовах економічної нестабільності. *Глобальні та національні проблеми економіки*. 2018. №. 23. С. 263–268.
5. Гожий А.П., Коваленко І.І. Системные технологии генерации и анализа сценариев. *Автоматика. Автоматизация. Електротехнічні комплекси та системи*. 2005. №. 2. С. 89–96.
6. Найдюш А.В., Лебедєв В.О., Спирінцев Д.В. Досвід організації профорієнтаційної роботи кафедри. *Вступна кампанія до закладів вищої освіти України: проблеми та перспективи* : зб. тез доп. міжнар. наук.-практ. конф., м. Київ, 13 травня 2019 р. Київ, 2019. С. 10–13
7. Гаріна С.М., Тверезовська Н.Т. Дослідження зв'язку між частотою пошукових запитів цільової аудиторії та кількістю абітурієнтів закладів вищої освіти. *Фізико-математична освіта*. 2018. №. 4. С. 31–36.
8. Гаріна С.М., Тверезовська Н.Т. Online складова в профорієнтаційній роботі закладу вищої освіти. *Гуманітарні студії: педагогіка, психологія, філософія*. 2019. Т. 2. № 10. С. 10–25.
9. Трахтенгерц Э.А. Компьютерная поддержка принятия решений. Москва : Синтез, 1998. 376 с.
10. Koziorowska A. Use of the brainstorm method in the innovation laboratory (i-lab). *Edukacja – Technika – Informatyka*. 2016.
11. Duin P. van der. Foresight in organizations: Methods and tools: *Foresight in Organizations: Methods and Tools*. 2016. 268 p.
12. Ab Latif R., Mohamed R., Dahlan, A. Using Delphi Technique: Making Sense of Consensus in Concept Mapping Structure and Multiple Choice Questions (MCQ). *Education in Medicine Journal*. 2016. Vol. 8. № 3. P. 89–98.
13. Zartha Sossa J.W., Halal W., Hernandez Zarta R. Delphi method: analysis of rounds, stakeholder and statistical indicators. *Foresight*. 2019. Vol. 21. № 5. P. 525–544.
14. Максимов В.И., Корноушенко Е.К., Качаев С.В. Когнитивные технологии для поддержки принятия управленческих решений *Информационное общество*. 1999. №. 2. С. 50–54.
15. Авдеева З.К., Коврига С.В., Макаренко Д.И. Когнитивное моделирование для решения задач управления слабоструктурированными системами (ситуациями). *Управление большими системами*. 2007. Т. 16. С. 26–39.

16. Батрак Ю.А. Элементы дискретной математики: пособие для самостоятельной работы. Николаев, 2011. 427 с.
17. Прохоров Г.В., Леденев М.А., Колбеев В.В. Пакет символьных вычислений Maple V. Москва : Петит, 1997. 200 с.
18. Андрейчиков А.В., Андрейчикова О.Н. Интеллектуальные информационные системы : учебник для вузов. Москва : Финансы и статистика, 2004. 464 с.
19. Кирьянов Д.В. Mathcad 15/Mathcad Prime 1.0. Санкт-Петербург : БХВ-Петербург, 2012. 432 с.
20. Документація Matlab: graph. URL: <https://uk.mathworks.com/help/matlab/ref/graph.html> (дата звернення: 26.02.2021).
21. Robinson, I., Webber, J., Eifrem, E. Graph Databases. New opportunities for connected data: Joe Celko's Complete Guide to NoSQL. Sebastopol, Californ: O'Reilly Media, 2015. 238 p.
22. Fowler M. Domain-Specific Languages. Boston: Addison-Wesley Professional, 2010. 640 p.
23. Фісун М.Т., Кандиба І.О. Використання програмної системи ANTLR для створення мови реляційної алгебри: *Одинадцята міжнародна науково-практична конференція ІОИ-2018* : зб. тез доп. наук.-практ. конф., м. Вінниця 22–25 травня 2018 р. Вінниця : ВНТУ, 2018. С. 287–288.
24. Aho A., Lam M., Sethi R. Compilers: Principles, Techniques, and Tools: Boston: Addison Wesley, 2006. 1040 p.
25. Meyer, J.M. UniCC: A universal LALR(1) Parser Generator. Dortmund: Phorward Software Technologies, 2018. 163 p.
26. Коваленко І.І., Швед А.В., Антипова К.О. Моделі подання та виведення знань у системах ситуаційного управління. Миколаїв : Іліон, 2018. 92 с.

Kandyba I.O., Fisun M.T., Horban H.V., Antipova K.O. GENERATION OF SCENARIOS FOR THE INTRODUCTORY CAMPAIGN OF A HIGHER EDUCATION INSTITUTION BASED ON A COGNITIVE MAP AND A DOMAIN-SPECIFIC LANGUAGE

In the article foresight methods based on expert evaluation are investigated. Modern research in the field of forecasting and prognostication the admission campaign of a higher education institution (IHE), methods of improving the efficiency of the use of certain resources during the admission campaign are analyzed. An algorithm for the implementation of methods for predicting the introductory campaign of the IHE on the basis of scenario analysis is proposed. Methods of scenario generation based on the subject area model are analyzed. The implementation of an information system for collecting expert assessments by applying the methods of brainstorming and Delphi are presented. A study of mathematical modeling tools that support the ability to work with graph structures is produced. A subject-oriented (domain-specific) programming language (DSL) has been developed that implements the functions of mathematical modeling tools based on the graph's database Neo4j. An algorithm for the development of this DSL using automated parser generators is presented. An expert assessment of the probable events of the introductory campaign was conducted by means of the presented information system. Based on the experts' assessments, a cognitive map of the university's admission campaign is constructed, where the vertices represent probable events, and the edges reflect the probability of calling one certain event to another. Describes Methods of storing and processing a cognitive map in Neo4j with using DSL are described. The method of generating a set of scenarios based on a cognitive map by implementing a specialized operator of subject-oriented language (DSL) is applied. The probability tree methodology is used to determine the probability of each generated scenario. An interface for viewing generated scenario using graphviz markup language has been developed. A set of scenarios was generated to predict the introductory campaign of the IHE.

Key words: foresight, scenario analysis, cognitive map, graph database, domain-specific language.