Egérdinamika-alapú bot detektálás

Mouse Dynamics-based Bot Detection

KOVÁCS Andrea, ANTAL Margit

Sapientia EMTE, Marosvásárhelyi Kar, Marosvásárhely
kovacs.andrea@student.ms.sapientia.ro, manyi@ms.sapientia.ro

Abstract

Nowadays bot detection is gaining more and more attention due to the damage caused by bots besides the fact that they perform automated tasks. Although the presence of bots is being examined from more and more aspects to prevent attacks, it is a complex process as state-of-the-art bots are already aimed at mimicking human behavior. In our research, we examine the human-like behavior of bots in terms of different generated mouse movements. Synthetic data sets were generated based on different parameterized versions of Bézier curves, by convolutional autoencoders and also by a generative adversarial network (TimeGAN) specialized in time series. We used our Sapimouse data set, which contains mouse dynamics data for 120 users. Using this data set and the synthetic data sets, we measured the performance of different anomaly detectors. The best performing detector, which proved to be the most successful in distinguishing between generated and real human mouse movements was the Local Outlier Factor (LOF). Regarding the generated data sets, the mouse movements generated by convolutional autoencoder were the most human-like.

Keywords: bot detection, mouse movement, anomaly detector.

Kivonat

A botok detektálását egyre nagyobb figyelem övezi napjainkban, hiszen a botok az automatizált feladatok elvégzésén túl számos kárt okoznak. Habár a támadások kivédése érdekében egyre több aspektusból vizsgálják a botok jelenlétét, ez egy komplex folyamatnak számít, hiszen a legmodernebb botok már az emberi viselkedés utánzását is megcélozzák. Kutatásunkban a botok emberszerű viselkedését vizsgáljuk különböző módszerekkel generált egérmozgások szempontjából. Bézier görbék különbözően paraméterezett változataival, konvolúciós autoenkóderrel, illetve egy idősorokra szakosodott generatív ellenséges hálózat (TimeGAN) segítségével szintetikus adathalmazokat generáltunk. A mérésekhez a Sapimouse adathalmazt használtuk, amely 120 felhasználó egérdinamikai adatait tartalmazza. Ezt az adathalmazt, illetve a szintetikus adathalmazokat felhasználva mértük különböző anomália detektorok teljesítményét. A legjobban teljesítő detektor, mely a legsikeresebbnek bizonyult a generált és valós emberi egérmozgások elkülönítésében az LOF lett. A generált adathalmazok tekintetében a konvolúciós autoenkóder által generált egérmozgások voltak a legemberszerűbbek.

Kulcsszavak: bot detektálás, egérmozgás, anomália detektor.

1. BEVEZETÉS

Napjainkban egyre nagyobb figyelmet kapnak a különböző webes botok, melyeknek számos változata ismert az egyszerű automatizált szkriptektől a fejlett webbotokig. Utóbbiak rendelkeznek a böngésző ujjlenyomatával, támogatják a böngésző főbb funkcióit és képesek akár emberszerű viselkedés tanúsítására is. Ezen tulajdonságaiknak köszönhetően észlelhetőségük egyre nehezebbé válik, ugyanakkor egy fontos problémát is jelent, hiszen a webes adatforgalom jelentős részét produkálják. Mindamellett, hogy számos hasznos funkcionalitással rendelkeznek (webindexelés, weboldal figyelés, adatkinyerés kereskedelmi célokra), számottevő rosszindulatú tevékenység mögött is megfigyelhető jelenlétük (különböző hitelkártyaszámok, ajándékkártya-számok és bejelentkezéshez szükséges adatok kipróbálása, az összes rendelkezésre álló készlet megvásárlása egy adott termékből, hogy később magasabb áron értékesítsék vagy éppen fiókok létrehozása spam üzenetek küldésére) [3] .

A legmodernebb megközelítések napjainkban botok detektálására gépi tanuláson alapulnak. Amint azt a weboldalak felhasználói megszokhatták, az egyik legelterjedtebb technikája a webbotok észlelésének a CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart), melynek különböző változatai léteznek: egyszerű képfelismerési feladatok, torzított karakterek felismerése vagy egyszerű jelölőnégyzetbe való klikkelés, mely során egyéb adatot is gyűjtenek a felhasználóról, ilyen például az egérmozgás.

Dolgozatunk a webes botok detektálásának problémáját egérmozgások vizsgálata szempontjából közelíti meg. Kihívást jelent minél többféle szintetikus adathalmaz generálása, továbbá egy bot detektáló rendszer kialakítása a különböző generált egérmozgások kivédése érdekében.

