ПРОГРАМНИЙ МОДУЛЬ ЗАХИСТУ ІНФОРМАЦІЇ В ХМАРНИХ ОБЧИСЛЕННЯХ НА ОСНОВІ МАШИННОГО НАВЧАННЯ

Величківський Ілля Олеговч
здобувач вищої освіти освітнього ступеня «Магістр», 2 курс
Національний авіаційний університет
м. Київ, Україна
ilya.velichkovskii@gmail.com

Ільєнко Анна Вадимівна,
к.т.н., доцент

Галата Лілія Павлівна
Доктор філософі

Вступ. / Introductions. У сучасну епоху цифрових технологій забезпечення безпеки даних у хмарних обчисленнях стає критично важливим завданням. З розвитком обсягу даних та збільшенням кількості кіберзагроз традиційні методи захисту вже не можуть повністю забезпечити належний рівень безпеки. У цьому контексті використання методів машинного навчання, зокрема навчання з підкріпленням (Reinforcement Learning, RL), є перспективним підходом. RL дозволяє створювати інтелектуальні системи захисту, які здатні адаптуватися до нових загроз і самостійно навчатися оптимальним діям на основі аналізу поведінки атак та дій користувачів. Це дозволяє не тільки своєчасно виявляти аномалії, але й ефективно реагувати на них, запобігаючи несанкціонованому доступу до даних у хмарних середовищах. Впровадження таких технологій може значно підвищити рівень кібербезпеки, забезпечуючи надійний захист інформаційних систем у складних і мінливих умовах.

Мета роботи. / Aim. Детальний огляд сучасних технологій захисту даних у хмарних обчисленнях. Дослідити існуючі можливості застосування алгоритмів машинного навчання для захисту інформації, розглянути їх переваги та недоліки, та провести порівняльний аналіз.

Матеріали та методи./Materials and methods. Навчання з підкріпленням (Reinforcement Learning, RL) є одним із найважливіших підходів у машинному навчанні. У цьому підході агент (суб'єкт, який приймає рішення) взаємодіє з середовищем, виконуючи певні дії, і отримує винагороду (або штраф) за кожну дію. Метою агента є максимізація накопиченої винагороди, знаходячи оптимальну стратегію дій у середовищі. Цей тип навчання використовується в ситуаціях, де складно побудувати явну модель середовища або визначити правильні дії заздалегідь. Найбільш популярними алгоритмами навчання з підкріпленням є Q-Learning та Deep Q-Learning.

1. Q-Learning є одним із популярних методів навчання з підкріпленням. Це модель, що дозволяє агенту вивчати оптимальну стратегію шляхом взаємодії із середовищем і отримання винагороди за кожну дію. Агент намагається максимізувати сумарну винагороду, поступово визначаючи найкращі дії в кожній ситуації. Однак Q-Learning потребує багато часу для навчання в складних середовищах, і його ефективність знижується у випадках із великою кількістю станів або дій. Крім того, якщо модель недостатньо навчена, вона може виконувати субоптимальні дії, що призводить до низької ефективності та неправильних висновків.
2. Deep Q-Learning — це розширення класичного Q-Learning, яке використовує нейронну мережу для оцінки значень Q-функції, що дозволяє ефективно працювати з великими й складними середовищами. Нейронна мережа допомагає обробляти дані з багатовимірними вхідними даними, що робить цей метод ефективним для задач з високою розмірністю станів, таких як відеоігри або робототехніка. Проте, Deep Q-Learning може бути

нестабільним і потребує великих обчислювальних ресурсів для навчання. Крім того, він вразливий до переобучення та може здійснювати некоректні дії, якщо навчальні дані є неповними або нерепрезентативними.

