Zaznavanje spremembe učnega koncepta na podatkovnih tokovih slik z metodo Gan Loss Drift Detection (GLDD)

Jan Kuhta, Zoran Bosnić

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, Večna pot 113, 1000 Ljubljana
E-pošta: jan.kuhta@gmail.com

Concept drift detection in image data streams using the Gan Loss Drift Detection Method (GLDD)

This paper presents an innovative method for detecting concept drifts in image data streams based on discriminator loss analysis within a Wasserstein generative adversarial network (WGAN) learning loop. We observed that when the distribution of the input data changes abruptly, the loss of the discriminator for a given number of iterations increases sharply in absolute value. We exploited this specific property in the development of the Gan Loss Drift Detection (GLDD) method. We thoroughly tested the method on an image dataset MNIST digits, which we had previously transformed into ten different distributions. During the experiments, the GLDD-KSWIN version performed particularly well, achieving an average precision of 0.79, a recall of 0.90, an F1-score of 0.84, and a latency of 41.64. The results show that the proposed method provides a promising foundation for further research in this area, which still remains largely untouched and challenging.

1 Uvod

V dobi digitalizacije se podatki okoli nas širijo z eksponentno hitrostjo. Zaradi preobsežne količine teh podatkov so številne tradicionalne modele strojnega učenja nadomestili inkrementalni modeli, ki obdelujejo podatke zaporedoma in se konstantno izboljšujejo. Prav tako ne potrebujejo shranjevanja večjih podatkovnih zbirk, saj lahko po obdelavi podatke zavržejo. Učenje iz podatkovnih tokov (imenovano tudi *inkrementalno učenje*) predstavlja vse bolj raziskovano področje zaradi njihovih specifičnih lastnosti in široke uporabe. Ena od ključnih lastnosti podatkovnih tokov je, da so lahko nestacionarni, saj lahko vhodni primeri prihajajo iz različnih porazdelitev, ali pa se le-ta čez čas spreminja.

Pojav, ki ga imenujemo sprememba učnega koncepta (angl. *concept drift*) [1], predstavlja velik izziv v strojnem učenju, saj lahko pomembno vpliva na učinkovitost modelov. Številni modeli pri spremembi koncepta postanejo manj natančni ali celo zastareli, zato je takšne spremembe treba pravočasno odkriti in model prilagoditi. Večino dosedanjega dela je bilo osredotočenega na besedilne ali numeričnime vhodne podatke, področje zaznavanja sprememb na slikovnih vhodnih podatkih pa zaenkrat ostaja še dokaj neraziskano.

V tem članku predstavljamo inovativno metodo za zaznavanje morebitnih sprememb koncepta v porazdelitvi dinamičnih slikovnih vhodnih podatkov. Predstavljena metoda, imenovana GLDD (Gan Loss Drift Detection), je integrirana v sistem, ki inkrementalno uči model za računalniško generiranje slik. Metoda temelji na vrednostih izgub diskriminatorja pri učenju generativne nasprotniške mreže Wasserstein z gradientno kaznijo (WGAN-GP, v nadaljevanju WGAN). Ob nenadnih porastih ali padcih, ki so statistično značilni, signalizira spremembo koncepta v vhodnih podatkih.

2 Sorodna dela

Sprememba koncepta se lahko zgodi na več načinov, odvisno od tega, kako se osnovni podatki oziroma učni koncepti (odvisnost odvisne spremenljivke od neodvisnih) spreminjajo s časom. Poznamo nenadno spremembo, kjer nov koncept nadomesti starega v zelo kratkem času, postopno, kjer se v fazi prehoda primeri iz obeh konceptov med seboj mešajo, in inkrementalno, kjer se koncept zvezno spremeni v novega. Kadar se spremenijo posteriorne verjetnosti razredov govorimo o pravi (angl *true*) spremembi koncepta, kadar pa se sprememba zgodi le v vhodnih podatkih, pripadajoče oznake pa ostanejo nespremenjene, pa govorimo o virtualni (angl .virtual) spremembi koncepta. V članku se bomo ukvarjali z nenadno virtualno spremembo koncepta, kjer se bo naenkrat spremenila porazdelitev, ki generira vhodne podatke [2].