Szintetikus adatok generálása számtalan megközelítés alapján történhet, melyeket szinte lehetetlen mind feltérképezni. Ilyenek például a Bézier görbén alapuló kísérletek, generatív neuronhálók, autoenkóderek vagy bármely megközelítés, melyet az emberi egérmozgásokat megfigyelve alakítanak ki. A szintetikus adatok változatosságából adódóan a detektorok a félig felügyelt tanulásnak megfelelően kizárólag emberi adatok alapján tanulnak.

A generált egérmozgások idősorok, melyekből érdemes kinyerni különböző jellemzőket a detektorok teljesítményének, illetve hatékonyságának növelése érdekében.

Az volt a célunk, hogy minél jobban feltérképezzük az emberi egérmozgások esszenciáját annak érdekében, hogy különböző stratégiákat dolgozhassunk ki emberszerű egérmozgások generálására, illetve felépítsünk egy olyan rendszert, amely nagy pontossággal el tudja különíteni egymástól az emberi egérmozgásokat a különböző megközelítések alapján generált egérmozgásoktól.

Az általunk kipróbált szintetikus adatgenerálási módszerek Bézier görbéken, autoenkódereken, illetve generatív ellenséges neuronhálókon alapulnak. Az első megközelítés, amely alapján az emberi egérmozgást próbáltuk megközelíteni, a hagyományos harmadfokú Bézier-görbe. Ennek is egy módosított változatát használtuk, mely emberszerűbb mozgássá teszi a Bézier görbét. A második megközelítés egy sajátos tanítási algoritmussal rendelkező autóenkóder, amelyet egy előző dolgozatunkban ismertettünk [2]. A harmadik megközelítés a TimeGAN, mely egy idősorok generálására szakosodott korszerű neuronháló.

A szintetikus adatok azonosítását anomália-detektorokkal végeztük. A detektorok teljesítményét az AUC (Area under the ROC curve, ROC görbe alatti terület) metrika alapján hasonlítottuk össze. A generált adat minőségét a detektorok teljesítménye határozza meg, hiszen minél sikeresebb egy detektor, annál gyengébbnek számít a generált adathalmaz. Gyengébb teljesítmény esetén pedig nem megbízható a rendszer. Egy megbízható bot detektáló rendszer kialakítása érdekében tehát egyre erősebb adathalmazokat generáltunk, majd próbáltuk ezeknek megfelelően javítani a detektorok teljesítményét.

Dolgozatunkban háromféle szintetikus adathalmazt állítottunk elő, melyek idősorokból épülnek fel. Az egérmozgások idősorként való reprezentációja nem hozott minden adathalmaz esetében jó eredményt, azonban az idősorokból kinyert jellemzők segítségével már minden esetben jó eredményeket értek el a detektorok. Természetesen a detektorok teljesítményét illetően vannak eltérések, de összességében elmondható, hogy a jellemzők alapján előállított minták esetében az AUC metrika szerint több kiemelkedő teljesítményű detektort is sikerült feltérképezni.

2. IRODALMI ÁTTEKINTÉS

Botok detektálására léteznek kognitív készségeken (karakterfelismerés, képfelismerés stb.), illetve viselkedési biometrián alapuló módszerek is. Az első kategóriába tartozó feladatok már nem jelentenek kihívást a mesterséges intelligencia tudományterületén, míg az utóbbi kategória igen. Utóbbihoz tartoznak az egérmozgásokon vagy egyéb adatok mellett egérmozgásokra is alapozó kutatások.

Wei és tsai. [5] az egérmozgások térbeli és kinematikai tulajdonságait figyelembe véve bevezetnek egy ábrázolási módszert, mely segítségével a mozgásokat képekké formálják. Bot detektálásra konvolúciós neurális hálót használtak, melynek segítségével a botok 96.2 százalékát sikerült detektálni.

Iliou és tsai. [3] egy webbot-észlelési keretrendszert javasolnak, melyben a botok észlelését két szempont alapján vizsgálják: webnaplók és egérmozgások. Ez a két szempont lehetővé teszi a behatolások idő- és térbeliségének vizsgálatát. Rendszerük hatékonyságát tesztelték olyan web-botokkal, melyek rendelkeznek a böngésző ujjlenyomattal, illetve fejlettebb web-botokkal is, melyek az emberi viselkedést is képesek utánozni. Ehhez egy webszervert használtak, amelyhez emberi látogatók és szimulált webbotok is egyaránt hozzáfértek. Bot detektálási keretrendszerük elsősorban kiszámít egy pontszámot az egérmozgás alapján, majd ha ez az érték egy előre meghatározott küszöbérték alá esik, akkor figyelembe veszi a naplózás eredményét is.

