

ІГРОВИЙ МЕТОД КЛАСТЕРИЗАЦІЇ ОНТОЛОГІЙ

П. О. Кравець, Є. В. Буров, В. В. Литвин

Національний університет “Львівська політехніка”, кафедра інформаційних систем та мереж,
Україна, м. Львів, вул. С. Бандери 12

E-mail: Petro.O.Kravets@lpnu.ua, ORCID: 0000-0001-8569-423X;

E-mail: Yevhen.V.Burov@lpnu.ua, ORCID: 0000-0001-8653-1520;

E-mail: Vasyl.V.Lytvyn@lpnu.ua, ORCID: 0000-0002-9676-0180

© *Кравець П. О., Буров Є. В., Литвин В. В., 2019*

Розглянуто актуальну проблему кластеризації онтологій для оптимізації операцій інтелектуального опрацювання даних в умовах невизначеності, зумовленої неточністю або неповнотою даних про предметну область. Кластеризація онтологій – це процес автоматичного розділення множини онтологій на групи (кластери) на основі ступеня їхньої подібності. Для розв'язування задачі кластеризації необхідно задати міри близькості онтологій, вибрати або розробити алгоритм кластеризації та виконати змістовну інтерпретацію результатів кластеризації.

Для кластеризації онтологій в умовах невизначеності запропоновано застосувати стохастичний ігровий метод. Повторювальна стохастична гра полягає у реалізації керованого випадкового процесу вибору кластерів онтологій. Для цього закріплені за онтологіями інтелектуальні агенти випадково, одночасно і незалежно вибирають один із кластерів у дискретні моменти часу. Для агентів, що обрали один кластер, обчислюють поточну міру подібності онтологій, яка може враховувати близькість концептів, атрибутів та відношень між концептами. Цю міру використовують для адаптивного перерахунку змішаних стратегій гравців. Збільшуються імовірності вибору тих кластерів, поточний склад яких призвів до зростання міри подібності онтологій. У ході повторювальної гри агенти сформуєть вектори змішаних стратегій, які забезпечать максимізацію усереднених мір подібності розділених на кластери онтологій.

Для розв'язування задачі ігрової кластеризації онтологій розроблено адаптивний марківський рекурентний метод на основі стохастичної апроксимації модифікованої умови доповняльної нежорсткості, справедливої у точках рівноваги за Нешем. Запропонований ігровий метод має фільтрувальні властивості щодо викидів у вхідних даних і практично не залежить від закону розподілу випадкових завад.

Комп'ютерне моделювання підтвердило можливість застосування моделі стохастичної гри для кластеризації онтологій із врахуванням факторів невизначеності. Збіжність ігрового методу забезпечується дотриманням фундаментальних умов та обмежень стохастичної оптимізації. Достовірність експериментальних досліджень підтверджується повторюваністю отриманих результатів для різних послідовностей випадкових величин.

Результати роботи доцільно використати для розв'язування задач інтелектуального аналізу даних, усунення дублювання інформації в базах знань, зменшення невизначеності у межах кластера онтологій, виявлення новизни інформації, організації високорівневої семантичної взаємодії між агентами під час розв'язування ними спільної задачі.

Ключові слова: стохастична гра, умови невизначеності, кластеризація, онтологія, база знань, інтелектуальний агент.

GAME METHOD OF ONTOLOGY CLUSTERING

P. O. Kravets, E. V. Burov, V. V. Lytvyn

Lviv Polytechnic National University, Information Systems and Networks Department,
12, S. Bandery Str., Lviv, Ukraine

E-mail: Petro.O.Kravets@lpnu.ua, ORCID: 0000-0001-8569-423X;

E-mail: Yevhen.V.Burov@lpnu.ua, ORCID: 0000-0001-8653-1520;

E-mail: Vasyl.V.Lytvyn@lpnu.ua, ORCID: 0000-0002-9676-0180

In this paper the important problem of ontology clustering is considered with the purpose of optimization of intelligent data processing in conditions of uncertainty caused by inaccuracy or incompleteness of data in the subject area. The clustering of ontologies is the process of automatic splitting of a set of ontologies into groups (clusters) based on their similarity degree. For the resolution of this problem it is necessary to adopt the set of measures for the affinity of ontologies, to choose or develop an algorithm of clusterization and to execute the thorough interpretation of clusterization results.

For the clustering of ontologies in conditions of uncertainty, it is proposed to use a stochastic game method. A repetitive stochastic game consists in the implementation of a controlled random process for selecting clusters of ontologies. To this effect, the intelligent agents, assigned to ontology, randomly, simultaneously and independently choose one of the clusters at discrete moments of time. For agents that have selected a cluster, the current measure of similarity of ontologies is calculated, which takes into account the proximity of concepts, attributes, and relationships between concepts. This measure is used to adapt the recalculation of mixed player strategies. Thus, the probability of selection is increased for clusters having the composition, which led to the growth of the ontologies similarity degree. During the repetitive game, agents will form vectors of mixed strategies that will maximize the averaged measures of similarity to clusters of ontologies.

To solve the problem of game clusterization for ontologies, an adaptive Markovian recurrent method was developed based on stochastic approximation of a modified complementary slackness condition, valid at the points of the Nash equilibrium. The proposed game method has filtering properties for spikes in the input data and practically does not depend on the law of distribution of random noises.

The computer modeling confirmed the possibility of using a stochastic game model for clustering ontologies, taking into account uncertainty factors. Convergence of the game method is ensured by observing the fundamental conditions and restrictions of stochastic optimization. The reliability of experimental studies is confirmed by the repeatability of results obtained for various sequences of random variables.

The results of the work could be used to solve the problems of intellectual data analysis, to eliminate duplication of information in knowledge bases, to reduce uncertainty within the cluster of ontologies, to identify the novelty of information, to organize high-level semantic interaction between agents in the course of executing their common task.

Key words: stochastic game, conditions of uncertainty, clusterization, ontology, knowledge base, intellectual agent.

Вступ

Складність сучасних інформаційно-пошукових систем та стрімкий розвиток телекомунікаційних технологій спонукає до застосування автоматизованих засобів опрацювання інформації за допомогою інтелектуальних агентів [1]. Під час розв'язування задачі в інформаційному середовищі агенти взаємодіють між собою, людиною та з різнорідними базами знань. Інтерфейс взаємодії повинен забезпечувати адекватну інтерпретацію запитів і відповідей на них як агентами, так і людьми. Для можливості змістовного аналізу інформації агенти повинні оперувати не

абстрактними наборами даних, а описом їх структурно-семантичної організації у вигляді метаданих (даних про дані). Такі метадані зручно подати у вигляді онтологій, які є сучасним розвитком семантичних мереж. Онтологія – це концептуальна модель предметної області з таксонометричною (інакше – класифікаційною) структурою. Онтологія надає користувачу або агенту узагальнений опис предметної області, формалізує та спрощує сприйняття знань та може доповнюватися новою інформацією [2–5].