Problema zaznavanja spremembe koncepta na slikovnih podatkih so Hashmani in sod. [3] lotili s spremljanjem odziva klasifikacijske točnosti napovednih modelov na dveh spremenjenih testnih množicah. Testiranje so izvedli na priznanih slikovnih učnih množicah MNIST digits, CIFAR10 in CALTECH 101. Podobno so spremembo koncepta na slikovnih množicah, sicer pravo spremembo koncepta, predstavili Budiman in sod. [4]. Pokazali so, da sprememba koncepta sovpada s padcem klasifikacijske točnosti napovednih modelov.

Ackerman in sod. [5] so za posamezno porazdelitev podatkov izvedli metodo zmanjševanja dimenzionalnosti (angl. *Principle Component Analysis, PCA*), nato pa gručenje. Spremembo koncepta so signalizirali v iteraciji, kjer so središča gruč odstopala od središč iz prejšnjih iteracij za vrednost, večjo od izbranega praga.

Generativne nasprotniške mreže (GAN) [6] so prinesle pomembne novosti na področju generativnega modeliranja. Sestavljene so iz dveh konvolucijskih nevronskih mrež, ki ju simultano treniramo: generatorja, ki se uči generiranja umetnih primerov, čim bolj podobnih realističnim primerom, iz naključnega šuma, in diskriminatorja, ki razlikuje med realnimi in generiranimi primeri. Cilj generatorja je minimizacija funkcije izgube, medtem ko je cilj diskriminatorja maksimizacija funkcije izgube. Matematično jo zapišemo kot:

$$L_{\text{GAN}}(D, G) = E_x[\log D(x)] + E_z[\log(1 - D(G(z)))]$$

kjer $D(x)$ predstavlja verjetnost, da je podatek x pravi podatek, $G(z)$ pa verjetnost, da je podatek generiran iz šuma z.

Tradicionalni arhitekturi GAN in DCGAN (Deep Convolutional GAN) je nadomestila arhitektura Wasserstein GAN (WGAN), ki rešuje težave s stabilnostjo in pojavom kolapsa načinov z uvedbo Wassersteinovo razdalje namesto Kullback-Leiblerjeve divergence in omogoča bolj stabilno merjenje podobnosti med verjetnostnimi porazdelitvami [7]. Dodatna izboljšava arhitekture WGAN je bila uvedba gradientne kazni (WGAN-GP), ki v izračunu izgube diskriminatorja kaznuje odstopanje gradientov od vrednosti 1, kar izboljšuje stabilnost treniranja [8].

3 Metodologija

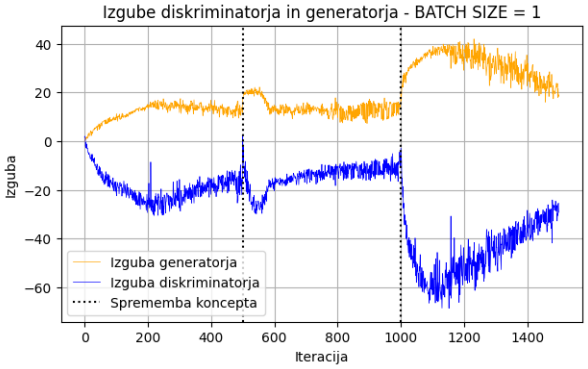
Učni sistem

Postopek učenja nevronskih mrež je že po naravi delno inkrementalen, navadno učenje poteka v malih paketih v več epohah. Mi smo želeli simulirati popoln inkrementalen sistem, kjer bi podatki prihajali zaporedno primer po primer in parametre posodobili po vsakem primeru. Testiranje je pokazalo, da nizka vrednost velikosti paketa povzroča nestabilnost učenja, vendar smo z optimizacijo ostalih parametrov dobili zadovoljive rezultate in želeno popolno inkrementalno učenje modela.

