ODKRIVANJE IZJEM NA PRIMERU INTELIGENTNEGA SISTEMA ZA KONTROLO PRISTOPA

Tea Tušar, Matjaž Gams
Odsek za inteligentne sisteme
Institut "Jožef Stefan"
Jamova 39, 1000 Ljubljana, Slovenija
tea.tusar@ijs.si, matjaz.gams@ijs.si

POVZETEK

Prispevek obravnava sistem za kontrolo pristopa, ki različne (biometrične) senzorje povezuje in nadgrajuje z uporabo inteligence. Ena izmed nalog takšnega inteligentnega sistema je odkrivanje izjem, tj. nenavadnih primerov obnašanja uporabnikov. V prispevku so opisani različni algoritmi za odkrivanje izjem, ki izjeme definirajo s pomočjo razdalj med primeri, na podlagi gostote ali pa z uporabo različnih klasifikatorjev. Na podlagi posebnih lastnosti podatkov sistema za kontrolo pristopa lahko ugotovimo, da so za ta namen najbolj primerne metode, kot so projekcije na prostore z manj dimenzijami, izračun lokalnega koeficienta izjemnosti in kombinacija obeh metod. Opravljena analiza bo postala še bolj informativna, ko bomo omenjene metode preizkusili na realnih podatkih o pristopih.

1 UVOD

Sistemi za kontrolo pristopa predstavljajo pomemben del celovitega zagotavljanja varnosti ljudi in premoženja. Pri kontroli pristopa ločujemo določanje identitete (tako imenovano *identifikacijo*) in preverjanje identitete (*verifikacijo*) posameznikov. V zadnjem času se za obe nalogi vedno pogosteje uporabljajo biometrični senzorji. Medtem ko se za identifikacijo poleg prstnih odtisov še vedno pogosto uporabljajo brezkontaktne kartice ali gesla oz. PIN kode, se za verifikacijo uporabnikov uporablja prepoznava biometričnih lastnosti človeka, kot so npr. šarenica, prstni odtis, geometrija dlani in podobno. Glavna prednost uporabe biometričnih metod je v tem, da so biometrične lastnosti človeka edinstvene in težko ponaredljive. Poleg tega so "sestavni del" človeka in jih zato uporabnik ne more izgubiti oz. pozabiti.

Poleg uporabe različnih (biometričnih) senzorjev, lahko sistem za kontrolo pristopa nadgradimo z uporabo inteligence. Inteligentni sistem za kontrolo pristopa lahko spremlja dve vrsti obnašanja:

- mikro obnašanje pred posamezno točko za nadzor vstopa, in
- makro obnašanje, tj. gibanje med različnimi točkami za nadzor vstopa.

Spremljanje mikro obnašanja temelji na predpostavki, da se uporabniki navadijo pristopati k preverjanju identitete na določen način, ki se v krajšem časovnem obdobju bistveno ne spreminja, je pa od uporabnika do uporabnika različen. Odvisen je od njegovih navad in motoričnih lastnosti (npr. od mesta, kjer ponavadi nosi identifikacijsko kartico, kateri prst je uporabil za prstni odtis, ali je motorično bolj ali manj spreten). S spremljanjem mikro obnašanja in kombiniranjem izhodov ostalih senzorjev, dobimo nov, "inteligentni virtualni senzor", ki lahko dodatno verificira uporabnika oz. nam pove, s kakšno verjetnostjo je človek pred senzorjem res tisti, za katerega se izdaja.

Po drugi strani *spremljanje makro obnašanja* pomeni opazovanje dnevne rutine uporabnikov. Pri tem se beleži kdaj in na katerih točkah uporabnik običajno vstopa. Tako se sistem lahko nauči značilnih vzorcev obnašanja za posamezne uporabnike in ugotovi, ko pride do izjem. Na primer, sistem bi moral opaziti, če se je namesto kadilca, ki vsako uro odhaja kadit, infiltriral nekadilec ali pa če kadilec zaradi živčnosti odhaja ven dvakrat pogosteje. Sistem spremlja tudi, kateri uporabniki prihajajo skupaj in druge časovne odvisnosti med njimi. Pravzaprav se sistem lahko nauči česarkoli, kar je opisljivo z zajetimi podatki. Naučeno znanje uporablja sproti za odkrivanje deviantnosti, hkrati pa so naučena pravila na voljo nadzornikom in morebitnim analitikom za kasnejši ročni pregled.