Цінність онтологій виявляється у можливості їх зіставлення та об'єднання, у результаті чого можна отримати нові відношення між концептами та нові знання. Так, інтелектуальному агенту часто необхідно порівнювати його власні онтології з іншими, знайденими у мережі, а також з онтологіями інших агентів у процесі взаємодії. Об'єднання онтологій є актуальним через складність використання уніфікованих онтологій у різних галузях знань. Онтології, створені різними розробниками для однієї предметної області, можуть відрізнитися за таксономією, лексикою та семантикою. Об'єднання онтологій створює результуючу онтологію, яка містить елементи вхідних онтологій. Для цього спочатку знаходять сполучну ланку (збіг ідентифікатора, тексту, концептів, ієрархічних відношень, екземплярів класів тощо), а потім за описом елементів знаходять їх місце у новій структурі. На завершення здійснюється формальна перевірка правильності об'єднання онтологій для забезпечення несуперечливості їх синтаксису, структури і семантики. Для перевірки концептуальної коректності результуючої онтології залучають спеціалістів із відповідних предметних областей.

Будуючи онтологію предметної області, розробники стикаються з проблемами її невизначеності: неточністю, неповнотою, відсутністю значень даних або неможливістю їх отримання у поточний момент часу. У зв'язку з цим виникають проблеми врахування невизначеностей при виконанні операцій над онтологіями [6].

У більшості випадків операцію об'єднання доцільно застосовувати для подібних онтологій, які мають непорожню множину сполучних ланок. Автоматизоване виявлення подібних онтологій можна виконати методом кластеризації – групування онтологій у колекції класів за ознаками їх подібності. Методи кластеризації широко застосовуються у задачах інтелектуального аналізу даних та Data mining [7, 8]. Очікується, що кластеризація призведе до зменшення обчислювальних затрат при виконанні операцій над онтологіями.

Кластеризація онтологій буде необхідна для:

- 1) зменшення кількості порівнянь онтологій – простіше працювати з одним кластером онтологій з обмеженою кількістю елементів, ніж з усіма онтологіями;
- 2) побудови загальної представницької онтології одного кластера, отриманої, наприклад, у вигляді його центра;
- 3) зменшення кількості онтологій та обсягу даних, які необхідно зберігати, за рахунок об'єднання подібних онтологій в одну;
- 4) зменшення неоднозначності злиття онтологій завдяки їх подібності у межах одного кластера;
- 5) зменшення невизначеності онтології шляхом об'єднання подібних онтологій одного кластера;
- 6) виявлення новизни інформації для онтологій з найменшою подібністю на інші онтології;
- 7) усунення дублювання, забезпечення цілісності та несуперечливості інформації у базах знань.

В умовах невизначеності онтологій застосування традиційних методів кластерного аналізу, які припускають чітке розбиття початкової множини на підмножини, за яким кожен елемент потрапляє тільки в один кластер, не завжди правильне. Найчастіше необхідно зробити розбиття так, щоб визначити ступінь належності кожної онтології до кожної множини. У цьому випадку доцільно використовувати нечіткі методи кластер-аналізу, наприклад, модифікований метод k-середніх, або метод Густафсона–Кесселя [9, 10]. Однак, як відзначається у [10], ці методи є нестійкими до збурень у вхідних даних та до перекриття кластерів.

У цій роботі пропонується нова модель кластеризації онтологій, основана на повторювальній стохастичній грі. Для цього вводиться поняття інтелектуальних агентів, які репрезентують відповідні онтології. Вступаючи у взаємодію між собою, агенти можуть виконувати потрібні операції над онтологіями, в основному – це визначення ступеня їх подібності. Кількість чистих стратегій агентів визначає кількість кластерів. Гра полягає у тому, що агенти онтологій одночасно, незалежно і випадково вибирають один із кластерів. Імовірність вибору кластерів визначається динамічними векторами змішаних стратегій гравців. Агенти, які вибрали однаковий кластер, утворюють його поточний склад. Для таких агентів обчислюється поточна міра близькості онтологій. Метою кожного кластера агентів є максимізація цієї міри у часі. Міра близькості онтологій використовується для адаптивного перерахунку змішаних стратегій гравців. Збільшуються імовірності вибору тих кластерів, поточний склад яких призвів до зростання міри близькості онтологій. З часом агенти сформують такі вектори змішаних стратегій, які забезпечать максимізацію середніх значень міри близькості розділених на кластери онтологій. Стохастична гра може бути використана для розділення на кластери як детермінованих онтологій, так і для онтологій в умовах невизначеності. Запропонований адаптивний ігровий метод має фільтрувальні властивості щодо викидів у вхідних даних і практично не залежить від закону розподілу випадкових завад.

Отже, **метою цієї роботи** є розроблення ігрової моделі та методу кластеризації онтологій. Для досягнення мети необхідно: визначити міри подібності онтологій; сформулювати ігрову задачу кластеризації онтологій; розробити метод розв'язування стохастичної гри; визначити обмеження на параметри, які визначатимуть збіжність ігрового методу.

Онтологічна модель інженерії знань

Формальна модель онтології задається триплетом:

$$O = \langle C, R, F \rangle,$$

де O – онтологія заданої предметної області; C – скінченна множина концептів; R – скінченна множина відношень між концептами; F – скінченна множина функцій інтерпретації онтології.

Концепти є колекцією сутностей, понять, термінів, категорій або класів [11, 12]. Відношення між концептами є набором атрибутів, результатом застосування яких є інші об'єкти. Функції інтерпретації онтології є аксіомами, які визначають обмеження на інтерпретацію концептів і відношень між ними. До розширеної моделі онтології можуть входити також інші елементи, наприклад: інкапсульовані у клас атрибути даних та їхні значення; екземпляри (об'єкти) класів, що містять конкретні дані, які описуються змістом класу; тематичний словник (тезаурус) термінів предметної області із семантичними відношеннями між лексичними одиницями; алгоритми виведення на онтології; ваги концептів та відношень, що визначають ступінь їх важливості (для адаптивних онтологій [13]).

