УДК 004.8:044.89

А. А. Яровий, д-р техн. наук, проф.,
О. Н. Романюк, д-р техн. наук, проф.,
І. Р. Арсенюк, канд. техн. наук, доц.,
Т. Д. Польгуль, аспірантка
Вінницький національний технічний університет, м. Вінниця, Україна
a.yarovyy@vntu.edu.ua

Виявлення шахрайства при інсталюванні програмних додатків з використанням інтелектуального аналізу даних

У роботі запропоновано систему виявлення шахрайства при інсталюванні програмних додатків з використанням інтелектуального аналізу даних на основі здійсненого аналізу шахрайських способів інсталювання програмних додатків та сучасних методів їх виявлення.
Ключові слова: виявлення шахрайства, кліковий спам, мобільне викрадення, ферми дій, виявлення аномалій, методи колаборативної фільтрації, багатовимірне шкалювання.

DOI: 10.31474/1996-1588-2017-2-25-126-131

Вступ

Дана тема є досить актуальною для компаній, які розробляють програмні додатки, адже для просування цих програмних додатків на IT-ринку, вони повинні вносити достатньо великі кошти. Одним із способів просування програмних додатків є маркетингові кампанії. Один із варіантів визначення дієвості маркетингової кампанії полягає у перевірці приведеної нею кількості інсталювань програмного додатку. Саме на цьому кроці варто знати, що частина або вся множина інсталювань програмних додатків могла бути здійснена шахрайським способом. Знаючи дійсну кількість органічних інсталювань програмного додатку та кількість інсталювань, здійснених шахрайським способом, можна визначити реальну вартість маркетингової кампанії, і чи виправдовує вона себе у фінансовому аспекті.

Метою дослідження є удосконалення процесу визначення схожості користувачів для виявлення шахрайства при інсталюванні програмних додатків.

Задачами дослідження є: здійснення класифікаційного аналізу аномалій; здійснення аналізу шахрайських способів інсталювання програмних додатків; розроблення системи виявлення шахрайства при інсталюванні програмних додатків з використанням інтелектуального аналізу даних.

Поняття аномалій та їх класифікація

Оскільки задача виявлення шахрайства відноситься до класу задач виявлення аномалій, доцільно уточнити поняття аномалії та проаналізувати критерії їх класифікації.

Аномалія – це шаблон даних, який не відповідає визначеному поняттю нормальної поведінки [1]. Розглянемо критерії, за якими можна класифікувати аномалії (доцільно зауважити, що ці критерії слід врахувати у процесі розробки системи виявлення шахрайства при інсталюванні програмних додатків):

1. *Характер вхідних даних* (Nature of Data).

Ключовим аспектом будь-якого методу виявлення аномалій є характер (природа) вхідних даних. Вхід, як правило, це сукупність екземплярів даних (об'єктів, записів, точок, векторів, шаблонів, подій, випадків, зразків, спостережень, організацій) [1]. Кожен екземпляр даних може бути описано за допомогою набору атрибутів (змінних, характеристик, функцій, полів, розмірів). Атрибути можуть бути різних типів, таких як бінарні, категорійні або безперервні. Кожен екземпляр даних може складатися тільки з одного (одновимірний) або кількох атрибутів (багатовимірний). У разі багатовимірних екземплярів даних, усі атрибути можуть бути або одного типу, або сумішшю різних типів даних. Характер (природа) атрибутів визначає придатність методів виявлення аномалій, зокрема шахрайства, у конкретній задачі [1].

2. *Тип аномалії* (Anomaly Type).

Важливим аспектом методів виявлення аномалій є тип аномалії. Аномалії можуть бути класифіковані так [1]:

- *точкові аномалії* (point anomalies) мають місце, якщо окремий екземпляр даних можна розглядати як аномальний по відношенню до решти даних;
- *контекстні (умовні) аномалії* (contextual anomalies) мають місце, якщо екземпляр даних є аномальним у конкретному контексті (але не інакше);
- *колективні аномалії* (collective anomalies) мають місце, якщо набір пов'язаних примірників даних є аномальним по відношенню до всього набору даних.