S spremljanjem obnašanja na obeh nivojih se inteligentni sistem za kontrolo pristopa lahko nauči prepoznavati ustaljene vzorce obnašanja za vsakega posameznega uporabnika in, kar je še pomembnejše, odkrivati izjeme, ki lahko predstavljajo poskus vstopa neavtoriziranе osebe.

Za odkrivanje izjem poznamo številne algoritme, ki se uspešno uporabljajo v ta namen in jih bomo predstavili v naslednjem razdelku. V nadaljevanju se bomo

posvetili tudi posebnostim podatkov inteligentnega sistema za kontrolo pristopa, ki postavljajo svoje zahteve za predstavljene algoritme. Prispevek bomo zaključili s pregledom nalog, ki nas za izvedbo takšnega sistema še čakajo.

2 ALGORITMI ZA ODKRIVANJE IZJEM

Odkrivanje izjem (primerov z nenavadnimi lastnostmi) je zelo zanimivo področje strojnega učenja, ki se uporablja pri reševanju mnogih nalog, kot so med drugim odkrivanje prevar [6], identificiranje vdorov v računalniška omrežja [13, 8] in prečiščevanje podatkov [14]. Z odkrivanjem izjem so se najprej ukvarjali v statistiki [10, 3], kjer pa so večinoma obravnavali enodimenzionalne podatke in podatke, za katere je vnaprej znana njihova distribucija. Ker za naše podatke opisani lastnosti ne veljata, v prispevku ne bomo obravnavali statističnih metod, ampak le metode, ki temeljijo na strojnem učenju.

2.1 Definicija izjeme

Ko govorimo o odkrivanju izjem, moramo najprej definirati, kaj izjema sploh je. Večina avtorjev izjemo definira s pomočjo *razdalje* do njenih najbližjih sosedov. Če torej pregledamo lokalno okolico (navadno vzamemo k najbližjih sosedov) nekega primera, je opazovani primer izjema, če se vsi sosedi iz lokalne okolice nahajajo daleč od njega. Prednost uporabe razdalje za določanje izjem je v tem, da ni potrebno poznati distribucije primerov in da lahko izjeme na ta način definiramo na vsakem prostoru, na katerem je definirana razdalja.

Tri najpogostejše definicije izjem so naslednje:

1. Izjeme so tisti primeri, za katere obstaja manj kot p drugih primerov, ki se nahajajo v razdalji manjši ali enaki d [12, 11].

2. Izjeme so tisti prvi n primeri, ki se nahajajo najdlje od k-tega najbližjega seseda [16].

3. Izjeme so tisti prvi n primeri, katerih povprečna razdalja do k najbližjih sosedov je največja [2, 8].

Med temi definicijami obstajajo manjše razlike. Prva izjem ne rangira in zahteva, da se določi mejna razdalja d, kar lahko včasih povzroča težave. Druga definicija ne upošteva informacije o primerih, bližjih od k-tega najbližjega primera. Tretja pa odpravlja pomanjkljivosti prvih dveh definicij, a je zato izvajanje metod na njeni podlagi časovno bolj zahtevno.

Vsem definicijam na podlagi razdalje je skupno, da znajo povedati le, da je primer izjemen, ne pa tudi koliko se razlikuje od ostalih primerov. To je mogoče doseči, če za definiranje izjem uporabimo *gostoto* [7, 15]. To pomeni, da je primer definiran kot izjema glede na to, kolikšna je njegova lokalna gostota glede na lokalne gostote njegovih sosedov.

Izjeme lahko poiščemo tudi na drugačen način. Z uporabo strojnega učenja se lahko na podatkih naučimo različnih pravil, ki opisujejo te podatke. Izjeme lahko potem določamo na podlagi *klasifikatorjev* tako, da je izjema vsak primer, ki ga klasifikatorji različno klasificirajo.

V nadaljevanju si bomo najprej ogledali štiri metode, ki izjeme iščejo s pomočjo razdalje, nato pa še metodo, ki temelji na gostoti.

