

С.Д. Погорілий М. С. Слинко, П.В. Білецький

ФОРМАЛЬНА ВЕРИФІКАЦІЯ ВЛАСТИВОСТЕЙ МОДЕЛІ ПОШУКУ КОРЕФЕРЕНТНИХ ОБ'ЄКТІВ НА ОСНОВІ ДЕРЕВ РІШЕНЬ

В роботі розглядається проблема пошуку кореферентних об'єктів, яка полягає в знаходженні об'єктів - слів або словосполучень в тексті, які посилаються на один і той же реальний чи уявний об'єкт. Розв'язання цієї задачі розглядається для україномовних текстів, використовуючи дерева рішень, які самостійно формують свою структуру на основі навчальної вибірки. Завдяки можливості графічного представлення дерева рішень, на відмінну від інших алгоритмів машинного навчання, таких як нейронні мережі, стає можливим здійснення аналізу внутрішньої структури. Її можна використовувати для пояснення окремих результатів роботи дерева, що значно полегшує проведення формальної верифікації їх властивостей. Для створення дерева рішень використовуються векторні представлення слів Elmo та інші лінгвістичні характеристики. Після формування дерева рішень використовуються для бінарної класифікації вхідної пари потенційно кореферентних об'єктів. На основі отриманого бінарного класифікатора здійснюється об'єднання кореферентних об'єктів в кластери та подальша оцінка правильності кластеризації за допомогою спеціальних метрик. В роботі проведено детальний опис реалізованого застосунку та структури сформованого дерева рішень, що є основою для подальшого аналізу. Запропоновано застосування апарату транзиторних систем для побудови високорівневої специфікації моделі пошуку кореферентних об'єктів. Модель на основі транзиторних систем дозволяє виконувати аналіз застосунку на нескінченній послідовності станів, забезпечуючи відсутність помилок в процесі виконання. Проведено формалізацію та використано автоматні моделі та лінійно-темпоральну логіку для верифікації набору властивостей отриманої специфікації. Створено автомати Бюхі, що акцептують слова, які підтверджують властивості, та знайдено приклади та контрприкладні властивостей, що аналізуються. Визначений в роботі метод є основою для створення автоматизованих аналізаторів застосунків пошуку кореферентних об'єктів на основі дерев рішень.

Ключові слова: штучний інтелект, обробка природних мов, кореферентність, дерева рішень, транзиторні системи, автомат Бюхі.

S.D. Pogorilyy, M. S. Slynko, P. V. Biletskyi

FORMAL VERIFICATION OF THE PROPERTIES OF COREFERENT RESOLUTION MODEL BASED ON DECISION TREES

The paper examines the problem of coreference resolution, which involves identifying objects - words or phrases in a text, that refer to the same real or imaginary entity. The solution of this task is explored for Ukrainian-language texts using decision trees, which autonomously structure themselves based on training data. Unlike other machine learning algorithms such as neural networks, decision trees allow for analysis of their internal structure through graphical representation. This feature facilitates explaining individual results produced by the tree, significantly easing formal verification of their properties. To create decision trees, vector representations of words (such as Elmo) and other linguistic features are used. After formation, decision trees are employed for binary classification of input pairs potentially referring to the same coreferent objects. Based on the obtained binary classifier, coreferent objects are grouped into clusters, followed by an evaluation of the clustering accuracy using specialized metrics. The paper provides a detailed description of the implemented application and the structure of the formed decision tree, which serves as the basis for further analysis. Additionally, the use of transition systems is proposed to construct a high-level specification model for coreference resolution. The transition system-based model enables analysis of application behavior on infinite state sequences, ensuring error-free execution. Formalization is carried out, and automata models along with linear-temporal logic are used to verify a set of properties of the obtained specification.

Büchi automata are created to accept words confirming the properties, and examples as well as counterexamples of the analyzed properties are found. The method defined in the paper serves as the foundation for creating automated analyzers for coreference resolution applications based on decision trees.

Key words: artificial intelligence, natural language processing, coreference resolution, decision trees, transition systems, Büchi automata.

Вступ

Пошук кореферентних об'єктів (Coreferene Resolution) – це задача, яка належить до обробки природної мови (Natural Language Processing, NLP). Задача пошуку кореферентних об'єктів полягає у знаходженні усіх мовленнєвих об'єктів тексту (таких як іменники, займенники та словосполучення), що позначають один реальний чи уявний об'єкт. Результатом розв'язку цієї задачі є встановлення відповідності між об'єктами тексту, що вказують на один і той же об'єкт; така відповідність встановлюється для пари об'єктів чи для їх кластеру.