Chu és tsai. [4] egy olyan rendszert javasolnak, amely kliens oldalon naplózza a felhasználók egérmozgásait és billentyű leütési adatait, majd ezek alapján egy szerveroldali osztályozó segítségével

megállapítja, hogy a felhasználó ember vagy bot. Rendszerük előnye, hogy passzív, tehát nem igényel a felhasználótól plusz erőfeszítést, mivel a felhasználó egy szessziója alatt végzett tevékenységének monitorizálásán alapszik. E kétféle adat segítségével 99%-nál magasabb detektálási pontosságot sikerült elérniük.

Acien és tsai. [1] kidolgoztak egy *BeCAPTCHA-Mouse* nevezetű bot detektort a generált és valós egérmozgások felügyelt osztályozására. A detektor az egérmozgások neuromotoros modellezése során előállított jellemzőkkel dolgozik. A jellemzőkinyerést a SigmaLognormal modell segítségével valósítják meg, mely képes az emberi egérmozgások sebességprofilját lognormális alakkal leírni, mely kiválóan jellemzi az emberi agyban található motoros kéreg által irányított egérmozgások természetét. Az osztályozók teljesítményének mérésére heurisztikus függvények és egy generatív neuronháló segítségével állítanak elő szintetikus egérmozgásokat. Rendszerük átlagosan 93%-os pontosságot ér el egyetlen egérmozgáson alapuló detektálás során.

Antal és tsai. [2] bot detektálás elősegítése érdekében megalkottak egy SapiAgent nevezetű modellt, mely egy kiváló alternatívát nyújt emberszerű egérmozgások előállítására. A SapiAgent megvalósítása érdekében autoenkódereket és egy új tanító algoritmust alkalmaznak. Kutatásukban bizonyítják, hogy az általuk javasolt generálási módszerrel realisztikusabb egérmozgások generálhatók, mint a Bézier görbékkel vagy a hagyományos autoenkóderekkel.

3. ADATOK ÉS MÓDSZEREK

3.1. SapiMouse adathalmaz

Munkánk során rendelkezésünkre állt a SapiMouse adathalmaz, mely 120 felhasználó egérdinamikai adatait tartalmazza (https://www.ms.sapientia.ro/~manyi/sapimouse/sapimouse.html). A mérésben résztvevő személyek a Sapientia Erdélyi Magyar Tudományegyetem munkatársai és diákjai közül kerültek ki önkéntes alapon, összesen 92 férfi és 28 nő, életkorukat tekintve 18 és 53 év közöttiek. Az önkéntesek többsége jobb kezes volt, csupán 9 személy volt balkezes a 120-ból. Az adathalmaz készítői egy webes játékot hoztak létre a mérések lebonyolítása érdekében, amelyben a felhasználóknak egérmozgásokat kellett végrehajtaniuk a képernyő különböző célkoordinátái között. Az alkalmazás folyamatosan naplózta az egér pozícióját egy körülbelüli 60 Hz-s mintavételezési frekvenciával. A játék több feladatból állt, különböző geometriai formákat jelenített meg véletlenszerű pozíciókban a képernyőn (háromszög, fordított háromszög, négyzet, kör). A felhasználóknak ezen geometriai alakzatokhoz kellett mozgatniuk a kurzort, majd egy bal, jobb vagy dupla kattintást, esetenként egy drag-and-drop műveletet kellett végrehajtaniuk. Ezen műveletek alatti mérések során létrejött egy adathalmaz, mely minden feladat esetén az egérkurzor pozíciójából (x, y), gombtípusból, eseménytípusból (mozgatás, húzás, lenyomva vagy felengedve), és a megfelelő időbélyegből épült fel. Az önkéntesek saját számítógépükön hajtották végre a feladatokat és előzetesen tájékoztatva voltak arról, hogy tevékenységük naplózva lesz. Két munkamenetet kellett teljesíteniük, melyek egyike 3, másik 1 percet vett igénybe. Mindkét munkamenet során a felhasználóknak a fent említett feladatokból kellett megoldaniuk minél többet [6].