3. *Мітки даних* (Data Labels).

Мітки, пов'язані з екземпляром даних, позначають цей екземпляр нормальним або аномальним. Слід зазначити, що отримання таких міток часто обходиться занадто дорого. Виставлення міток часто робиться вручну за допомогою людини-експерта, а отже, вимагає значних зусиль, щоб отримати помічений набір навчальних даних. Як правило, отримання поміченого набору аномальних примірників даних, які охоплюють усі можливі типи аномальної поведінки є більш складним, ніж отримання мітки для нормальної поведінки. Крім того, аномальна поведінка часто має динамічний характер, наприклад, можуть виникнути нові типи аномалій, для яких немає даних, помічених під час навчання. На підставі того, наскільки мітки доступні, методи виявлення аномалій можуть працювати в одному з таких трьох режимів [1]:

- *контрольоване виявлення аномалій* (виявлення аномалій з учителем – supervised anomaly detection). Методи контрольованого виявлення аномалій припускають наявність великої кількості навчальних даних, які мають екземпляри для нормального класу, а також класу аномалій. У межах розв'язання задачі виявлення шахрайства при інсталюванні програмних додатків цей режим виявлення аномалій не є ефективним, оскільки шахрайські способи можуть з'являтися кожного дня, і неможливо створити базу з навчальними даними, яка б покривала усі випадки;
- *напівконтрольоване виявлення аномалій* (semi-supervised anomaly detection). Методи, які працюють у даному режимі, припускають, що тренувальні дані позначають примірники тільки для нормального класу. Оскільки вони не вимагають мітки для класу аномалій, та мають більш широке застосування, ніж контрольовані методи;
- *виявлення аномалій без нагляду* (некотрольовані алгоритми – unsupervised anomaly detection). Методи, які працюють у неконтрольованому режимі, не вимагають підготовки даних, і, таким чином, найбільш широко застосовуються. Методи у цій категорії роблять неявне припущення, що нормальні екземпляри зустрічаються набагато частіше, ніж аномальні екземпляри у тестових даних. Якщо це припущення не правильне, то такі методи характеризуються високим рівнем помилкових тривог. Багато напівконтрольованих методів може бути пристосовано для роботи у неконтрольованому режимі, використовуючи зразок немміченого набору даних у якості навчальних даних. Така адаптація передбачає, що дані випробувань містять дуже мало аномалій і модель, отримана у ході навчання, є стійкою до цих кількох аномалій.

4. *Вихід виявлення аномалій* (Output of Anomaly Detection).

Важливим аспектом для будь-якого методу виявлення аномалій є спосіб, яким повідомляється про аномалії. Як правило, кінцевими сигналами методів виявлення аномалій є один з таких двох типів [1]:

- *рейтинг* (scores) – методи рейтингування призначають аномальний результат кожному примірнику в тестових даних залежно від того, в якій мірі цей екземпляр вважається аномалією. Отже, виходом у таких методах є ранжований список аномалій. Аналітик може вибрати або проаналізувати кілька верхніх аномалій або використати поріг відсікання для вибору аномалії;
- *мітки* (labels) – методи у цій категорії прикріплюють мітку (нормальний або аномальний) для кожного екземпляру тесту.

Методи рейтингування для виявлення аномалій дозволяють аналітику використовувати певний поріг, щоб вибрати найзначущі аномалії. Методи, які забезпечують бінарні мітки для тестових екземплярів, безпосередньо не дозволяють аналітикам зробити такий вибір, хоча це можна контролювати додатково через вибір параметрів у рамках кожного методу [1].

Оскільки у задачі виявлення шахрайства при інсталюванні програмних додатків має місце робота з Big Data, то доцільно обрати метод, який би ранжував вихідну інформацію.