Ключові переваги навчання з підкріпленням:

1. Адаптивне прийняття рішень. Системи, побудовані на основі навчання з підкріпленням, можуть навчатися на основі власного досвіду взаємодії зі середовищем, коригуючи свою стратегію для досягнення максимальної винагороди. Це дозволяє створювати моделі, здатні приймати оптимальні рішення навіть у невідомих або динамічних умовах.
2. Можливість роботи у невизначених середовищах. Навчання з підкріпленням особливо корисне у випадках, коли середовище є складним, змінюваним або не має повної інформації про стани й дії. Алгоритми RL дозволяють агентам адаптувати свою поведінку без повного знання правил або моделей середовища.
3. Ефективне використання досвіду. Агент зберігає інформацію про свій попередній досвід (історію дій та отриманих винагород), що дозволяє йому покращувати свої стратегії з часом, враховуючи попередні помилки і успіхи. Це призводить до поступового вдосконалення поведінки й підвищення ефективності.
4. Вирішення складних послідовних завдань. RL ефективний для задач, де необхідно приймати серії рішень, кожне з яких впливає на кінцевий результат (наприклад, у робототехніці, управлінні або іграх). Агент може навчитися планувати свої дії, щоб досягти довгострокових цілей.
5. Гнучкість і генералізація. Алгоритми RL можуть використовуватися для вирішення різних типів завдань без

значних змін у своїй структурі. Вони здатні до узагальнення знань і можуть застосовувати накопичений досвід для вирішення нових, схожих завдань.

Результати та обговорення./Results and discussion.

Таблиця 1. Зведена таблиця порівняння алгоритмів машинного навчання для захисту інформації

Особливість	**Q-Learning**	**Deep Q-Learning**
Представлення станів	Використовує таблицю для збереження всіх значень Q для кожного стану і дії. Підходить для малих і простих середовищ.	Використовує нейронну мережу для представлення станів, що дозволяє працювати з великими та складними середовищами.
Масштабованість	Обмежена масштабованість: не підходить для великих середовищ через експоненційне зростання таблиці станів.	Висока масштабованість: завдяки нейронним мережам здатний працювати з великими і складними середовищами.
Час навчання	Повільне навчання в середовищах з великою кількістю станів і дій.	Навчання може бути швидшим завдяки використанню нейронних мереж, але потребує більше обчислювальних ресурсів.
Потреба у пам'яті	Вимагає великої пам'яті для зберігання таблиці Q, особливо в середовищах з великою кількістю станів.	Менша потреба в пам'яті порівняно з Q-Learning, оскільки нейронна мережа узагальнює значення Q-функції.
Обчислювальна складність	Невелика обчислювальна складність для простих задач. Може легко впоратися зі завданнями невеликих розмірів.	Висока обчислювальна складність через використання нейронних мереж, що потребує GPU для ефективного навчання.
Узагальнення	Не здатний до узагальнення: кожен стан має окреме значення Q.	Вміє узагальнювати: здатний використовувати набуті знання для нових, схожих станів завдяки навчанню нейронної мережі.
Використання у складних середовищах	Ефективний для простих середовищ з обмеженою кількістю станів і дій.	Здатний працювати у складних середовищах, де багато можливих станів і дій (ігри, робототехніка).

Висновки./Conclusions. Підсумовуючи можна зробити висновок, що розробка та реалізація програмного модуля захисту інформації в хмарних обчисленнях на основі методів навчання з підкріпленням відкриває нові горизонти у сфері кібербезпеки. Використання алгоритмів RL для динамічного виявлення загроз та адаптивного реагування на них дозволяє створювати системи, здатні ефективно захищати дані в умовах складних і мінливих середовищ. Завдяки здатності агентів самостійно навчатися оптимальним діям і стратегічно реагувати на атаки, такі системи забезпечують більш високий рівень безпеки і надійності.

Проте, як і будь-які інші технології, використання навчання з підкріпленням у хмарних обчисленнях має свої виклики. Зокрема, значні обчислювальні ресурси, необхідні для навчання та роботи агентів, можуть створювати додаткові навантаження на інфраструктуру, особливо в реальних умовах. Крім того, нестабільність моделей RL і ризик переобучення можуть призвести до неправильної оцінки загроз або неефективної поведінки у невідомих ситуаціях.

Оскільки обсяги та складність кіберзагроз постійно зростають, необхідність у розробці більш досконалих і гнучких рішень стає все більш актуальною. Програмні модулі на основі навчання з підкріпленням є перспективним напрямком для забезпечення безпеки даних у хмарних обчисленнях, що дозволяє не тільки ефективно виявляти і реагувати на загрози, а й прогнозувати потенційні атаки на основі аналізу поведінки зловмисників.