3.2. Szintetikus adathalmazok

Háromféle szintetikus adathalmazt készítettünk rendre a következő módszerekkel: Bézier görbék, TimeGAN neuronháló, illetve egy sajátos autoenkóder segítségével. A Bézier görbék, illetve az autoenkóder esetében a szintetikus adathalmaz egérmozgásait a SapiMouse 1 perces adathalmaz alapján állítottuk elő úgy, hogy a szintetikus adatok az emberi egérmozgások kezdő és végpontjai között generálódtak és ezzel megegyező számú köztes pontot tartalmaznak.

3.2.1 Bézier görbe alapú adathalmaz

A Bézier görbe egy elterjedt módszer két rögzített pont közötti görbe előállítására. Ez egy parametrikus görbe, mely kontrollpontok és egy *t* paraméter függvényében generálódik. Az első és utolsó kontrollpont rajta van a görbén, míg a többi a görbe alakulását befolyásolja. A görbék előállítására a pyclick (https://github.com/patrikoss/pyclick) csomagból a *HumanCurve* osztályt használtuk. A pyclick egy Pythonban készült csomag, amelynek segítségével Bézier görbe alapú, emberszerű egérmozgásokat lehet előállítani. Lehetőség van olyan paraméterek beállítására, amelyek a mozgás sebességét és annak gyorsulást szabályozzák.

3.2.2. TimeGAN neuronhálóval előállított adathalmaz

A generatív ellenséges neuronhálók (GAN - Generative Adversarial Network) két kompetitív hálóból tevődnek össze. Az egyik a generátor, mely random zajból állít elő szintetikus adatokat, a másik a diszkriminátor, melynek célja a valós és generált adatok megkülönböztetése. A cél az, hogy a generátor olyan adatot tudjon előállítani, melyet a diszkriminátor nem tud megkülönböztetni a valós mintáktól.

Az idősorok szekvenciális adatok, melyek fontos tulajdonsága az időbeliség. Ennek következtében a változók közötti eloszláson túl a modellnek meg kell tanulnia az adatok időbeli dinamikáját is. A TimeGAN [7] egy generatív neuronháló, mely sikeresnek bizonyult szekvenciális adatok időbeliségének elsajátításában. Ezt a szerzők a felügyelt és felügyelet nélküli tanítás ötvözésének technikájával támasztják alá. Munkánk során a Stefan Jansen féle implementációt [10] alakítottuk át igényeinknek megfelelően. Mivel a TimeGAN segítségével csak rögzített hosszúságú egérmozgásokat lehet előállítani, ezért a hossznak 32-t használtunk, mert a SapiMouse adathalmaz esetében ez volt az átlagos hossz (egérmozgás során érintett képernyőpontok száma).

3.2.3. Autoenkóderrel előállított adathalmaz

Antal és tsai. [2] egy sajátos módszerrel tanított autoenkódert javasoltak egérmozgások generálására. A módszer tulajdonképpen azt tanítja meg az autoenkódernek, hogy hogyan generáljon egy egyenes vonalból olyan törtvonalat (diszkrét görbét), amely emberszerű egérmozgáshoz hasonlít. A konvolúciós (CNN-AE) és a rekurrens (RNN-AE) architektúrájú autoenkóderek közül a konvolúciós emberszerűbb egérmozgásokat generált, ezért jelen munkánkban ezt fogjuk használni.

3.3. Jellemzőkinyerés

A bot detektálás végezhető nyers adatokból is (trajektóriák), de méréseink azt mutatták, hogy az általunk használt anomália detektorok hatékonyabban működtek amennyiben a trajektóriákat egy rögzített hosszúságú jellemzősorozattal reprezentáltuk. Legyen $T = \{(x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n)\}$ egy n pontból álló trajektória, amelyből kiszámítjuk az x, illetve y irányú elmozdulássorozatokat $Dx = \{dx_1, dx_2, \ldots dx_n\}$, $Dy = \{dy_1, dy_2, \ldots, dy_n\}$, ahol $dx_i = x_i - x_{i-1}$, $dy_i = y_i - y_{i-1}$, i=1, 2, …n.

A Dx, Dy elmozdulássorozatokból minden trajektória esetén összesen 29 jellemzőt állítottunk elő. Ezek jellegüket tekintve az alábbiak: statisztikai, hisztogram-alapú, simasággal és a mozgás hatékonyságával kapcsolatos jellemzők.