З погляду об'єктно-орієнтованого програмування концепти є оголошеннями класів, що складаються з окремих атрибутів, які можуть мати різний тип, та методів роботи з цими атрибутами. Класи описують множину об'єктів, які мають спільні властивості. Класи впорядковані за ієрархією (таксономія класів).

Між класами можуть бути наявні відношення належності до певної категорії, серед яких часто використовуються такі: відношення класифікації “є” (is-a, member-of) – визначає, що об'єкт належить множині об'єктів; родовидове відношення гіпонімії (гіпонім – гіперонім) або частковий – загальний, або “різновид” (a-kind-of), або “підмножина” (subset-of) – кожен елемент першої множини входить до другої множини; відношення метонімії (меронім – холонім) або частина – ціле (a-part-of, has-part) – описує частини або складові об'єкти онтології; відношення “містить” (contains) – один клас є контейнером для об'єктів інших класів: існує у вигляді агрегації (aggregation) – якщо контейнер буде знищено, то його вміст не буде знищено, та композиції (composition) – якщо контейнер буде знищено, то весь його вміст буде знищено; відношення асоціації (association) – об'єкти одного класу пов'язані з об'єктами іншого класу; відношення успадкування класів (inheritance) – один клас має властивості іншого класу і може мати додаткові властивості; підклас

класу (subclass-of); бути екземпляром класу (is-an-instance-of); слідувати за (is-consequent); зв'язаний з (connected-to); бути причиною (is-a-cause-of); мати подібність з (has-similarity-with); нащадки одних батьків або родинна подібність одного рівня з (sibling-with); несумісний з (disjoint-with); інші відношення – кількісні, часові, логічні, функціональні, атрибутивні, лінгвістичні тощо.

Аксіоми визначають незаперечні твердження, що пов'язують поняття та відношення. Вони задають інформацію, яка не може бути відображена в онтології за допомогою ієрархії понять та відношень між ними. Також аксіоми визначають правила для поповнення онтології новими даними та дозволяють або забороняють виведення тверджень у межах онтології.

Екземпляри або інстанції – це об'єкти відповідних класів, заповнені конкретними даними. Онтологія разом з множиною екземплярів складає базу знань.

Процес побудови онтології полягає у розділенні цілісної спеціалізованої області знань на окремі об'єкти та формування відношень між ними у зрозумілому для людини вигляді з метою формального опрацювання доступними сьогодні комп'ютерними засобами. Створення онтології, з однієї сторони, є творчим процесом, оскільки автор вкладає у її структуру та семантику своє розуміння предметної області, а з іншої – інженерною справою, оскільки існує велика кількість відповідних засобів, які автоматизують роботу з онтологіями. До таких засобів належать [14–16]:

1) мови і специфікації онтологій (KIF – Knowledge Interchange Format, FrameOntology, Ontolingua, OKBC – Open Knowledge Base Connectivity, XOL – XML-based Ontology Exchange Language, FLogic – Frame Logic, KL-ONE – Knowledge Language ONE, LOOM – Lexical OWL Ontology Matcher, CLASSIC, FaCT – Fast Classification of Terminologies, OCML – Operational Conceptual Modeling Language, XML – eXtensible Markup Language, RDF – Resource Description Framework, RDF Schema, OIL – Ontology Interchange Language, DAML+OIL – DARPA Agent Markup Language+OIL, OWL – Web Ontology Language, SHOE – Simple HTML Ontology Extension, OML – Ontology Markup Language);

2) методології проектування онтологій (METHONTOLOGY, TOVE – Toronto Virtual Enterprise, SENSUS, CommonKADs – Common Knowledge Acquisition and Documentation Structuring, KACTUS, Plinius, ONIONS – ONtologic Integration Of Naive Sources);

3) інструментальні засоби інженерії онтологій (WebOnto, Protege, OntoSaurus, ODE – Ontological Design Environment, KADS22, OntoEdit, SHOE's Knowledge Annotator, i-com).

Онтології широко застосовуються для розв'язування різноманітних інформаційних задач: спільне розуміння людьми та програмними агентами структури різноманітної інформації предметної області; стандартизація та систематизація знань про предметну область; організація високорівневого інтерфейсу для доступу до бази знань предметної області; повторне використання знань у предметній області; аналіз знань у предметній області; організація пошуку інформації; в інформаційно-довідкових системах; для видобування нових знань; виявлення неточностей та логічних помилок в організації бази знань; усунення дублювання інформації; створення глобальної семантичної павутини Semantic Web; організація діалогу між комп'ютером та людиною; машинний переклад текстів; у системах розуміння природної мови – автоматичне реферування тексту, розподіл за рубриками та інше; підтримання безпеки інформаційних ресурсів; у системах електронної комерції; розроблення інтелектуальних агентів та використання мультиагентних систем; побудова алгоритмів виведення в дескрипційних логіках на базі знань агента; організація взаємодії між агентами під час розв'язування задачі.

За рівнем універсальності розрізняють такі види онтологій: верхнього рівня або метаонтології, онтології предметних областей та онтології задач. Прикладами онтологій є [4]: концептуальні – OpenCyc (100 тис. концептів), DOLCE (4 тис. концептів), SUMO (1 тис. концептів); лінгвістичні – WordNet (100 тис. концептів), Omega (120 тис. концептів), MikroKosmos (7 тис. концептів); змішані – Sensus (70 тис. концептів), PropBank (4 тис. концептів), FrameNet (900 фреймів).

Для розв'язування задач використовують різноманітні операції над онтологіями, наприклад [15, 16]: порівняння або зіставлення (matching) – знаходження ступеня відповідності між семантично пов'язаними сутностями двох онтологій; відображення (mapping) – спосіб переведення понять та відношень однієї онтології в іншу, знаходження семантичних зв'язків подібних елементів

з різних онтологій; вирівнювання (alignment) – встановлення різних видів взаємних (в обох напрямках) відповідностей між двома онтологіями з метою їх подальшого використання; об'єднання (merging) – генерування однієї узгодженої онтології з двох заданих; інтеграція (integration) – знаходження однакових частин (відповідностей) видових онтологій з метою побудови родової онтології; успадкування (inheritance) – одна (видова) онтологія успадковує весь вміст іншої (родової) онтології, привносячи свої додаткові властивості; візуалізація (visualization) – зображення концептів та зв'язків між ними у зручному для сприйняття графічному вигляді; масштабування (scaling) – збільшення або зменшення зображення онтології з метою укрупнення або деталізації зв'язків між концептами; фільтрування (filtering) – маскування або демаскування зображення елементів за заданим критерієм, наприклад, за належністю до певного класу; розрахунок характеристик онтології – кількість концептів або зв'язків, віддаленість концептів, ступінь подібності онтологій тощо.