Приклади кореферентних об'єктів [1] наведені далі; референт (іменник) виділений жирним шрифтом, займенники - підкреслені.

- Проста анафора (іменник стоїть перед займенником в тексті): «Він перейшов через **гору**. Вона була високою.»
- Проста катафора (іменник згадується після займенника): «Вона вийшла на дорогу, що вела праворуч. **Марія** сьогодні мала гарний настрій.»
- Складений референт: «**Іван, Михайло та Остап** – всі вони працювали під землею.»

Методи автоматизованого розв'язання задачі пошуку кореферентних об'єктів включають в себе алгоритми на основі строгих правил, що формуються кваліфікованими лінгвістами та методи штучного інтелекту (Artificial Intelligence, AI). До методів штучного інтелекту відносять нейронні мережі, мовні моделі та дерева рішень. Дерева рішень, на відмінну від інших методів штучного інтелекту, мають будову, що уможливує відносно легкий аналіз логіки їх внутрішньої роботи та внесення корекцій в їхню структуру для зміни логіки класифікації.

В процесі створення великих, складних програмних систем часто виникають помилки. Тому актуальним є питання забезпечення надійності таких систем. Зазвичай для пошуку помилок у процесі розробки програмних продуктів виконується його те-

стування. Воно дозволяє знаходити помилки в програмі, але не гарантує їх виявлення. Для детальнішого дослідження надійності застосунків використовують методи формальної верифікації системи.

Для проведення формальної верифікації необхідно побудувати формальну модель системи. Подати систему формально можна використовуючи розмічені транзиційні системи [2]. Вони дозволяють описати систему набором дискретних станів, між якими існують переходи, що здійснюються під час виконання певних умов та позначають операції, які виконуються системою у процесі переходу між станами.

Наступним кроком після створення формальної моделі необхідно визначити властивості системи, які будуть аналізуватися та подати їх формально.

Після цього, отримавши модель системи та властивості для її перевірки, використовуються формальні методи доведення виконання (чи невиконання) властивостей. Для цього використовують автомати Бюхі [3].

У роботі розглядається проблема пошуку кореферентних об'єктів в україномовних текстах, використовуючи дерева рішень. Запропоновано застосування апарату транзиційних систем для побудови високорівневої специфікації моделі пошуку кореферентних об'єктів. Проведено формалізацію та використано автоматні моделі та лійно-темпоральну логіку для верифікації набору властивостей отриманої специфікації. Створено автомати Бюхі, що акцептують слова, які підтверджують властивості та знайдено приклади й контрприкладів властивостей, що аналізуються.

Використання дерев рішень для пошуку кореферентних об'єктів

Дерево рішень - це ієрархічна структура, що складається з вузлів (кореня – початкового вузла, внутрішніх та кінцевих вузлів). Кожен некінцевий вузол такого дерева посилається на два піддерева (або вузла-нащадка). Формування дерева рішень

може здійснюється автоматично на основі набору даних.

Набір даних повинен містити елементи та мітки класів для них. Кожен елемент складається з характеристик (features), які можуть приймати дійсні або булеві значення (тобто, значення, які підтримують операцію порівняння, що необхідно для функціонування дерева). В роботі [1] набір даних для пошуку кореферентних об'єктів складається із елементів, кожен з яких описує пару потенційно кореферентних об'єктів та мітки, що вказує, чи ці об'єкти кореферентні. Кожен елемент містить в собі характеристики пари, такі як: співпадіння числа, роду, частина мови першого та другого об'єкту, лематизованої версії об'єктів, кількість слів між об'єктами, міру косинусної схожості векторів об'єктів, що розглядаються та інші. Всі ці характеристики отримані автоматично, використовуючи бібліотеку UDpipe [4], модель для створення векторних представлень слів ELMo [5] та за допомогою власних алгоритмів. Набір даних (2500 текстів) був розділений на навчальну (1500) та перевіірочну (1000) вибірки.

Дерево рішень для пошуку кореферентних об'єктів у роботі [1], створене за допомогою бібліотеки scikit-learn [6], формується шляхом вибору певної характеристики на кожному кроці в процесі поглиблення. Ця характеристика обирається таким чином, щоб найкраще розділити множину, що розглядається в конкретному піддереві, на класи. Для вибору цієї характеристики використовується коефіцієнт Джині (Gini impurity) [7], який дозволяє оцінити імовірність невірної класифікації випадково вибраного об'єкту з підгрупи. Дерево рішень для пошуку кореферентних об'єктів показано на рис.1, 2.