З вищенаведеного аналізу аномалій, до яких відноситься і шахрайство, можна зробити висновок, що для розроблення системи виявлення шахрайства при інсталюванні програмних додатків, слід визначити природу існуючих та можливих шахрайських способів інсталювання додатків.

Аналіз шахрайських способів інсталювання програмних додатків

Можна виділити такі шахрайські способи під час інсталювання програмних додатків:

1. *Кліковий спам* (Click Spamming) [2-4]. Цей спосіб відноситься до програм, які генерують підроблені запити кліків програмним способом. Тут шахрай спочатку запускає тести, щоб отримати URL-маркери (URL tokens) відповідної маркетингової кампанії. Після цього на кожній ітерації виконуються програми для генерації великого обсягу запитів клікання, рандомізації ідентифікаторів пристроїв (device-IDs) і програмних агентів (user agents). Основна задача полягає в тому, щоб заробити кошти від цих запитів і, на додаток, випадковим чином згенерувати потенційно-органічні установки. Типові наслідки цього сценарію: наявність величезних обсягів кліків, низький CVRs (conversion rate – відношення кількості відвідувачів сайту, що виконали на ньому певні цільові дії до загального числа відвідувачів сайту

у відсотках), велика кількість інсталювань програмних додатків і великий трафік, який надійшов через декілька точок мережі чи сайтів. Варто зауважити, що click-спаму в інтернеті, де неетичні видавці використовують шкідливі або обманні шляхи, зараз дуже багато.

2. *Мобільне викрадення* (Mobile Hijacking) [2] відбувається, коли справжній програмний додаток для користувача виконує деякі несанкціоновані дії, наприклад, запускає приховані оголошення і формує кліки від імені користувача у фоновому режимі. У цілому, програма буде працювати за сценарієм, що максимально імітує поведінку людини. Проблема полягає в тому, що усі загальні точки даних (IP-адреса і програмний агент, а також ідентифікатор пристрою) виглядають дійсними. Основна задача полягає в тому, що потенційно-органічні встановлення від цього користувача будуть неправильно прив'язані до видавця. Цей сценарій буде генерувати велику кількість інсталювань програмних додатків через кліки і покази та характеризуватиметься низькими CVRs і кращою за середню активністю користувачів. Крім того, з огляду на те, що шахрай не може контролювати, коли установка органічно проводиться користувачем, аналіз проміжку часу від кліка до установки призведе до дуже атипових розподілів на рівні користувача (видавця – publisher або IP-адреси).

3. *Ферми дій* (Action Farms) [2]. Шахраї винагороджують людей по всьому світу за установлення програмних додатків у ручному режимі, тобто фактично відбувається найняття людей для того, щоб вони встановлювали програмні додатки. Кількість подій, яку буде досягнено за цією схемою, безумовно, не буде настільки ж великою, як у програмних схемах, але у фінансовому аспекті ця схема набагато дорожча, ніж попередні.

Серед сучасних та найбільш очевидних контрзаходів до шахрайських способів інсталювання програмних додатків, які вже стали своєрідним стандартом в усій галузі, можна виділити: фільтрацію IP-адреси, блокування видавця, зовнішню фільтрацію кліків, визначення частоти кліків або запитів інсталювання, визначення співвідношення населення по геолокації (шляхом порівняння дійсної кількості населення у вказаній геолокації з кількістю інсталювань з цієї геолокації), аналіз проміжку часу між подіями (використовують, наприклад, такі відомі фірми як Adjust та Kochava [2]).