Операції порівняння та об'єднання є найбільш затратними та неоднозначними у зв'язку з тим, що необхідно порівнювати великі набори структурованих даних за різними критеріями (семантичним, структурним або іншим). Для цього використовуються такі програмні інструментальні засоби [14]: Chimaera, OntoMerge, Ontomorph, GLUE, OBSERVER, FCA-Merge – Formal Concept Analysis and Merging, OntEx – Ontology Exploration, ONION – ONtology compositiON, Novu, PROMPT.

На життєвий шлях онтології впливають невизначеності, які можуть бути зумовлені такими факторами: початкова структура онтології залежить від розробника (багатозначність побудови онтології); суперечності у визначенні концептів, відношень між даними, обмежень та аксіом; недовизначеність атрибутів (чи екземплярів) онтології, врахування імовірнісних даних; протиріччя онтологій, наприклад, однаковим елементам онтологій можуть відповідати різні ідентифікатори URI (Uniform Resource Identifier); синонімічна повторюваність термінів веде до неоднозначності інтерпретації запитів; навчання, уточнення онтології та розвиток її структури у часі; структура онтології може змінюватися завдяки виконаним над нею операціям; масштабування онтології та застосування динамічного фільтра призводить до різної (часткової) інтерпретації складу онтології; в онтології можуть виникати осередки збудження та гальмування, які визначаються зміною ваг зв'язків між елементами онтології, тобто активність онтології може змінюватися у часі залежно від передісторії запитів; старіння інформації, закладеної в онтологію – деякі дані або факти з часом перестають бути актуальними; зловмисні спотворення онтологій як новий вид вірусних атак.

Тому результат порівняння онтологій стає неоднозначним. Для роботи з онтологіями в умовах невизначеності необхідно застосовувати адаптивні методи.

Міри подібності онтологій

Результати порівняння, об'єднання і кластеризації онтологій переважно будуть визначатися застосованими мірами оцінювання їх подібності. Найбільш ефективними вважаються гібридні міри, які враховують декілька критеріїв порівняння сутностей. Найчастіше гібридна міра задається адитивною згортокою декількох мір:

$$S(c_1, c_2) = \sum_{i=1}^n w_i S^i(c_1, c_2), \quad (1)$$

де S^i – міра близькості за визначеним критерієм; $w_i > 0$ – вага критерія; $\sum_{i=1}^n w_i = 1$; n – кількість критеріїв; c_1, c_2 – концепти порівнюваних онтологій.

Розрізняють міри близькості понять (концептів), атрибутів та відношень. Детальний огляд мір семантичної близькості розглянуто в [17].

Гібридну міру (1) можна обчислити на основі таксономічної $S^T(c_1, c_2)$, реляційної $S^R(c_1, c_2)$ та атрибутивної $S^A(c_1, c_2)$ мір близькості концептів:

$$S(c_1, c_2) = w_T \cdot S^T(c_1, c_2) + w_R \cdot S^R(c_1, c_2) + w_A \cdot S^A(c_1, c_2), \quad (2)$$

де $w_T, w_R, w_A \in [0, 1]$ – вагові коефіцієнти, $w_T + w_R + w_A = 1$.

Значення гібридної міри $S(c_1, c_2) \in [0, 1]$, причому якщо $S(c_1, c_2) = 0$, то концепти не мають спільних характеристик, і якщо $S(c_1, c_2) = 1$, то концепти ідентичні.

У загальному випадку обчислити міру подібності можна для множини концептів онтології.

Постановка ігрової задачі кластеризації онтологій

Нехай задано скінченну множину онтологій $\Omega = \{O_1, O_2, \dots, O_K\}$, яку необхідно поділити на N кластерів Θ_n ($n = 1..N$). Виділяємо кластери методом стохастичної гри [18, 19], яка задається кортежем $(I, U^i, \Xi^i | i \in I)$, де I – множина гравців; $K = |I| = |\Omega|$ – кількість гравців; $U^i = \{u^i(1), \dots, u^i(N)\}$ – множина чистих стратегій i -го гравця, які визначають вибір одного із кластерів; N – кількість стратегій i -го гравця або кількість кластерів; $\Xi^i : U \rightarrow R^1$ – функція вигравів i -го гравця; $U = \times_{i \in I} U^i$ – множина комбінованих стратегій гравців. Кожен ігровий агент репрезентує одну із заданих онтологій.

Суть гри полягає у керованому випадковому переміщенні агентів з одного кластера в інший. Для цього у моменти часу $t = 1, 2, \dots$ кожен ігровий агент на основі генератора випадкових подій незалежно від інших вибирає чисту стратегію $u^i \in U^i$, яка визначає його входження в один із кластерів. Враховуючи (2), після реалізації комбінованого варіанта $u \in U$ агенти отримують випадкові виграти $x^i(u)$ з апіорі невідомими стохастичними характеристиками:

$$x_t^i = \kappa^i \sqrt{\prod_{j \in I} c(u_t^i = u_t^j) S(c_i, c_j)} + m_t \quad \forall i \in I, \quad (3)$$

де $K_t^i = \sum_{j \in I} c(u_t^i = u_t^j)$ – поточна кількість елементів кластера, до якого входить i -й гравець;

$c() \in \{0, 1\}$ – індикаторна функція події; $m_t \sim Normal(0, d)$ – білий гауссівський шум, або нормально розподілена випадкова величина, яка моделює невизначеність онтології; d – дисперсія розподілу.

Близькість онтологій визначається як середнє геометричне невід’ємних мір подібності порівнюваних концептів. Мультиплікативна згортка (3) не допускає взаємної компенсації мір подібності онтологій у межах кластера: якщо одна із мір подібності дорівнює нулю, то результуюча міра теж дорівнює нулю.

Ефективність ігрової кластеризації визначається функціями середніх вигравів:

$$\Xi_t^i = \frac{1}{t} \sum_{t=1}^t x_t^i \quad \forall i \in I. \quad (4)$$

Мета гри полягає у максимізації системи функцій середніх вигравів (3) у часі:

$$\lim_{t \rightarrow \infty} \Xi_t^i \rightarrow \max \quad \forall i \in I. \quad (5)$$

Отже, на основі спостереження поточних вигравів $\{x_n^i\}$ кожен гравець $i \in I$ повинен навчитися вибирати чисті стратегії $\{u_t^i\}$ так, щоби з часом $t = 1, 2, \dots$ забезпечити виконання системи критеріїв (5).