Az első csoporthoz olyan statisztikából ismert jellemzők tartoznak, melyek relevánsnak bizonyultak az egérmozgásokra vonatkozóan. Ilyenek a minimum, kvartilisek (első, harmadik), medián, maximum, módusz, ferdeség (skewness). Ezek mindegyikét kiszámítottuk a Dx és Dy esetében is, így összesen 14 jellemzővel bővült a jellemzők listája.

A hisztogram alapú jellemzők esetében 5 intervallumos felosztást használtunk mind a Dx, mind pedig a Dy esetében. Ez 10 jellemzőt eredményezett összesen. Az intervallumokat empirikusan határoztuk meg, annak érdekében, hogy a lehető legjobban elkülöníthetőek legyenek egymástól a különféle típusú mozgások.

Gianvecchio és tsai. [9] az egérmozgás hatékonyságát úgy értelmezi, mint a két végpont közti távolság és a trajektória hosszának aránya. A botok által generált trajektóriák magasabb hatékonysággal rendelkeznek, mint az emberi trajektóriák. Ezt a jellemzőt mi is felhasználtuk.

Megfigyeltük, hogy a botok által generált trajektóriák Dx és Dy elmozdulássorozatai simábbak, mint az emberi egérmozgások [2]. A simaságot kiszámítottuk mind az elsőrendű különbségekre (Dx, Dy), mind pedig a másodrendű különbségekre, így összesen 4 jellemzőt kaptunk.

3.4. Anomália-detektorok

Az anomália-detektálás több szakterület esetében is fontos és kutatott témának számít. Alkalmazzák többek között a hitelkártyákkal kapcsolatos csalások felderítésében, képfeldolgozásban, behatolás észlelésnél, kiberbiztonságban és kritikus rendszerek hiba észlelésénél is. Lényege, hogy olyan mintákat találjunk egy adathalmazban, amelyek nem az elvárt módon viselkednek, azaz különböznek az adatok többségétől.

Az anomália-detektoroknak számos típusa létezik. Megkülönböztetünk valószínűség alapúakat (pl. COPOD), közelség alapúakat (kNN, LOF), léteznek lineárisak modellek (PCA, OCSVM), valamint ötvözöttek (IForest). Az említett detektorokat a PyOD [8] szolgáltatta, mely egy Python-eszközkészlet a többváltozós

adatokban található anomáliák észlelésére.

4. MÉRÉSEK

4.1. Mérési protokoll

Anomália-detektálás során az első lépésben a detektorok megtanulják a normális viselkedést, vagyis azt, hogy az emberre milyen egérmozgás jellemző. Ezt a tudást használják fel a következő lépésben arra, hogy különböző pontszámokat társítsanak a kapott mintákhoz. Mivel az emberi egérmozgást több megközelítés alapján is lehet utánozni és nehéz feltérképezni az összes ilyen módszert, a tanítási adathalmaz kizárólag emberi egérmozgásokat tartalmaz. A modellek tesztelését mind emberi, mind pedig szintetikus mozgásokból előállított adathalmazokkal is végrehajtottuk.

A tanításhoz minden esetben a SapiMouse 3 perces munkamenetekből származó adatokat használtuk. A kiértékelés során az 1 perces munkamenetekből előállított adathalmazt használtuk, mint humán adathalmaz, valamint a háromféle generált adathalmazt felváltva, mint szintetikus adathalmaz.

A tanítási adathalmaz 18308 mintát tartalmaz, a teszteléshez használt adathalmazok mindenike pedig 5919 mintát. A teszteléshez minden egyes mérés esetében a SapiMouse 1 perces adathalmaz mintáit használtuk (5919 minta), mint emberi minták, és valamely szintetikus adathalmaz mintáit (5919 minta). Minden mérés eredménye 5919 pozitív és ugyanannyi negatív pontszám (score) lesz, amelyekből kiszámítjuk a ROC görbe alatti terület értékét (AUC).

4.2. Eredmények

A mérési eredményeket az 1. táblázat összesíti. Ahogy a mérési protokollban leírtuk, minden detektor esetében egyetlen tanítás történt csak emberi adatok használatával. Ezután három tesztelés következett, amelyben a háromféle szintetikus adat esetében kiszámítottuk az AUC értéket. Valós alkalmazások szimulálása érdekében, amelyben az egér dinamikáját hosszabb ideig rögzíthetik, a mérési eredményeket 10 egérmozgásból álló kötegekre ismertetjük.