Визначення шахрайства при інсталюванні програмних додатків

Зважаючи на вищевказані шахрайські способи інсталювання програмних додатків, можна зробити висновок, що події, виконані під час шахрайства, мають спільні ознаки. Так, наприклад, цілком очевидно, що технічне забезпечення працює за певним алгоритмом, який має багато рандомізованих даних, але хід дій таких алгоритмів має закономірність (в основному, маркетингові кампанії, які гарантують проведення будь-якої кількості інсталювань програмного додатку за будь-який період часу, використовують один з вищезазначених способів шахрайства). Співробітники ферм дій здійснюють інсталювання величезної кількості програмних додатків теж за певним алгоритмом, оскільки не придумують унікальний хід дій для кожної з тисяч інсталювань.

Аналізуючи вищенаведену інформацію можна зробити припущення, що користувачі, залучені шахрайським способом, повинні мати певні спільні ознаки; аналогічно, користувачі, які створені при органічному інсталюванні програмного додатку також повинні мати ряд спільних ознак. Базуючись на цьому припущенні, можна зазначити, що використовуючи методи кластеризації, до одного з кластерів можна віднести користувачів, створених при органічному інсталюванні, а до другого – користувачів, приведених шахрайським способом. Але дуже важливим питанням при цьому є правильний вибір ознак, за якими слід здійснювати кластеризацію. Розбивши дані по кластерах та візуалізувавши результати, буде видно і користувачів, створених за допомогою одного із шахрайських способів, і користувачів, що були створені при органічному інсталюванні.

Проаналізувавши відомі методи кластеризації та класифікації [5 – 15], у даній роботі запропоновано підхід для визначення схожості користувачів, що базується на модифікованому методі колаборативної фільтрації [9] та розв'язує багатокритеріальну задачу визначення схожості користувачів. Слід зазначити, що методи колаборативної фільтрації є достатньо дієвими для задач пошуку схожості користувачів (зокрема, вони використовуються у рекомендаційних системах таких великих компаній як Netflix, Amazon тощо).
Розглянемо структурну схему системи визначення схожості користувачів у межах задачі виявлення шахрайства при інсталюванні програмних додатків. Дана система складається з трьох модулів: модуля збору інформації, модуля визначення схожості дій користувачів та модуля визначення схожості користувачів (рис. 1).

Модуль збору інформації отримує на вхід необроблену («сиру») інформацію (raw data), після чого структурує та розподіляє отримані дані у базу даних користувачів і базу даних дій користувачів та зберігає їх у структурованому вигляді. Цей модуль обробляє такі вхідні дані системи: дії користувачів під час та після інсталювання програмного додатку, геолокація по кожній дії кожного користувача, час дії, IP-адреса, операційна система пристрою тощо.
Модуль визначення схожості дій користувачів оперує набором інформації по кожній з них, тобто

векторами з множинами значень по кожній з ознак. Відомо, що для визначення схожості дій користувачів, маючи набір зібраної інформації по кожному з них, використовуються різні коефіцієнти схожості: коефіцієнт косинусної схожості між двома векторами (приймає на вхід лише бінарні дані), коефіцієнт Танімото, коефіцієнт кореляції Пірсона (коригує знецінювання оцінок, але у поставленій задачі цього робити не потрібно), евклідова відстань тощо [9]. Кожен з цих коефіцієнтів не дає бажаного результату для поставленої задачі виявлення шахрайства при інсталюванні програмних додатків.

Рисунок 1 – Структурна схема системи визначення схожості користувачів

У даному дослідженні запропоновано використання комбінованої метрики схожості дій користувачів, яка формується на основі коефіцієнта косинусної схожості між двома векторами (1) та коефіцієнта Танімото (2) [14].

$$kc = \cos(a,b) = \frac{(a \cdot b)}{|a| \cdot |b|}, \quad (1)$$

де a, b – вектори, елементами яких є частоти появи окремих ознак у заданому наборі інформації.

$$kt = T(a,b) = \frac{Nc}{Na + Nb - Nc}, \quad (2)$$

де Na – кількість елементів у наборі даних користувача a, Nb – кількість елементів у наборі даних користувача b, Nc – кількість елементів у їх перетині.