Метод розв’язування ігрової задачі

Нехай гравці вибирають чисті стратегії u_t^i на основі векторів змішаних стратегій p_t^i , значення яких змінюється у часі:

$$u_t^i = \left\{ U^i(k) \mid k = \arg \left(\min_k \sum_{j=1}^k p_t^i(u_t^i(j)) > w \right), k = 1..N \right\} \quad \forall i \in I, \quad (6)$$

де $w \in [0, 1]$ – дійсне випадкове число з рівномірним розподілом.

Динаміка векторів змішаних стратегій повинна забезпечувати досягнення оптимального колективного розв'язку, наприклад, Неша або Парето під час максимізації (5) функцій середніх виграшів гравців [20].

Метод розв'язування стохастичної гри побудуємо на основі детермінованої матричної гри. Для такої гри визначимо полілінійну функцію середніх виграшів:

$$V^i(p) = \sum_{u \in U} v^i(u) \prod_{j \in I; u^j \in u} p^j(u^j),$$

де $v^i(u) = E\{x_t^i(u)\}$ – математичне сподівання поточних виграшів i -го гравця.

У точках рівноваги за Нешем у змішаних стратегіях справедлива умова доповняльної нежорсткості [18, 21]:

$$\nabla_{p^i} V^i(p) = e^{N_i} V^i(p),$$

де $\nabla_{p^i} V^i(p)$ – градієнт функції середніх виграшів; $e^N = (1_j | j=1..N)$ – вектор, всі компоненти якого дорівнюють 1; $p \in S^M$ – комбіновані змішані стратегії гравців, задані на опуклому одиничному симплексі S^M ($M = N^K$).

Нехай вектор $d = (d_1, \dots, d_N)$ визначає похибку виконання умови доповняльної нежорсткості на симплексі S^M :

$$d = \nabla_{p^i} V^i(p) - e^{N_i} V^i(p) \quad \forall i \in D. \quad (7)$$

У точках рівноваги за Нешем у змішаних стратегіях векторна похибка $d = 0$. Для врахування розв'язків за Нешем у вершинах одиничного симплексу виконаємо зважування похибки (7) елементами векторів змішаних стратегій:

$$diag(p^i) d = 0 \quad \forall i \in D, \quad (8)$$

де $diag(p^i)$ – квадратна діагональна матриця порядку N , побудована з елементів вектора p^i .

Враховуючи, що $diag(p^i) d = E\{x_t^i [e(u_t^i) - p_t^i] | p_t^i = p^i\}$, з (8) на основі методу стохастичної апроксимації [22] отримаємо такий рекурентний марківський метод:

$$p_{t+1}^i = p_{e_{t+1}}^N \left\{ p_t^i + g_t x_t^i (e(u_t^i) - p_t^i) \right\} \quad \forall i \in I, \quad (9)$$

де E – символ математичного сподівання; $p_{e_{t+1}}^N$ – проектор на одиничний N -вимірний симплекс S^N [19]; $g_t > 0$, $e_t > 0$ – монотонно спадні послідовності додатних величин; $e(u_t^i)$ – одиничний вектор, який вказує на вибір чистої стратегії $u_t^i = u^i \in U^i$. Поточні виграші гравців x_t^i визначаються мірою подібності онтологій (3).

Згідно з (9), імовірність вибраного варіанта u_t^i зростає пропорційно до отриманого за це виграшу x_t^i , а імовірності усіх інших варіантів зменшуються. Завдяки таким адаптивним властивостям у процесі гри зростатиме частота вибору тих стратегій, які забезпечують найбільший середній виграш, або найбільше значення міри подібності порівнюваних онтологій. У результаті гри в один кластер потраплять найбільш подібні онтології.

Детерміновані функції g_t та e_t можуть бути задані так:

$$g_t = g t^{-a}, \quad e_t = e t^{-b}, \quad (10)$$

де $g > 0$; $a > 0$; $e > 0$; $b > 0$.

Проектування на розширюваний e_t -симплекс $S_{e_{t+1}}^N$ забезпечує виконання умови $p_t^i[j] \geq e_t$ ($j=1..N$), необхідної для збирання повної статистичної інформації про вибрані чисті стратегії, а параметр $e_t \rightarrow 0$, $t=1,2,\dots$ використовується як додатковий елемент керування збіжністю рекурентного методу.

Необхідною умовою збіжності стратегій (9) до оптимальних значень з імовірністю 1 та у середньоквадратичному є дотримання базових умов стохастичної апроксимації [22].

За умови незалежності випадкових вигащів $\{x_n^i\}$, незалежності вибору чистих стратегій $\{u_n^i\} \forall i \in I$ та виконання умов $g_n > 0$, $g_{n+1} < g_n$, $\sum_{n \rightarrow \infty} g_n = \infty$, $e_n \in (0, \min_{i \in D} N_i^{-1})$, $e_{n+1} < e_n$ метод (9) забезпечує виконання умови доповняльної нежорсткості (8) у знакододатному середовищі $v_{\min}^i > 0 \forall i \in I$ з імовірністю 1, якщо $\sum_{n=1}^{\infty} (|e_n - e_{n-1}| + g_n^2) < \infty$, та у середньоквадратичному, якщо $\lim_{n \rightarrow \infty} (|e_n - e_{n-1}| + g_n) = 0$.

Стохастична гра розпочинається з ненавчених векторів змішаних стратегій зі значеннями елементів $p_0^i(j) = 1/N$, де $j = 1..N$. У наступні моменти часу динаміка векторів змішаних стратегій визначається марківським рекурентним методом (9).

Отже, у момент часу t кожен гравець на основі змішаної стратегії p_t^i вибирає чисту стратегію u_n^i , за що до моменту часу $t+1$ отримує поточний вигащ x_t^i , після чого обчислює змішану стратегію p_{t+1}^i згідно з (9). Завдяки динамічній перебудові змішаних стратегій на основі опрацювання поточних вигащів, метод (9) забезпечує адаптивний вибір чистих стратегій у часі.