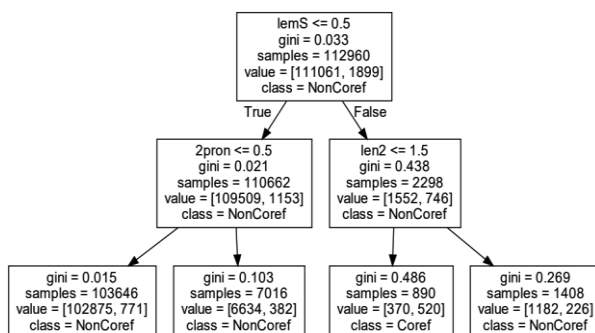


Рис. 1. Дерево рішень [1]

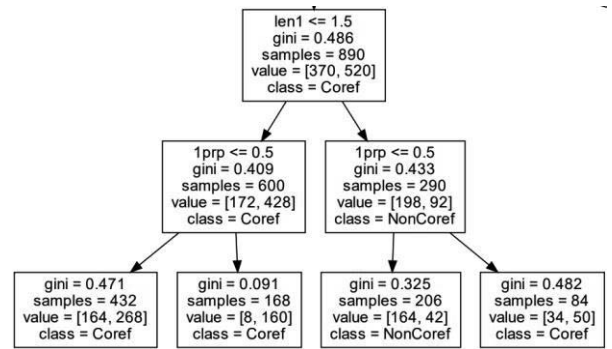


Рис. 2. Піддерево дерева рішень [1]

Як і іншим методам штучного інтелекту, деревам рішень властиве явище перенавчання (overfitting), яке полягає в надмірному пристосуванні структури дерева до набору даних, що використовується для його створення. Водночас точність роботи алгоритму на даних, які не використовувалися для навчання, падає. Для подолання цього явища в роботі [1] використано параметр `min_impurity_decrease`, який дозволяє обмежити поглиблення дерева, якщо із подальшим розділенням на підмножини зменшення коефіцієнта Джині менше порогового значення.

Створене дерево рішень дозволяє класифікувати вхідні об'єкти шляхом переходу в піддерева, починаючи з кореня, у відповідності із правилом, вказаним у поточному вузлі. Оскільки кожен вхідний об'єкт описує пару потенційно кореферентних об'єктів, дерево рішень виконує задачу бінарної класифікації.

Кореферентні зв'язки можуть існувати більше ніж між двома об'єктами в тексті, наприклад, включати три, чотири та більше об'єктів. Тому для отримання результатів кореферентні об'єкти об'єднуються в кластери. В роботі [1] початково всі потенційно кореферентні об'єкти вважаються кластерами. Їх злиття відбувається, якщо хоча б одна пара потенційно кореферентних об'єктів з першого та другого кластерів розпізнана як кореферентна.

Оцінка якості кластеризації виконується шляхом порівняння отриманих кластерів з оригінальними, використовуючи спеціальні метрики. Така оцінка виконується на перевіірочній вибірці. Отримані в роботі [1] результати показують високу

ефективність алгоритму, близькі до результатів моделі на основі нейронних мереж BiLSTM [8].

Формальна верифікація алгоритмів із використанням дерев рішень

Як показано в оглядовій статті [9], в існуючих дослідженнях використання формальної верифікації для моделей машинного навчання застосовується апарат розв’язності формул у теоріях (Satisfiability Modulo Theories, SMT) та лінійної оптимізації (Linear Programming, LP). У роботі пропонується підхід до верифікації з використанням автоматних моделей та лінійно-темпоральної логіки, що, на відміну від SMT, дозволяє досліджувати темпоральні характеристики моделі на потенційно нескінченній послідовності станів. Як правило, математична модель дискретної системи являє собою граф, вершини якого відповідають станам (або класам станів), в яких може перебувати система в різні моменти часу; а ребрам – переходи між станами, які можуть мати позначки, що зображують дії чи події, які виконує система. Функціонування системи за такого зображення представляється послідовностями переходів від одного стану до іншого. Якщо ребро має позначку, то ця позначка являє собою дію системи, що виконується під час переходу від стану на початку ребра до стану в його кінці. Далі в роботі використовуються розмічені транзиційні системи (розмічені ТС, або РТС) як дискретна модель обчислень загального типу [2].