2.2 Ugnezdene zanke

Najenostavnejši algoritem za odkrivanje izjem je *algoritem ugnezdenih zank* (angl. nested loops) [12, 11, 16]. V osnovni različici algoritem izračuna razdalje med vsakim parom primerov in to uporabi za ugotavljanje izjem po eni od zgornjih definicij. Algoritem ima kvadratno časovno zahtevnost $\mathcal{O}(N^2)$ glede na število vseh primerov N. V primeru številnih podatkov je to prevelika zahtevnost, zato so raziskovalci veliko napora namenili razvoju algoritmov z manjšo časovno zahtevnostjo.

2.3 Prostorske indeksne strukture

Izjeme lahko odkrivamo tudi s pomočjo *prostorskih indeksnih struktur*, kot so KD-drevo [4], R-drevo [9] ali X-drevo [5], s katerimi lahko poiščemo najbližjega soseda za obravnavani primer celo v času $\mathcal{O}(\log N)$ [12, 11, 16]. Odkrivanje izjem tako zahteva čas $\mathcal{O}(N \log N)$. Vendar pa to drži le za prostore z malo dimenzijami, saj drevesa hitro odpovedo, če je število dimenzij večje od pet.

2.4 Particije prostora

Zahtevnost algoritma za odkrivanje izjem se lahko zmanjša, če prostor razdelimo na predele in tako omogočimo hitrejše iskanje najbližjih sosedov. Za vsak predel si zapomnimo določene podatke, kot je npr. minimalni mejni pravokotnik. Ko iščemo najbližje sosede nekega primera, primerjamo primer z mejnim pravokotnikom in tako ugotovimo ali lahko najbližji sosed prihaja iz tistega predela. Če to ni možno, potem noben primer iz tistega predela ne more biti najbližji sosed opazovanega primera. V [12] prostor razdelijo na hiperpravokotnike, s čimer dosežejo časovno zahtevnost, ki je linearna glede na število primerov N, a eksponentna glede na število dimenzij. Zato je primerna samo za iskanje po prostorih z manj kot petimi dimenzijami. V [16, 8] prostor razdelijo na particije s pomočjo gruč. Ta algoritem se je izkazal za boljšega od algoritmov z ugnezdenimi zankami ali prostorskimi indeksnimi strukturami, a je bil preizkušen samo na prostorih z malo dimenzijami.

2.5 Projekcije

Nekateri raziskovalci uporabljajo projekcije prostora, s katerimi se skušajo izogniti problemom, ki jih prinesejo večdimenzionalni prostori. Če namreč gledamo razdalje med primeri v večdimenzionalnem prostoru, se lahko zgodi, da ne ugotovimo izjem, ki se močno razlikujejo v eni dimenziji in malo v drugih, saj se navadno razlike po posameznih dimenzijah seštevajo. Zato v [1] predlagajo, da se prostor projicira na manjdimenzionalne prostore. V [2] pa prostor večkrat projicirajo na interval $[0, 1]$ s Hilbertovimi krivuljami. Vsaka naslednja projekcija izboljša oceno "izjemnosti" primera. Njihovi rezultati kažejo, da algoritem dosega skoraj linearno časovno zahtevnost.

2.6 Lokalni koeficient izjemnosti

Lokalni koeficient izjemnosti (angl. local outlier factor, LOF) je za vsak primer definiran s pomočjo lokalne gostote primera in lokalne gostote njegovih sosedov [7]. Število sosedov, ki jih upoštevamo v tem izračunu, je parameter algoritma. Tako dobljeni koeficient vsakemu primeru določi število večje ali enako 1, ki pove, kolikšna je izjemnost primera. Primeri, ki niso izjeme, imajo LOF = 1. Večji kot je LOF, večja je izjemnost primera. Rezultati dobljeni z uporabo koeficienta izjemnosti so zelo dobri – metoda pravilno določi izjeme tudi v primerih, ko so podatki različno distribuirani. Na takšnih podatkih imajo navadno metode, ki temeljijo na razdalji, težave. V [15] so raziskovalci metodo razvili korak naprej tako, da so edini parameter metode (število sosedov) določili avtomatsko iz podatkov. Slabost uporabe lokalnega koeficienta izjemnosti pa je časovna zahtevnost metode, saj je kvadratna glede na število primerov.