Nenadne spremembe učnega koncepta smo simulirali s spreminjanjem porazdelitve vhodnih podatkov. To smo dosegli z uporabo podatkov iz različnih transformiranih množic, pri čemer smo iz vsake uporabili določeno število primerov. Inkrementalno smo učili naš sistem WGAN in spremljali nenadne poraste in padce vrednosti izgub diskriminatorja in generatorja. Izkaže se, da se izgube ob spremembi koncepta praviloma povečajo za določeno število iteracij, nato pa počasi konvergirajo proti nižjim absolutnim vrednostim. Slika 1 prikazuje gibanje izgub diskriminatorja in generatorja za arhitekturo WGAN na enostavnih, umetno sintetiziranih slikovnih podatkih, pri dveh spremembah koncepta (iteraciji 500 in 1000) in velikosti parametra batch_size = 1.

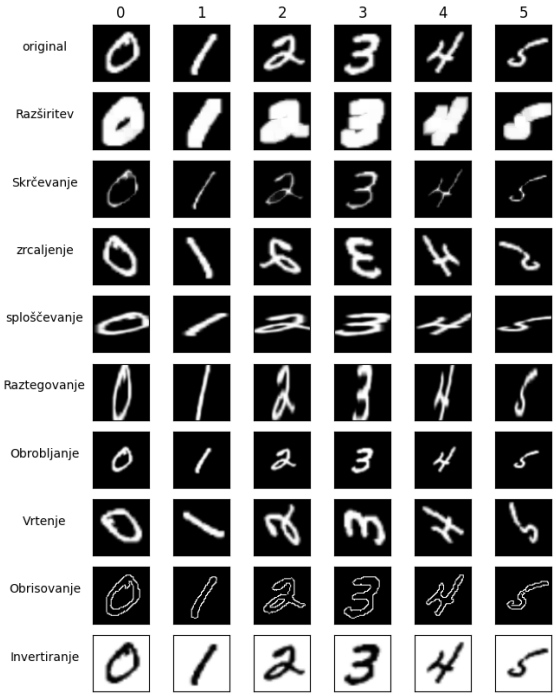
Priprava podatkov

Učno množico je sestavljalo 70.000 črno-belih sličic iz slikovne množice MNIST digits. Za zagotovitev različnih porazdelitev smo učne podatke transformirali z uporabo naslednjih transformacij: razširitev, skrčitev, zrcaljenje, sploščevanje, raztegovanje, obrobljanje, vrtenje, obrisovanje in invertiranje (slika 2). Transformirane sličice smo



Slika 1: Graf izgub pri dveh spremembah koncepta

razširili na velikost 64 × 64 in normalizirali vrednosti na interval [-1, 1].



Slika 2: Primeri sličic iz transformiranih množic

Metoda GLDD

Metoda GLDD se opira na analitično zmožnost diskriminatorja v modelu WGAN za ocenjevanje razlike med resničnimi in umetnimi podatkovnimi porazdelitvami. Čeprav konvergenca izgube diskriminatorja ni nujno pogojena, se izkaže, da je opazovanje trendov izgube informativno v smislu spremembe v porazdelitvi podatkov, saj sprememba sproži nenaden porast (ali padec) izgub. Na tem principu temelji naša metoda GLDD, ki omogoča učinkovito zaznavanje sprememb koncepta v slikovnih podatkih. Kot informativnejša in stabilnejša se je izkazala izguba diskriminatorja, zato smo se osredotočili le nanj.

Metoda GLDD je integrirana v učni proces sistema WGAN, kjer v zanki učenja spremljamo vrednosti izgube

diskriminatorja. Na začetku učenja inicializiramo algoritem za zaznavanje spremembe koncepta, ki spremlja nihanje izgub. V vsaki iteraciji posodobimo trenutno vrednost in preverimo, ali je sprememba zaznana, in jo v tem primeru signaliziramo. Metoda sprejme naslednje parametre:

- podatkovni tok izgub,
- algoritem za zaznavanje spremembe koncepta,
- topel zagon (angl. *warm start*), ki onemogoči zaznavanje spremembe koncepta v začetnih iteracijah, in preprečuje napačne zaznave sprememb koncepta v začetni fazi učenja, kjer so izgube modela GAN zelo nestabilne in nepredvidljive,
- obdobje tolerance (angl. *grace period*), ki po zaznani spremembi začasno zamrzne ponovno zaznavanje, s čimer preprečimo večkratno zaznavanje iste spremembe.