Створення РТС моделей на високому рівні абстракції

З точки зору моделювання застосунків, що використовує дерева рішень для пошуку корелюючих об’єктів, можна представити у вигляді взаємодії таких систем:

- ТС 1, або “керуюча” система: відповідає за взаємодію з зовнішніми ресурсами;
- ТС 2, або “ядро”: система що представляє прохід по дереву рішень.

Керуюча система моделюється ТС

$M = (\{v_0, v_1, v_2\}, \{a_1, a_2, a_3\}, \alpha, \beta, v_0)$, (1)
де v_0 - система в стані доступності; v_1 - стан, в якому система опрацьовує вхідні дані; v_2 - стан, в якому система видає результат. Переходи інтерпретуються наступним чином: a_1 - отримання нового набору вхідних даних; a_2 - формування результату класифікації; a_3 - перехід в стан доступності.

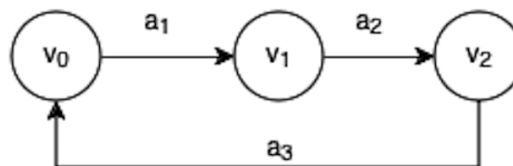


Рис. 3. Представлення ТС моделі керуючої системи

Для наочності за ядро використаємо піддерево дерева рішень, отриманого у [1]. Зазначимо, що в процесі створення моделі до піддерева було додано 2 сурогатні стани: початковий s_0 та кінцевий s_6 ; а також перехід t_{11} між ними. Це необхідно для представлення дерева рішень як неперервно функціонуючої системи, що дозволяє використання темпоральної логіки для подальшого аналізу. Кінцева модель визначається як

$$S_K = \{s_0, s_1, s_2, s_3, s_4, s_5, s_6\}, \quad (2)$$

$$T_K = \{t_1, t_2, \dots, t_{11}\}, \quad (3)$$

$$K = (S_K, T_K, \alpha, \beta, s_0), \quad (4)$$

а множина пропозиційних формул, асоційованих зі станами, та функція позначок для кожного стану наведені на рис. 4.

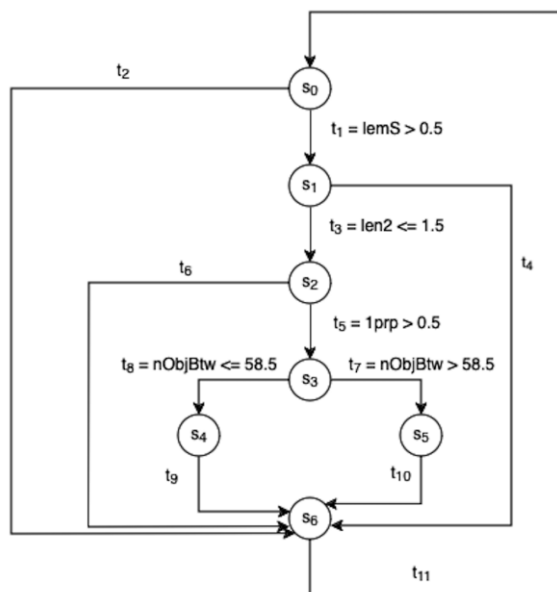


Рис. 4. Представлення ТС підмоделі ядра

Важливо зазначити, що у обраному для моделювання піддереві кожен із підмножини станів $\{s_2, s_3, s_4\}$ фіксує кореферентність вхідних даних. Відповідно, кожен зі станів $\{s_0, s_1, s_5\}$ фіксує відсутність кореферентності, а стан s_6 зберігає клас кореферентності, визначений раніше.

Для вищенаведених ТС будуємо синхронізовану паралельну композицію (синхронний добуток) з глобальними переходами, що моделює роботу застосунку в цілому. Множина обмежень синхронізації містить наступні елементи (ε - тотожна дія, що означає відсутність переходу в ТС):

$$T = \{(a_1, t_1), (a_1, t_2), (\varepsilon, t_3), (\varepsilon, t_4), (\varepsilon, t_5), (\varepsilon, t_6), (\varepsilon, t_7), (\varepsilon, t_8), (\varepsilon, t_9), (\varepsilon, t_{10}), (a_2, \varepsilon), (a_3, t_{11})\}. \quad (5)$$