Якість ігрової кластеризації даних оцінюється:

1) функцією інтегрованих середніх вигащів:

$$\Xi_t = \frac{1}{K} \sum_{i=1}^K \Xi_t^i, \quad (11)$$

де $K = |I|$ – потужність множини гравців;

2) середньою нормою змішаних стратегій гравців:

$$\Delta_t = \frac{1}{K \cdot t} \sum_{t=1}^t \sum_{i=1}^L \|p_t^i\|. \quad (12)$$

Алгоритм розв'язування стохастичної гри

Крок 1. Задати початкові значення параметрів:

$t = 0$ – початковий момент часу;

$K = |I|$ – кількість гравців;

$\Omega = \{O_1, O_2, \dots, O_K\}$ – множина призначених для кластеризації онтологій;

N – кількість кластерів та кількість чистих стратегій гравців;

$U^i = \{u^i(1), u^i(2), \dots, u^i(N)\}$, $i = 1..K$ – вектори чистих стратегій гравців;

$p_0^i = (1/N, \dots, 1/N)$, $i = 1..K$ – початкові змішані стратегії гравців;

$g > 0$ – параметр кроку навчання;

$a \in (0, 1]$ – порядок зменшення кроку навчання;

e – параметр e -симплекса;

$b > 0$ – порядок розширення e -симплекса;

$d > 0$ – дисперсія завад;

t_{\max} – максимальна кількість кроків методу.

Крок 2. Вибрати варіанти дій $u_t^i \in U^i$, $i = 1..K$ згідно з (6).

Крок 3. Отримати значення поточних вигащів x_t^i , $i = 1..K$ згідно з (3).

Крок 4. Обчислити значення параметрів g_t , e_t згідно з (10).

Крок 5. Обчислити елементи векторів змішаних стратегій p_t^i , $i=1..K$ згідно з (9).

Крок 6. Обчислити характеристики якості кластеризації Ξ_t (11), Δ_t (12).

Крок 7. Задати наступний момент часу $t := t + 1$.

Крок 8. Якщо $t < t_{\max}$, то перейти на крок 2, інакше – кінець.

Результати комп'ютерного моделювання

Приклад 1. Необхідно виконати кластеризацію $K=5$ онтологій за деяким концептом C , який може мати різну реалізацію $C=(c_1, c_2, c_3, c_4, c_5)$ у кожній онтології. Обчислені згідно з (2) міри подібності концептів задано у симетричній матриці:

$$S = \begin{matrix} & \begin{matrix} c_1 & c_2 & c_3 & c_4 & c_5 \end{matrix} \\ \begin{matrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \end{matrix} & \begin{bmatrix} 1 & 0.9 & 0.6 & 0 & 0 \\ 0.9 & 1 & 0.8 & 0 & 0 \\ 0.6 & 0.8 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0.7 \\ 0 & 0 & 0 & 0.7 & 1 \end{bmatrix} \end{matrix}.$$

На основі матриці S обчислено середні значення виграшів агентів, онтологій яких віднесено до кластера $\Theta_{n,t}$:

$$v_{n,t} = L_{n,t} \sqrt{\prod_{i,j \in \Theta_{n,t}} s_{i,j}}, \quad n=1..N,$$

де $\Theta_{n,t}$ – поточний склад множини онтологій Θ_n ; $L_{n,t} = \frac{(k_{n,t} - 1) \cdot k_{n,t}}{2}$ – поточна кількість зв'язків між порівнюваними концептами онтологій одного кластера; $k_{n,t} = |\Theta_{n,t}|$ – поточна кількість елементів (потужність) множини $\Theta_{n,t}$, причому, якщо $k_{n,t} = 1$, то $L_{n,t} = 1$; $s_{i,j}$ – елементи матриці S .

Усі агенти одного кластера отримують однакові поточні виграші:

$$x_{n,t}^i = v_{n,t} + m_t \quad \forall O_i \in \Theta_{n,t}.$$

Значення гауссівського шуму можна обчислити так:

$$m_t = \sqrt{d} \left(\sum_{j=1}^{12} w_{j,t} - 6 \right),$$

де $w \in [0,1]$ – дійсне випадкове число з рівномірним законом розподілу; $d > 0$ – значення дисперсії.

Варіанти розділення онтологій на $N=2$ кластери та відповідні значення середньої міри подібності онтологій подано у табл. 1 для $d=0$. Інші можливі варіанти є симетричними.

Таблиця 1

Варіанти кластеризації онтологій

Θ_1	\emptyset	{1}	{2}	{3}	{4}	{5}	{1, 2}	{1, 3}
$v_{1,t}$	0	1	1	1	1	1	0.9	0.6
Θ_2	{1, 2, 3, 4, 5}	{2, 3, 4, 5}	{1, 3, 4, 5}	{1, 2, 4, 5}	{1, 2, 3, 5}	{1, 2, 3, 4}	{3, 4, 5}	{2, 4, 5}
$v_{2,t}$	0	0	0	0	0	0	0	0
Θ_1	{1, 4}	{1, 5}	{2, 3}	{2, 4}	{2, 5}	{3, 4}	{3, 5}	{4, 5}
$v_{1,t}$	0	0	0.8	0	0	0	0	0.7
Θ_2	{2, 3, 5}	{2, 3, 4}	{1, 4, 5}	{1, 3, 5}	{1, 3, 4}	{1, 2, 5}	{1, 2, 4}	{1, 2, 3}
$v_{2,t}$	0	0	0	0	0	0	0	0.756

Результати моделювання ігрової кластеризації онтологій подано на рис. 1 для таких параметрів стохастичної гри: $K = 5$, $N = 2$, $U^i = (1, 2)$, $g = 1$, $e = 0.999/N$, $a = 0.01$, $b = 1$, $d = 0$, $t_{\max} = 10^4$.

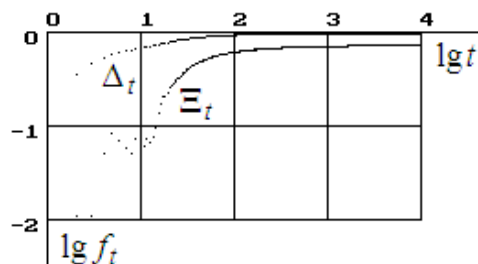


Рис. 1. Характеристики збіжності стохастичної гри

Зростання інтегральної функції середніх виграшів Ξ_t свідчить про те, що у кожен із сформованих кластерів потрапили найподібніші онтології. Наближення інтегральної норми Δ_t^i змішаних стратегій до логарифмічного нуля свідчить про збіжність ігрового методу у чистих стратегіях.

У табл. 2 подано значення стратегій гравців на момент завершення гри.