Predlagamo dve varianti metode GLDD, ki se med seboj razlikujeta po algoritmu za zaznavanje spremembe koncepta: GLDD-KSWIN in GLDD-SPC. Prva temelji na principu uporabe algoritma drsečih oken KSWIN (Kolmogorov-Smirnov Windowing). Temelji na neparametričnem statističnem testu Kolmogorov-Smirnova (KS), ki vzdržuje drseče okno ψ fiksne velikosti n. Iz okna ψ sestavi dve manjši enako veliki podokni R in W. Podokno R predstavlja najnovejši koncept in ga sestavlja r najnovejše prispelih primerov, podokno W pa vzorči podatke iz prvih $n - r$ primerov v ψ in predstavlja približni zadnji koncept. KSWIN zazna spremembo koncepta, ko je razlika med porazdelitvama v podoknih W in R statistično značilna [9].

Druga metoda GLDD-SPC deluje v stilu statističnega nadzorovanja procesov (angl. *Statistical Process Control*, SPC). Klasično metodo SPC [10] smo predelali in prilagodili na naš scenarij, tako da lahko sprejme vrednosti izgube diskriminatorja, ki niso omejene na interval $(0, 1)$ in da dopušča negativne vrednosti, prav tako pa spremlja in zaznava tako močne poraste kot tudi močne padce v vrednostih. Osrednjo statistiko predstavlja Eksponentno uteženo gibajoče povprečje (angl. *Exponentially Weighted Moving Average, EWMA*), ki povpreči podatke na način, da daje novejšim podatkom večjo težo. Definiramo spodnjo in zgornjo kontrolno mejo, ki ločuje stanje pod nadzorom od stanja izpod nadzora. Metoda SPC za zaznavo spremembe koncepta vzdržuje statistiki EWMA ter EWMA Var, ter za vsak primer x_t definira stanje pod nadzorom (1) in izpod nadzora (2):

$$EWMA_t - 3 \cdot \sigma \leq x_t \leq EWMA_t + 3 \cdot \sigma \quad (1)$$

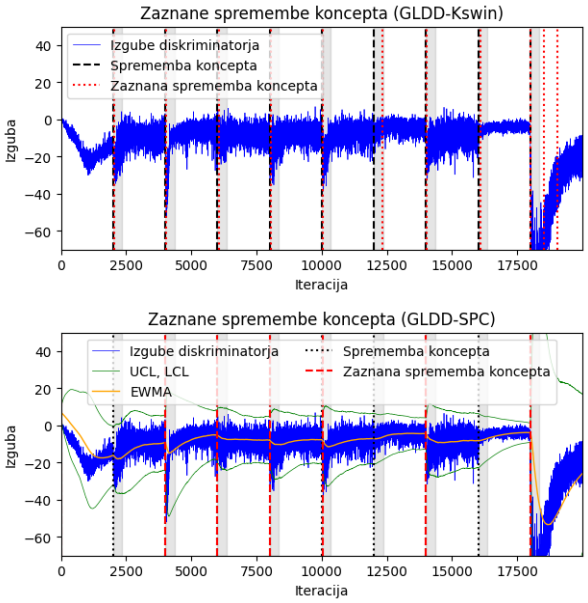
$$x_t > EWMA_t + 3 \cdot \sigma \vee x_t < EWMA_t - 3 \cdot \sigma \quad (2)$$

Ker imamo opravka z izgubami pri generativnih nasprotniških mrežah, ki so že po naravi precej nestabilne, definiramo še drseče okno $window$ fiksne velikosti w_len in signalni prag t. Če je v oknu vsaj t vrednosti preseglo kontrolno mejo, signaliziramo spremembo koncepta. S tem zmanjšamo občutljivost na šum v podatkih.

4 Rezultati

Izvedli smo dve ločeni testiranji predlaganih metod. Vsaka transformirana množica je predstavljala eno porazdelitev in smo jo v učni množic uporabili natanko enkrat. Iz vsake porazdelitve smo naključno vzorčili 2.000 primerov, nato pa porazdelitev zamenjali in simulirali nenadno spremembo koncepta. Testiranje smo ponovili večkrat in optimizirali parametre, ki smo jih nato uporabili v glavnem testiranju.