3 POSEBNOSTI PODATKOV SISTEMA ZA KONTROLO PRISTOPA

Številni primeri uporabe odkrivanja izjem, kot je npr. prečiščevanje podatkov, iščejo izjeme izmed N primeri šele potem, ko so vsi primeri že znani. Pri sistemu za kontrolo pristopa pa izjema lahko pomeni poskus neavtoriziranega vstopa in zato želimo, da se izjeme odkrivajo sproti in kar se da hitro. Na ta način lahko sistem ukrepa takoj – sproži alarm ali opozorilo. Sprotno odkrivanje izjem prinaša tudi prednosti: ker želimo ugotoviti, ali je dani (zadnji dobljeni) primer izjema, naenkrat obdelujemo samo en primer. Časovna (in prostorska) zahtevnost takšnega postopka je zato manjša.

Seveda ni vseeno, koliko se novi primer razlikuje od preostalih. Vzemimo za primer uporabnika, ki navadno prihaja v službo vsak dan ob osmih. Izjemni prihod ob devetih je manj pomemben kot izjemni prihod ob dvajsetih. Algoritem za odkrivanje izjem mora torej znati razlikovati med izjemami različnih veličin, da se lahko sistem nanje ustrezno odziva. Prihod ob devetih bi tako lahko sprožil manjše opozorilo, medtem kot bi prihod ob dvajsetih lahko bil dovolj nenavaden, da bi sprožil alarm.

Ker so navade in obnašanje pri pristopih vezane na posameznega uporabnika, mora sistem iskati izjeme v okviru podatkov za vsakega uporabnika posebej. To pomeni, da ima metoda za na voljo malo primerov. Po eni strani je to dobrodošlo, saj bo tako metoda za odkrivanje izjem gotovo hitra, po drugi strani pa majhno število primerov pomeni težjo nalogo – toliko bolj, če je dimenzionalnost prostora velika. Poleg tega velja, da vsi primeri niso enako pomembni. Če npr. spremljamo podatke o makro obnašanju v obdobju več let, so starejši primeri manj pomembni od novejših, saj se človek (in s tem njegove navade) skozi čas spreminja. Primere je zato treba utežiti glede na njihovo aktualnost.

Dodatna lastnost podatkov sistema za kontrolo pristopa je obstoj nominalnih oz. poimenskih značilk. Večina metod za odkrivanje izjem (razen tistih, ki temeljijo na klasifikatorjih) potrebuje definicijo razdalje za vsako značilko. Pri nominalnih značilkah se navadno uporablja dvojiška razdalja: razdalja je enaka 1, če sta nominalni značilki različni, in 0 sicer. Vendar pa so določene značilke takšne, da bi dopuščale tudi drugačno definicijo razdalje. Sistemi za kontrolo pristopa beležijo tipe dogodkov, kot so npr. "prihod na delo", "odhod na malico", "odhod z dela" in podobni. Namesto običajne, dvojiške definicije, tu lahko razdalje določimo tako, da so si npr. prihodi med sabo bolj podobni kot prihod in odhod.

Glede na predstavljene lastnosti podatkov sistema za kontrolo pristopa lahko ugotovimo, da bi bile izmed algoritmov, ki izjeme definirajo s pomočjo razdalje, za naše potrebe še najbolj primerne projekcije. Le-te se namreč ukvarjajo z večdimenzionalnimi prostori in lahko odkrijejo tudi izjeme, ki se močno razlikujejo samo v eni dimenziji oz. značilki. Poleg tega algoritem, ki uporablja projekcije, odlikuje skoraj linearna časovna zahtevnost. Ker za obravnavani primer želimo poznati tudi stopnjo izjemnosti, velja poskusiti algoritem, ki izračuna lokalni koeficient izjemnosti. Tudi kombinacija obeh algoritmov lahko prinese zelo dobre rezultate. Kot popolnoma drugačen pristop bi bilo smiselno preizkusiti tudi kakšne izjeme lahko določimo s pomočjo klasifikatorjev.

4 ZAKLJUČEK

V prispevku smo obravnavali sistem za kontrolo pristopa, ki združuje več različnih biometričnih senzorjev in svoje delovanje nadgrajuje z uporabo inteligence. Ključna prednost takšnega sistema je zmožnost odkrivanja odstopanj od ustaljenih vzorcev obnašanja uporabnikov. Takšne izjeme lahko pomenijo nevarnost, kot je npr. poskus vstopa neavtorizirane osebe ali pa neobičajno obnašanje sicer pravega uporabnika, ki pa se giblje drugače kot navadno zaradi poškodbe ali vinjenosti.