Таблиця 2

Розв'язок гри у чистих стратегіях

	p^1	p^2	p^3	p^4	p^5
u^1	0	0	0	1	1
u^2	1	1	1	0	0

Рівні 1 значення елементів векторів змішаних стратегій вказують на вибір гравцями одного із кластерів. З отриманих результатів видно, що у процесі стохастичної гри сформовано два кластери онтологій: $\Theta_1 = \{O_4, O_5\}$ та $\Theta_2 = \{O_1, O_2, O_3\}$.

Якщо знайдеться онтологія з однаковою мірою подібності до онтологій двох (або більше) кластерів, то отримаємо розв'язок гри у змішаних стратегіях. Тоді елементи вектора змішаних стратегій будуть задавати імовірності належності онтології до декількох кластерів.

Приклад 2. Нехай матриця мір подібності варіантів реалізації концепту у $K = 5$ онтологіях має вигляд:

$$S = \begin{matrix} & \begin{matrix} c_1 & c_2 & c_3 & c_4 & c_5 \end{matrix} \\ \begin{bmatrix} 1 & 0.7 & 0 & 0 & 0.3 \\ 0.7 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0.7 & 0 \\ 0 & 0 & 0.7 & 1 & 0.3 \\ 0.3 & 0 & 0 & 0.3 & 1 \end{bmatrix} & \begin{matrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \end{matrix} \end{matrix}.$$

Як і раніше, виконаємо розбиття онтологій на $N = 2$ кластери. Зі структури матриці видно, що реалізація концепту в онтології O_5 однаковою мірою подібна до його реалізації в неперетинних онтологіях O_1 та O_4 . Тому слід очікувати розв'язок гри у змішаних стратегіях. Отримані під час гри фінальні значення змішаних стратегій зображено у табл. 3.

Таблиця 3

Розв'язок гри у змішаних стратегіях

	p^1	p^2	p^3	p^4	p^5
u^1	1	1	0	0	0.5
u^2	0	0	1	1	0.5

З наведених у табл. 3 результатів випливає, що сформовано розмите розбиття онтологій на кластери $\Theta_1 = \{O_1, O_2, O_5\}$ та $\Theta_2 = \{O_3, O_4, O_5\}$. Онтологія O_5 одночасно потрапляє у два кластери. Ступінь належності онтологій до конкурентних кластерів визначається імовірностями, що є елементами векторів змішаних стратегій. Для нечіткої кластеризації зображений на рис. 1 графік інтегральної норми Δ_i^j змішаних стратегій не досягатиме логарифмічного нуля.

Дія завад на визначення міри подібності онтологій забезпечує детерміновану кластеризацію замість нечіткої. У результаті онтологія O_5 з близькою до 1 імовірністю належатиме до одного із конкурентних кластерів. Це пов'язано з селективними властивостями запропонованого ігрового методу. Диференційний зсув однакової міри подібності онтологій, який виникає за дії білого шуму, підсилює імовірність вибору одного із кластерів. У табл. 4 наведено результати стохастичної гри для мір подібності онтологій прикладу 2, змінених дією білого шуму зі значенням дисперсії $d = 0.25$. Замість розв'язку гри у змішаних стратегіях отримано розв'язок у чистих стратегіях.

Таблиця 4

Розв'язок гри в умовах стохастичної невизначеності

	p^1	p^2	p^3	p^4	p^5
u^1	1	1	0	0	0 (1)
u^2	0	0	1	1	1 (0)

Період навчання стохастичної гри залежить від розмірності ігрової задачі, дисперсії завад та співвідношень регульованих параметрів ігрового методу. Слід зазначити, що в умовах стохастичної невизначеності ($d > 0$) збільшується період навчання стохастичної гри, необхідний для пошукової кластеризації онтологій.

Висновки

У цій статті запропоновано новий метод адаптивної кластеризації онтологій, що базується на моделі стохастичної гри. Метод побудовано застосуванням стохастичної апроксимації умови доповняльної нежорсткості, яка описує рівноважний за Нешем розв'язок гри. Розроблений ігровий метод забезпечує кластеризацію онтологій в умовах стохастичної невизначеності.

Результати комп'ютерного експерименту підтверджують збіжність ігрового методу у процесі пошукової кластеризації онтологій за дотримання умов та обмежень стохастичної оптимізації. Вірогідність отриманих результатів підтверджується повторюваністю розподілу онтологій на кластери, отриманого для різних послідовностей випадкових величин.

Недоліком запропонованого методу є невисока (зі степеневим порядком) швидкість збіжності, зумовлена процесом навчання ігрових агентів в умовах апріорної невизначеності. Позитивним моментом є природна можливість розпаралелювання ігрової задачі із застосуванням потужних обчислювальних засобів для прискорення процесу кластеризації.

Ігрова кластеризація може бути використана для оптимізації виконання операцій над онтологіями і як окремий приклад самоорганізації стохастичної гри інтелектуальних агентів.