Ponovitev z optimalnimi parametri smo vizualizirali z grafoma na sliki 3, kjer je z modro označen tok izgube diskriminatorja. Črne navpične črte predstavljajo resnične spremembe konceptov, rdeče črte pa predstavljajo zaznane spremembe konceptov. Sivo ozadje predstavlja obdobje tolerance, v katerem zaznano spremembo koncepta še smatramo kot pravilno pozitiven primer.



Slika 3: Zaznane spremembe konceptov metode GLDD-KSWIN (zgoraj) in GLDD-SPC (spodaj)

Metoda GLDD-KSWIN je na osnovnem testiranju uspešno zaznala vseh devet sprememb konceptov. Težavo je imela le pri zadnji spremembi, ki je najbolj izrazita (invertiranje) in se izgube dalj časa vračajo proti nižjim vrednostim, kjer je metoda isto spremembo zaznala trikrat. Metoda GLDD-SPC je uspešno zaznala šest sprememb in zgrešila zaznavo pri spremembah, ki so bile manj očitne, ni pa imela težav z večkratnim zaznavanjem iste spremembe koncepta.

V glavnem testiranju smo posamezne porazdelitve naključno vzorčili iz tabele vseh transformiranih množic, s tem, da nismo dopustili, da bi dve zaporedni porazdelitvi bili enaki. Iz vsake porazdelitve smo naključno vzorčili primere, tako da nobena množica ni bila enaka, tudi če so primeri prihajali iz iste porazdelitve. Izvedli smo 50 prehodov, vsaka porazdelitev pa je štela 3.000 iteracij. Določili smo parametre warm_start = 1000, grace_period = 350 in tolerance = 350.

Istočasno smo preizkusili obe varianti metode GLDD

in za vsako izračunali natančnost (angl. *precision*), priklic (angl. *recall*), F1-oceno in zamik zaznave, ki je predstavljal povprečno število iteracij med dejanskimi in zaznanimi spremembami koncepta. Testiranje smo ponovili desetkrat, rezultate povprečili ter izračunali standardne odklone. Ker se v praksi slikovni vhodni primeri pogosto srečujejo s šumom, smo želeli preveriti odpornost naše metode na prisotnost šuma v podatkih. Šum smo vpeljali tako, da smo z 10-odstotno verjetnostjo primere vzorčili iz naključne druge porazdelitve. Obdržali smo enake parametre kot v testiranju brez prisotnega šuma in testiranje ponovili. Rezultati testiranja na čistih množicah brez prisotnosti šuma so zbrani v tabeli 1, rezultati testiranja na šumnih podatkih pa v tabeli 2.

Tabela 1: Povprečni rezultati obeh metod na čistih podatkih

Metrika	GLDD-KSWIN	GLDD-SPC
natančnost	0.79 ± 0.03	$\mathbf{0.92 \pm 0.04}$
priklic	$\mathbf{0.90 \pm 0.06}$	0.67 ± 0.03
F1-ocena	$\mathbf{0.84 \pm 0.03}$	0.77 ± 0.02
zamik zaznave	41.64 ± 4.10	$\mathbf{13.00 \pm 3.14}$

Rezultati testiranja na čistih podatkih so pokazali, da sta metodi dokaj uspešni. Zelo natančna je metoda GLDD-SPC z natančnostjo 0.92, nekoliko manj natančna pa je metoda GLDD-KSWIN (0.79). Slednja je bistveno prehitela metodo GLDD-SPC v priklicu (0.90) ter F1-oceni (0.84). Nekoliko več dejanskih sprememb je izpustila metoda GLDD-SPC z nižjim priklicem (0.67). GLDD-SPC izstopa po zelo nizkem zamiku zaznave, saj je povprečno porabila le 13 primerov za uspešno zaznavo spremembe koncepta.