Razvoj inteligentnega sistema je v začetni fazi in pred nami so še številne naloge. Najprej moramo iz podatkov o pristopih izluščiti tiste značilke, ki najbolje opisujejo obnašanje uporabnika. Za nominalne značilke moramo definirati razdalje, ki so lahko bodisi binarne bodisi upoštevajo znanje o značilkah in so posledično bolj informativne. Nato moramo opisane algoritme za odkrivanje izjem preizkusiti na teh podatkih. Odkrivanje izjem je oblika nenadzorovanega učenja, pri katerem poseben izziv predstavlja tudi vrednotenje kakovosti algoritmov.

Ker se lahko sistem uspešno uči šele potem, ko že vsebuje nekaj uporabnih podatkov, je treba predvideti delovanje sistema tudi, ko teh podatkov še nima. To pomeni, da mora biti v sistem vključeno domensko predznanje v obliki ontologij o značilnem obnašanju posameznikov in skupin. To znanje bo najpomembnejše pri prvih pristopih uporabnikov oz. pri zagonu sistema, vendar se bo uporabljalo tudi kasneje v povezavi z naučenim znanjem.

Literatura

[1] C. C. Aggarwal and P. S. Yu. Outlier detection for high dimensional data. In *Proceedings of the International Conference on Management of data (SIGMOD'01)*, pages 37–46, 2001.

[2] F. Angiulli and C. Pizzuti. Fast outlier detection in high dimensional spaces. In *Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery (PKDD'02)*, pages 15–26, 2002.

[3] V. Barnett and T. Lewis. *Outliers in Statistical Data*. John Wiley & Sons, 1994.

[4] J. L. Bentley. Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9):509–517, 1975.

[5] S. Berchtold, D. A. Keim, and H.-P. Kriegel. The X-tree: An index structure for high-dimensional data. In *Proceedings of the 22th International Conference on Very Large Data Bases (VLDB'96)*, pages 28–39, 1996.

[6] R. J. Bolton and D. J. Hand. Statistical fraud detection: A review. *Statistical Science*, 17(3):235–255, 2002.

[7] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander. LOF: Identifying density-based local outliers. In *Proceedings of the International Conference on Management of Data (SIGMOD'00)*, pages 93–104, 2000.

[8] E. Eskin, A. Arnold, M. Prerau, L. Portnoy, and S. Stolfo. A geometric framework for unsupervised anomaly detection: Detecting intrusions in unlabeled data. In *Proceedings of the Data Mining for Security Applications Workshop*, 2002.

[9] A. Guttman. R-trees: A dynamic index structure for spatial searching. In *Proceedings of the International Conference on Management of data (SIGMOD'84)*, pages 47–57, 1984.

[10] D. Hawkins. *Identification of Outliers*. Chapman and Hall, 1980.

[11] E. M. Knorr, R. T. Ng, and V. Tucakov. Distance-based outliers: algorithms and applications. *The VLDB Journal*, 8(3-4):237–253, 2000.

[12] E. M. Knorr and R. T. Ng. Finding intensional knowledge of distance-based outliers. In *Proceedings of the 25th International Conference on Very Large Data Bases (VLDB'99)*, pages 211–222, 1999.

[13] T. Lane and C. E. Brodley. Temporal sequence learning and data reduction for anomaly detection. *ACM Transactions on Information System Security*, 2(3):295–331, 1999.

[14] A. Loureiro, L. Torgo, and C. Soares. Outlier detection using clustering methods: A data cleaning application. In *Proceedings of the Data Mining for Business Workshop*, 2005.

[15] S. Papadimitriou, H. Kitagawa, P. Gibbons, and C. Faloutsos. LOCI: Fast outlier detection using the local correlation integral. In *Proceedings of the 19th International Conference on Data Engineering (ICDE)*, pages 315–326, 2003.

[16] S. Ramaswamy, R. Rastogi, and K. Shim. Efficient algorithms for mining outliers from large data sets. In *Proceedings of the International Conference on Management of Data (SIGMOD'00)*, pages 427–438, 2000.