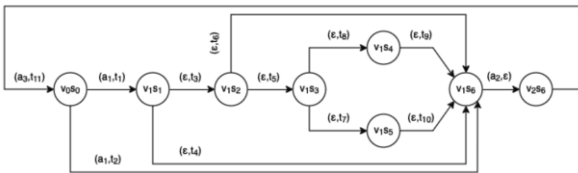


Рис. 5. Синхронний добуток ТС 1 і 2

Верифікація властивостей моделі автомата Бюхі

В роботі пропонується використання наступного алгоритму перевірки лінійно-темпоральної формули P , що представляє властивість, яка визначає семантичну коректність роботи системи:

1. Створити автомат Бюхі, що акцептує слова, які підтверджують P .
2. Побудувати добуток автомата і ТС, що моделює вихідну систему.
3. Знайти переріз шляхів, які генерує загальна транзиторна система (ЗТС), і шляхів, які акцептує автомат. Подальший аналіз досяжних станів перерізу дозволяє знайти як приклади, так і контрприклад формули P [3].

Розглянемо використання алгоритму на прикладі: нехай існує гіпотеза, що, у разі аналізу кореферентності двох об'єктів довжина другого об'єкту є малою, то об'єкти кореферентні. Така властивість представляється формулою лінійно-темпоральної логіки

$$P_1 = (\text{len2} \leq 1.5) \rightarrow G(F(\text{coref})), \quad (6)$$

тобто якщо умова справджується, то система рано чи пізно перейде в стан, що фіксує клас кореферентності, і залишиться в такому стані.



Рис. 6. Зображення автомата Бюхі, що відповідає LT формулі P_1

Автомат Бюхі, що допускає слова, які відповідають формулі P_1 , містить два стани: початковий p_0 та кінцевий p_1 , перейшовши в який, автомат вже не змінює свій стан незалежно від поданих слів.

Побудуємо перетин автомата Бюхі (рис. 6) та ТС, що моделює синхронний добуток ТС 1 і 2 (рис. 5). Спрощене візуальне представлення отриманого перетину наведено на рис. 7. Для наочності на рис. 7 приховані недосяжні стани.

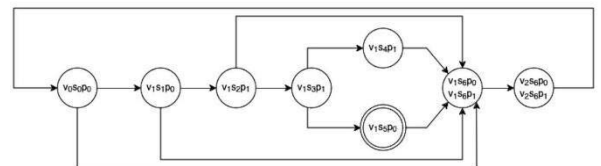


Рис. 7. Добуток автомата Бюхі та синхронного добутку ТС

Отриманий перетин дозволяє провести низку аналітичних дослідів можливих шляхів, циклів і трас. Зокрема, він дозволяє стверджувати, що можливий шлях, за якого відбувається перехід із стану p_1 у стан p_0 (кінцевий стан такого переходу виділено двома концентричними колами), тобто знайдено контрприклад властивості P . Дійсно, лише інформації про довжину одного з об'єктів мало, щоб стверджувати про їх кореферентність: обхід дерева з вказаним обмеженням може завершитися як у стані s_4 , що фіксує кореферентність, так і в стані s_5 , що фіксує її відсутність. Якщо ж розглянути іншу формулу лінійно-темпоральної логіки, наприклад,

$$P_2 = (\text{lemS} \wedge \underline{1prp} \wedge \text{len2} \leq 1.5 \wedge$$

$$\wedge len1 \leq 1.5 \wedge nObjBtw \leq 58.5) \rightarrow G(F(coref)), (7)$$

то аналіз добутку автомата Бюхі та синхронного добутку ТС покаже відсутність шляху-циклу таких, що є доступним із початкового стану, та який включає в себе стан із множини недопустимих станів, тобто формула P_2 є істинною, а, отже, властивість, представлена нею, виконується завжди.

Висновки

Досліджено використання математичного апарату транзиційних систем, який дозволяє отримати формалізовану специфікацію системи, що аналізується. До переваг сформованої моделі належить можливість здійснювати формальну верифікацію властивостей (формул) на потенційно нескінченній послідовності станів, таким чином гарантуючи відсутність помилок протягом всього часу функціонування. Це є неможливим за використання інших поширених апаратів формалізації (як SMT), що вирішують задачу верифікації лише в конкретний статичний момент часу.

В роботі виконано перетворення набору властивостей системи пошуку кореферентних об'єктів з використанням дерев рішень у формули лінійно-темпоральної логіки. Створено автомати Бюхі, що акцептують слова, які підтверджують формули та проілюстровано використання частини отриманої моделі як для знаходження контрприкладу властивості, що аналізується, так і для верифікації їх відсутності. Поданий підхід є базою для створення автоматизованих аналізаторів систем дослідження кореферентності текстів із використанням дерев рішень.