Tabela 2: Povprečni rezultati obeh metod na šumnih podatkih

Metrika	GLDD-KSWIN	GLDD-SPC
natančnost	$\mathbf{0.84 \pm 0.02}$	0.48 ± 0.07
priklic	$\mathbf{0.87 \pm 0.09}$	0.45 ± 0.11
F1-ocena	$\mathbf{0.85 \pm 0.04}$	0.46 ± 0.09
zamik zaznave	37.77 ± 3.40	$\mathbf{9.88 \pm 5.58}$

Iz rezultatov testiranja na šumnih podatkih ugotovimo, da je metoda GLDD-KSWIN dobro odporna na šum, saj se rezultati ne razlikujejo bistveno od rezultatov na čistih podatkih. Povprečna F1-ocena je dosegla visoko vrednost, 0.85. Nasprotno pa se metoda GLDD-SPC odziva bistveno slabše na prisotnost šuma v vhodnih podatkih, saj zvišano število napačno pozitivnih detekcij znatno poslabša njeno natančnost, kar se odraža v zmanjšanju F1-ocene za 0.31. Ker so bili parametri optimizirani na podatkih brez šuma, bi lahko uspešnost metode mogoče izboljšali z dodatno optimizacijo za šumne podatke.

5 Zaključek

Področje zaznavanja spremembe konceptov pri inkrementalnem učenju na slikovnih množicah je zaenkrat še dokaj neraziskano. Nekateri so se problema lotili s spremljanjem klasifikacijske točnosti modelov čez čas, drugi so uporabili pristop zmanjševanja dimenzij in gručenja. V članku smo predstavili inovativno metodo GLDD, ki omenjen problem prepleta z inkrementalnim učenjem generativnih nasprotniških mrež. Na podlagi izgub diskriminatorja in uporabe posebnih algoritmov (KSWIN, SPC) zaznava spremembe koncepta v slikovnih vhodnih podatkih. Predstavili smo dve variaciji metode GLDD, ki se razlikujeta po vrsti uporabljenega algoritma za detektiranje sprememb. Kot najboljša (po F1 oceni) se je izkazala metoda GLDD-KSWIN, nekoliko manj uspešna, pa vendar še dokaj dobra je bila metoda GLDD-SPC, ki je izstopala po natančnosti (0.92) in odzivnosti oziroma zamiku zaznave (13 iteracij).

V prihodnosti bi bilo metodo smiselno testirati na bolj praktičnih testnih domenah, kot je npr. spreminjanje tipografije. Prav tako bi lahko preizkusili njeno delovanje v scenarijih z inkrementalnimi ali postopnimi spremembami ter na kompleksnejših slikovnih množicah, kot sta CIFAR ali CeleB. Smiselna nadgradnja naše metode bi bila tudi implementacija hitre in učinkovitejše adaptacije sistema GAN na nov učni koncept po zaznani spremembi koncepta s strani naše metode.

Literatura

[1] C. Raab, M. Heusinger, in F.-M. Schleif, "Reactive soft prototype computing for concept drift streams,". V: *Neurocomputing*, vol. 416, str. 340–351 (2020).

[2] J. Gama, *Knowledge discovery from data streams* (CRC Press, 2010).

[3] M. A. Hashmani, S. M. Jameel, H. Alhussain, M. Rehman, in A. Budiman, "Accuracy performance degradation in image classification models due to concept drift,". V: *International Journal of Advanced Computer Science and Applications*, (2019).

[4] Arif Budiman, Mohamad Ivan Fanany, in Chan Basaruddin, "Adaptive Convolutional ELM For Concept Drift Handling in Online Stream Data," *ArXiv* (2016). Dostopno na: `https://api.semanticscholar.org/CorpusID:18995640`.

[5] Samuel Ackerman, Eitan Farchi, Orna Raz, Marcel Zalmanovici, in Parijat Dube, "Detection of data drift and outliers affecting machine learning model performance over time," *ArXiv* (2020). Dostopno na: `https://api.semanticscholar.org/C`.

[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, in Y. Bengio, "Generative adversarial nets,". V: *Advances in neural information processing systems* (2014).

[7] M. Arjovsky, S. Chintala, in L. Bottou, "Wasserstein generative adversarial networks,". V: *International conference on machine learning*, str. 214–223 (2017).

[8] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin in A. Courville, "Improved training of Wasserstein GANs,". V: *Advances in neural information processing systems* (2017).

[9] M. U. Togbe, Y. Chabchoub, A. Boly, M. Barry, R. Chiky, in M. Bahri, "Anomalies detection using isolation in concept-drifting data streams,". V: *Computers*, vol. 10, no. 1, str. 13 (2021).

[10] J. Gama, P. Medas, G. Castillo, in P. Rodrigues, "Learning with drift detection,". V: *Advances in Artificial Intelligence* (2004).