Коефіцієнт комбінованої метрики схожості дій користувачів дозволить знаходити схожість між усіма парами користувачів. Зазначимо, що покращення визначення коефіцієнта схожості є досить актуальним питанням, що підлягає вирішенню і на міжнародних IT-конкурсах, зокрема у конкурсі Netflix Prize [16].

Модуль визначення схожості користувачів отримує вектор із зібраною та структурованою інформацією у двовимірному просторі та надсилає на вихід відсортований вектор зі схожими користувачами.

Ще одним дуже важливим питанням є те, як визначати схожість користувачів, маючи набір різнорідних ознак, серед яких є і геолокація, і час виконання різних дій користувачем, і IP-адреса тощо, та підібрати для кожної з ознак певний коефіцієнт значимості. Для вирішення цього питання пропонуються такі кроки:

- оскільки набір даних містить як числові, так і дискретні дані, слід дискретні дані перевести у числові;

- необхідно використати багатовимірне шкалювання [9], що застосовується саме для того, щоб зрозуміти, як різнорідні дані пов'язані між собою [9]. Алгоритм створює подання набору даних у просторі меншої розмірності, намагаючись, по можливості, зберегти вихідні відстані між елементами. Наприклад, якщо мова йде про подання на екрані або на папері, то багатовимірний набір представляється у двовимірному просторі [9].

У ході дослідження шахрайства при інсталюванні програмних додатків та самого процесу інсталювання програмних додатків було розроблено програмний додаток, на якому проводились експерименти. На даний момент програмний додаток доступний під iOS, Android та PlayPhone у багатьох країнах світу. З додатку зібрано дані по кожному з користувачів, а саме: IP-адреса програмного пристрою, геолокація, тип операційної системи, усі дії користувача, зроблені під час інсталювання програмного додатку та під час знаходження користувача у програмному додатку (серед них історія покупок, геолокація, час та назва виконаних дій тощо).

На основі результатів роботи та аналізу розробленого програмного додатку по кожному користувачу, можна зробити висновок, що користувачі, які приведені одним із способів шахрайства, через певний період часу взагалі не мають дій всередині програмного додатку (in-app events), а відразу ж після встановлення додатку виконують дуже схожу послідовність дій. Також можна зробити висновок, що події, які створені при шахрайстві, мають спільні ознаки, що підтверджує зроблене припущення про використання кластеризації для розв'язання поставленої задачі.

Висновок

Отже, у даній роботі проаналізовано критерії, від яких залежить вибір методу для виявлення аномалій (шахрайства), проаналізовано шахрайські способи при інсталюванні програмних додатків. На основі проаналізованої інформації зроблено припущення, що створені шахрайським способом користувачі мають спільні ознаки, так само як і користувачі, що створені при

органічному інсталюванні програмного додатку. Базуючись на цьому припущенні, запропоновано та розроблено систему виявлення шахрайства при інсталюванні програмних додатків з використанням інтелектуального аналізу даних. Запропоновано один із варіантів удосконалення процесу визначення схожості користувачів на основі методу колаборативної фільтрації для знаходження користувачів, створених при шахрайських та при органічних інсталюваннях програмних додатків. В результаті дослідження шахрайства при інсталюванні програмних додатків та самого процесу інсталювання програмних додатків було розроблено програмний додаток, на якому проводились експерименти, що довели правильність зробленого припущення про доцільність використання методу колаборативної фільтрації для виявлення шахрайства при інсталюванні програмних додатків.