Література

1. С.Д. Погорілий, П.В. Білецький. Алгоритм пошуку кореферентних об'єктів в україномовних текстах із використанням дерев рішень. Проблеми програмування. 2022. № 3-4. с. 85-91.
2. Ю.В. Бойко, С.Л. Кривий, С.Д. Погорілий та ін. Методи та новітні підходи до проектування, управління і застосування високопродуктивних ІТ-інфра-

структур. – К.: ВПЦ «Київський університет». 2016. 447 с

3. S.L. Kryvyi et al. Method of semantic application verification in GPGPU technology. System Research & Information Technologies, 2020, № 3, pp. 7-22
4. Бібліотека UDpipe. Accessed: 06.04.2024. <https://lindat.mff.cuni.cz/services/udpipe/>
5. M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer.. Deep contextualized word representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2018, № 1. С. 2227–2237.
6. Scikit learn для дерев рішень. Accessed: 06.04.2024. <https://scikit-learn.org/stable/modules/tree.html>
7. S. Tangirala. Evaluating the Impact of GINI Index and Information Gain on Classification using Decision Tree Classifier Algorithm, International Journal of Advanced Computer Science and Applications, 2020, pp. 612-619.
8. S. Telenyk, S. Pogorilyy, A. Kramov. The complex method of coreferent clusters detection based on a BiLSTM neural network, Knowledge Based Systems. 2021. С. 205-210
9. M. Krichen et al., Are Formal Methods Applicable To Machine Learning And Artificial Intelligence? Proceedings of 2nd International Conference of Smart Systems and Emerging Technologies (SMARTTECH), 2022, pp. 48-53.

References

1. S. Pogorilyy, P. Biletskyi, Coreference resolution algorithm for Ukrainian-language texts using decision trees, Problems in programming, 2022, №3-4: pp. 85-91.
2. BOYKO Yu.V., KRYVYI S.L., POGORILYY S.D. et al (2016) Methods and innovative approaches to designing, managing, and deploying high-performant IT infrastructures. PPC “Kyiv University”, 447p. [in Ukrainian]
3. KRYVYI S.L. , POGORILYY S.D., SLYNKO M.S., KRAMOV A.A. (2020) Method of semantic application verification in GPGPU technology.

- System Research & Information Technologies № 3, pp. 7-22
4. UDpipe library. Accessed: 06.04.2024. <https://lindat.mff.cuni.cz/services/udpipe/>
 5. M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer.. Deep contextualized word representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2018, № 1. С. 2227–2237.
 6. Scikit learn for decision trees. Accessed: 06.04.2024. <https://scikit-learn.org/stable/modules/tree.html>
 7. S. Tangirala. Evaluating the Impact of GINI Index and Information Gain on Classification using Decision Tree Classifier Algorithm, International Journal of Advanced Computer Science and Applications, 2020, pp. 612-619.
 8. S. Telenyk, S. Pogorilyy, A. Kramov. The complex method of coreferent clusters detection based on a BiLSTM neural network, Knowledge Based Systems. 2021. С. 205-210.
 9. KRICHEN M. et al (2022) Are Formal Methods Applicable To Machine Learning And Artificial Intelligence? In Proceedings of 2nd International Conference of Smart Systems and Emerging Technologies (SMARTTECH), pp. 48-53

Одержано: 10.04.2024

Внутрішня рецензія отримана: 19.04.2024

Зовнішня рецензія отримана: 25.04.2024

Про авторів:

¹Погорілий Сергій Дем'янович,
доктор технічних наук,
професор,
sdp77@i.ua ,
<https://orcid.org/0000-0002-6497-5056>

Слинько Максим Сергійович,
доктор філософії,
maxim.slinko@gmail.com ,
<https://orcid.org/0000-0001-9667-8729>

¹Білецький Павло Володимирович,
аспірант,
асистент,
1234bpv@i.ua ,
<https://orcid.org/0000-0001-5425-3706>

Місце роботи авторів:

¹Факультет радіофізики,
електроніки та комп'ютерних систем,
Київський національний університет
імені Тараса Шевченка,
03187, м. Київ, проспект
Академіка Глушкова, 4г
Тел.: (38)(044) 521-05-32
E-mail: tex@knu.ua