Список літератури

1. Wooldridge, M. (2009). *An Introduction to Multiagent Systems*. United Kingdom: John Wiley & Sons.
2. Рогушина, Ю. В. (2018). Теоретичні засади застосування онтологій для систематизації ресурсів WEB. *Проблеми програмування*, 2–3, 197–203.
3. Hashemi, P., Khadivar, A., Shamizanjani, M. (2018). Developing a domain ontology for knowledge management technologies. *Online Information Review*, 42 (1), 28–44.
4. Довгий, С. О., Велічко, В. Ю., Глоба, Л. С. та ін. (2013). *Комп'ютерні онтології та їх використання у навчальному процесі. Теорія і практика: Монографія*. – К.: Інститут обдарованої дитини.
5. Буров, Є. В., Пасічник, В. В. (2015). Програмні системи на базі онтологічних моделей задач. *Вісник Нац. ун-ту “Львівська політехніка”. Серія: “Інформаційні системи та мережі”*, 829, 36–57.
6. Berko, A., Aliksieiev, V. (2018) A method to solve uncertainty problem for Big Data sources. *Proceedings of the 2018 IEEE 2nd International Conference on Data Stream Mining and Processing (DSMP)*. Lviv, Ukraine, August 21–25, 32–37.
7. Mirkin, B.G. (2005). *Clustering for Data Mining. A Data Recovery Approach*. CRC Press.
8. Batet, M. (2011). Ontology-based semantic clustering. *AI Communications*, 24 (3), 291–292.
9. Зайченко, Ю. П., Гончар, М. А. (2007) Нечеткие методы кластерного анализа в задачах автоматической классификации в экономике. *Вісник НТУ України “Київський політехнічний інститут”. Серія: “Інформатика, управління та обчислювальна техніка”*, 47, 198–206.
10. Бодянский, Е. В., Колчигин, Б. В., Волкова, В. В., Плисс, И. П. (2013). Адаптивная нечеткая кластеризация данных на основе метода Густафсона-Кесселя. *Управляющие системы и машины*, 2, 40–46.
11. Литвин, В. В., Висоцька, В. А., Досин, Д. Г., Гірняк, М. Г. (2015). Розроблення методів та засобів побудови інтелектуальних систем опрацювання інформаційних ресурсів з використанням онтологічного підходу // *Вісник Нац. ун-ту “Львівська політехніка”. Серія: “Інформаційні системи та мережі”*, 832, 295–314.
12. Aleman, Y., Somodevilla, M. J. (2017). A proposal for domain ontological learning. *Research in Computing Science*, 133, 63-70.
13. Литвин, В. В. (2011). Інтелектуальні агенти пошуку релевантних прецедентів на основі адаптивних онтологій. *Математичні машини і системи*, 3, 66–72.
14. Овдій, О. М., Проскуріна, Г. Ю. (2004). Онтології у контексті інтеграції інформації: представлення, методи та інструменти побудови. *Проблеми програмування*, 2–3, 353–365.
15. Чистякова, И. С. (2014). Инженерия онтологий. *Инженерия программного обеспечения*, 4 (20), 53–68.
16. Slimani, T. (2015). Ontology Development: A Comparing Study on Tools, Languages and Formalisms. *Indian Journal of Science and Technology*, 8 (24), 1–12.
17. Крюков, К. В., Панкова, Л. А., Пронина, В. А., Суховеров, В. С., Шипилина, Л. Б. (2010). Меры семантической близости в онтологии. *Проблемы управления*, 5, 2–14.
18. Neyman, A., Sorin, S. (2012). *Stochastic Games and Applications*. Springer Science & Business Media.
19. Назин, А. В., Позняк, А. С. (1986). *Адаптивный выбор вариантов: Рекуррентные алгоритмы*. Москва: Наука.
20. Petrosjan, L. A., Mazalov, V. V. (2007). *Game Theory and Application*. New York: Nova Science Publishers.
21. Neogy, S. K., Vapat, Ravindra B., Dubey, Dipti. (2018). *Mathematical Programming and Game Theory*. Springer.

22. Kushner, H., Yin, G. G. (2013). *Stochastic Approximation and Recursive Algorithms and Applications*. Springer Science & Business Media.

References

1. Wooldridge, M. (2009). *An Introduction to Multiagent Systems*. United Kingdom: John Wiley & Sons.
2. Rogushina, Yu. V. (2018). Theoretical principles of application of ontologies for systematization of WEB resources. *Problems of programming*, 2–3, 197–203.
3. Hashemi, P., Khadivar, A., Shamizanjani, M. (2018). Developing a domain ontology for knowledge management technologies. *Online Information Review*, 42 (1), 28–44.
4. Dovgy, S. O., Velychko, V. Yu., Globa, L. S., et al. (2013). Computer ontologies and their use in the educational process. Theory and practice: Monograph. *Kyiv: Institute of Gifted Child*.
5. Burov, E. V., Pasichnyk, V. V. (2015). Software systems based on ontological task models. *Bulletin of the Lviv Polytechnic National University. Series: "Information Systems and Networks"*, 829, 36–57.
6. Berko, A., Aliksieiev, V. (2018) A method to solve uncertainty problem for Big Data sources. *Proceedings of the 2018 IEEE 2nd International Conference on Data Stream Mining and Processing (DSMP)*. Lviv, Ukraine, August 21–25, 32–37.
7. Mirkin, B. G. (2005). *Clustering for Data Mining. A Data Recovery Approach*. CRC Press.
8. Batet, M. (2011). Ontology-based semantic clustering. *AI Communications*, 24 (3), 291–292.
9. Zaychenko, Yu. P., Gonchar, M. A. (2007) Fuzzy methods of cluster analysis in problems of automatic classification in economics. *Bulletin of the NTU of Ukraine "Kyiv Polytechnic Institute". Informatics. Series: "Management and Computing"*, 47, 198–206.
10. Bodiansky, E. V., Kolchigin, B. V., Volkova, V. V., Pliss, I. P. (2013). Adaptive fuzzy clustering of data based on the Gustafson-Kessel method. *Control systems and machines*, 2, 40–46.
11. Lytvyn, V. V., Vysotska, V. A., Dosyn, D. G., Girnyak, M. G. (2015). Development of methods and means of constructing intelligent systems for processing information resources using the ontological approach. *Bulletin of the Lviv Polytechnic National University. Series: "Information systems and networks"*, 832, 295–314.
12. Aleman, Y., Somodevilla, M. J. (2017). A proposal for domain ontological learning. *Research in Computing Science*, 133, 63–70.
13. Lytvyn, V. V. (2011). Intelligent search agents of relevant precedents based on adaptive ontologies. *Mathematical Machines and Systems*, 3, 66–72.
14. Ovdii, O. M., Proskurina, G. Yu. (2004). Ontology in the context of information integration: concepts, methods and construction tools. *Problems of Programming*, 2–3, 353–365.
15. Chistyakova, I. S. (2014). Ontology engineering. *Software Engineering*, 4 (20), 53–68.
16. Slimani, T. (2015). Ontology Development: A Comparing Study on Tools, Languages and Formalisms. *Indian Journal of Science and Technology*, 8 (24), 1–12.
17. Krjukov, K. V., Pankova, L. A., Pronina, V. A., Sukhoverov, V. S., Shipilina, L. B. (2010). Measures of semantic proximity in ontology. *Problems of Management*, 5, 2–14.
18. Neyman, A., Sorin, S. (2012). *Stochastic Games and Applications*. Springer Science & Business Media.
19. Nazin, A. V., Poznyak, A. S. (1986). *Adaptive Choice of Variants: Recurrence Algorithms*. Moscow: Science.
20. Petrosjan, L. A., Mazalov, V. V. (2007). *Game Theory and Application*. New York: Nova Science Publishers.
21. Neogy, S. K., Bapat, Ravindra B., Dubey, Dipti. (2018). *Mathematical Programming and Game Theory*. Springer.
22. Kushner, H., Yin, G. G. (2013). *Stochastic Approximation and Recursive Algorithms and Applications*. Springer Science & Business Media.