Список літератури

1. Varun Chandola. Anomaly Detection : A Survey / Varun Chandola, Arindam Banerjee, Vipin Kumar [Електронний ресурс]. – Режим доступу: http://cucis.ece.northwestern.edu/projects/DMS/publications/AnomalyDetection.pdf
2. Our take on mobile fraud detection [Електронний ресурс]. – Режим доступу: http://geeks.jampp.com/data-science/mobile-fraud/
3. Vacha Dave. ViceROI: Catching Click-Spam in Search Ad Networks / Vacha Dave, Saikat Guha, Yin Zhang [Електронний ресурс]. – Режим доступу: http://www.sysnet.ucsd.edu/~vacha/ccs13.pdf
4. Dave V. Measuring and Fingerprinting Click-Spam in Ad Networks. In Proceedings of the Annual Conference of the ACM Special Interest Group on Data Communication (SIGCOMM) / Dave, V., Guha, S., Zhang Y. – Helsinki, Finland, Aug. 2012. – 175 – 186 pp.
5. MachineLearning.ru [Електронний ресурс]. – Режим доступу: http://www.machinelearning.ru
6. Agarwal D. An empirical bayes approach to detect anomalies in dynamic multidimensional arrays. In Proceedings of the 5th IEEE International Conference on Data Mining. – Washington, DC, USA: IEEE Computer Society, 2005. – 26–33 pp.
7. Agarwal D. Detecting anomalies in cross-classified streams: a bayesian approach. – Knowledge and Information Systems 11 (1), 2006. – 29–44 pp.
8. Agrawal R. Mining sequential patterns. In Proceedings of the 11th International Conference on Data Engineering. / Agrawal R., Srikant R. – Washington, DC, USA: IEEE Computer Society, 1995. – 3–14 pp.
9. Сегаран Т. Программируем коллективный разум. / Т. Сегаран; пер. с англ. А. Слинкина – СПб: Символ-Плюс, 2008. – 368 с., ил. – ISBN 5-93286-119-3.
10. Кюльян А. Г. / Математична модель рекомендаційного сервісу на основі методі колаборативної фільтрації. / Кюльян А. Г., Польгуль Т. Д., Хазін М. Б. [Електронний ресурс]. – Режим доступу: http://ir.lib.vntu.edu.ua/bitstream/handle/123456789/7911/226-227.pdf?sequence=1&isAllowed=y
11. Alexander T. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions – IEEE Trans. On Knowledge and Data Engineering, vol. 17, no. 6, June 2005 – 734-749 pp.
12. Linden G. D. Collaborative Recommendations Using Item-to-Item Similarity Mappings / G. D. Linden, J. A. Jacobi, E. A. Benson – Washington, D.C.: US Patent 6,266,649 (to Amazon.com), Patentand Trade-mark Office, 2001.
13. Sarwar B. M. Item-Based Collaborative Filtering Recommendation Algorithms – 10th Int'l World Wide Web Conference, ACM Press, 2001 – 285-295 pp.
14. Vacha Dave / ViceROI: Catching Click-Spam in Search Ad Networks / Vacha Dave, Saikat Guha, Yin Zhang [Електронний ресурс]. – Режим доступу: http://www.sysnet.ucsd.edu/~vacha/ccs13.pdf
15. Aleskerov, E. Cardwatch: A neural network based database mining system for credit card fraud detection / Aleskerov, E., Freisleben, B., and Rao, B. – In Proceedings of IEEE Computational Intelligence for Financial Engineering, 1997 – 220 – 226 pp.
16. Netflix Prize [Електронний ресурс]. – Режим доступу: http://www.netflixprize.com/

References

1. Varun Chandola. Anomaly Detection : A Survey / Varun Chandola, Arindam Banerjee, Vipin Kumar, available at: http://cucis.ece.northwestern.edu/projects/DMS/publications/AnomalyDetection.pdf
2. Our take on mobile fraud detection, available at: http://geeks.jampp.com/data-science/mobile-fraud/
3. Vacha Dave. ViceROI: Catching Click-Spam in Search Ad Networks / Vacha Dave, Saikat Guha, Yin Zhang, available at: http://www.sysnet.ucsd.edu/~vacha/ccs13.pdf
4. Dave, V.б Guha, S., Zhang Y. (2012) Measuring and Fingerprinting Click-Spam in Ad Networks. In Proceedings of the Annual Conference of the ACM Special Interest Group on Data Communication (SIGCOMM), Helsinki, Finland, pp. 175 – 186.