1. táblázat Különböző bot típusok detektálási eredményei anomália-detektorokkal. A detektor teljesítménye: AUC metrika; döntés: 10 egérmozgás alapján. Minél kisebb az AUC érték, annál erősebb a bot által generált adat - annál nehezebb elkülöníteni ezeket az emberi adatoktól.

Detektor	**AUC** **Szintetikus adathalmaz (bot)**		
	Bézier (HumanLike)	**TimeGAN**	**CNN-AE**
PCA	0,89	0,77	0,67
LOF	0,96	0,84	0,73
OCSVM	0,70	0,84	0,69
IForest	0,88	0,74	0,55

A legjobban teljesítő detektor az LOF volt, amely minden egyes szintetikus adat esetében a legjobban teljesített. A generált adatok emberszerűsége tekintetében megállapíthatjuk, hogy a konvolúciós autoenkóderrel (CNN-AE) generált egérmozgásokat volt a legnehezebb elkülöníteni az emberi egérmozgásoktól, de a TimeGAN által generált adatok is sokkal emberszerűbbek, mint a módosított Bézier-rel előállított adatok.

5. KÖVETKEZTETÉSEK

Dolgozatunkban különböző stratégiákkal szintetikus egérmozgásokat állítunk elő, majd ezek emberszerűségét vizsgáljuk anomália detektorok segítségével. Háromféle szintetikus adatot állítunk elő: emberszerűesített Bézier görbe, sajátosan tanított autoenkóder, illetve a TimeGAN, amely egy idősorokra szakosodott generatív ellenséges neuronháló. A legemberszerűbb egérmozgásokat a konvolúciós autoenkóder segítségével sikerült előállítani, ezt követte a TimeGAN, majd az emberszerűesített Bézier görbék. A pyOD csomagban található anomália-detektorok közül az LOF detektor volt a leghatékonyabb mindhárom típusú szintetikus adat esetében.

IRODALMI HIVATKOZÁSOK

[1] Alejandro Acien, Aythami Morales, Julian Fierrez, Ruben Vera-Rodriguez, BeCAPTCHA-Mouse: Synthetic mouse trajectories and improved bot detection, Pattern Recognition, vol. 127, pp. 108643, 2022.

[2] M. Antal, K. Buza and N. Fejer, "SapiAgent: A Bot Based on Deep Learning to Generate Human-Like Mouse Trajectories," in *IEEE Access*, vol. 9, pp. 124396-124408, 2021.

[3] Christos Iliou, Theodoros Kostoulas, Theodora Tsikrika, Certh Vasilis Katos, Stefanos Vrochidis, Ioannis Kompatsiaris, Detection of Advanced Web Bots by Combining Web Logs with Mouse Behavioural Biometrics, ACM Journals, vol. 2 (24), pp 1 - 26, 2021.

[4] Chu, Z., Gianvecchio, S., Wang, H. (2018). Bot or Human? A Behavior-Based Online Bot Detection System. In: Samarati, P., Ray, I., Ray, I. (eds) From Database to Cyber Security. Lecture Notes in Computer Science(), vol 11170. Springer, Cham.

[5] Wei, A., Zhao, Y., Cai, Z. (2019). A Deep Learning Approach to Web Bot Detection Using Mouse Behavioral Biometrics. In: Sun, Z., He, R., Feng, J., Shan, S., Guo, Z. (eds) Biometric Recognition. CCBR 2019. Lecture Notes in Computer Science(), vol 11818. Springer, Cham.

[6] M. Antal, N. Fejér and K. Buza, "SapiMouse: Mouse Dynamics-based User Authentication Using Deep Feature Learning," *2021 IEEE 15th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, 2021, pp. 61-66.

[7] Yoon Jinsung, Jarret Daniel, van der Schaar Mihaela, Time-series Generative Adversarial Networks, Advances in Neural Information Processing Systems, vol. 32, 2019.

[8] Zhao, Y., Nasrullah, Z. and Li, Z., 2019. PyOD: A Python Toolbox for Scalable Outlier Detection. Journal of machine learning research (JMLR), 20(96), pp.1-7.

[9] Steven Gianvecchio, Zhenyu Wu, Mengjun Xie, and Haining Wang. 2009. Battle of Botcraft: fighting bots in online games with human observational proofs. In Proceedings of the 16th ACM conference on Computer and communications security (CCS '09), 256–268.

[10] https://github.com/stefan-jansen/machine-learning-for-trading/tree/main/21_gans_for_synthetic_time_series (Utolsó letöltés: 2019. 04.10).