5. MachineLearning.ru, available at: http://www.machinelearning.ru
6. Agarwal, D. (2005) An empirical bayes approach to detect anomalies in dynamic multidimensional arrays. In Proceedings of the 5th IEEE International Conference on Data Mining, Washington, DC, USA: IEEE Computer Society, pp. 26–33.
7. Agarwal, D. (2006) Detecting anomalies in cross-classified streams: a bayesian approach. Knowledge and Information Systems, No. 11 (1), pp. 29–44.
8. Agrawal, R., Srikant, R. (1995) Mining sequential patterns. In Proceedings of the 11th International Conference on Data Engineering. Washington, DC, USA: IEEE Computer Society, pp. 3–14.
9. Segaran, T. (2008) Programming of collective mind [Programmiruem kollektivniy razum], Sankt-Peterbourg, Simvol-Plus.
10. Kyulian, A/, Polgul, N., Khazin, M. Mathematical model of recommendation service based on collaborative filtering method [Matematychna model′ rekomendatsiynoho servisu na osnovi metodi kolaboratyvnoyi fil′tratsiyi], available at: http://ir.lib.vntu.edu.ua/bitstream/handle/123456789/7911/226-227.pdf?sequence=1&isAllowed=y
11. Alexander, T. (2005) Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions, IEEE Trans. On Knowledge and Data Engineering, vol. 17, no. 6, pp. 734-749.
12. Linden, G. D., Jacobi, J. A., Benson, E. A. (2001). Collaborative Recommendations Using Item-to-Item Similarity Mappings, Washington, D.C.: US Patent 6,266,649 (to Amazon.com), Patentand Trade-mark Office.
13. Sarwar, B. M. (2001) Item-Based Collaborative Filtering Recommendation Algorithms. 10th Int'l World Wide Web Conference, ACM Press, pp. 285-295.
14. Vacha Dave, Saikat Guha, Yin Zhang. ViceROI: Catching Click-Spam in Search Ad Networks, available at: http://www.sysnet.ucsd.edu/~vacha/ccs13.pdf
15. Aleskerov, E., Freisleben, B., and Rao, B. (1997) Cardwatch: A neural network based database mining system for credit card fraud detection, Proceedings of IEEE Computational Intelligence for Financial Engineering, 1997 – pp. 220 – 226.
16. Netflix Prize. available at: http://www.netflixprize.com/

Надійшла до редакції 15.11.2017

А. А. ЯРОВОЙ, А. Н. РОМАНЮК, И. Р. АРСЕНЮК, Т. Д. ПОЛЬГУЛЬ
Винницкий национальный технический университет (Украина)

ВЫЯВЛЕНИЕ МОШЕННИЧЕСТВА ПРИ ИНСТАЛЛИРОВАНИЯ ПРИЛОЖЕНИЙ С ИСПОЛЬЗОВАНИЕМ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ

В работе предложена система обнаружения мошенничества при инсталлирования приложений с использованием интеллектуального анализа данных на основе проведенного анализа мошеннических способов инсталляции приложений и современных методов их обнаружения.

Ключевые слова: выявление мошенничества, кликовый спам, мобильное похищение, фермы действий, выявление аномалий, методы коллаборативной фильтрации, многомерное шкалирование.

A. A. YAROVYI, O. N. ROMANYUK, I. R. ARSENYUK, T. D. POLHUL
Vinnytsia National Technical University (Ukraine)

PROGRAM APPLICATIONS INSTALL FRAUD DETECTION USING DATA MINING

Program applications install fraud detection technique using data mining was offered. It was based on a comparative analysis of mobile install fraud techniques and detection of techniques of mobile install fraud.

Key words: fraud detection, click spamming, mobile hijacking, action farming, anomaly detection, collaborative filtering methods, multidimensional scaling.