Sigurnost strojnog učenja

Stjepan Picek[1,2,3]
[1]Sveučilište u Zagrebu Fakultet elektrotehnike i računarstva, Unska 3, 10000 Zagreb
[2]Radboud University, Houtlaan 4, 6525 XZ Nijmegen, Nizozemska
[3]University of Bergen, Muséplassen 1, 5007 Bergen, Norveška

Sažetak: *Strojno učenje brzo prelazi iz istraživačke domene u temeljni element moderne digitalne infrastrukture. Pritom sustavi strojnog učenja postaju i meta sve sofisticiranijih napada, čime se otvara nova i rastuća površina rizika. Ovaj pregledni rad analizira sigurnost strojnog učenja s naglaskom na ranjivosti svojstvene sustavima temeljenima na podacima te na protivnike koji ih iskorištavaju. Nakon razgraničenja sigurnosti umjetne inteligencije i umjetne inteligencije za sigurnost, fokus stavljamo na prvi pristup te razlikujemo namjerne napade od nenamjernih pogrešaka i kvarova modela. Namjerne napade klasificiramo prema fazi životnog ciklusa strojnog učenja na koju su usmjereni, odnosno prema treniranju i inferenciji, te prema ciljevima napada u okviru CIA trijade. Nadalje, razmatramo sigurnost velikih jezičnih modela, osobito prompt injection i jailbreak napade. U završnom dijelu razmatramo regulatorni kontekst, posebno AI Act Europske unije i GDPR.*

Ključne riječi: *sigurnost strojnog učenja, adversarijalno strojno učenje, prompt injection, jailbreak, AI Act*

1. Uvod

Strojno učenje (engl. *machine learning*, ML) danas je ugrađeno u velik broj kritičnih sustava. Modeli se koriste za medicinsku dijagnostiku, procjenu kreditnog rizika, filtriranje sadržaja, detekciju prijevara, autonomnu vožnju, biometriju, obradu prirodnog jezika i pomoć pri donošenju odluka u organizacijama. Takva rasprostranjenost znači da pogreška modela više nije samo tehnički problem kvalitete predikcije, nego može imati i operativne, financijske, pravne te sigurnosne posljedice. Zbog toga se sigurnost strojnog učenja više ne može promatrati kao usko tehničko pitanje robusnosti modela, nego kao širi problem pouzdanosti sustava koji nastaje na sjecištu podataka, algoritama, infrastrukture i načina uporabe [1, 2].

U literaturi se često razlikuju dva pristupa: sigurnost umjetne inteligencije (engl. *security of AI*) i umjetna inteligencija za sigurnost (engl. *AI for security*). Prvi pristup bavi se zaštitom modela, podataka i aplikacija koje koriste AI od napada, manipulacije, curenja informacija i degradacije performansi. Drugi pristup promatra AI kao alat za obranu, primjerice za otkrivanje anomalija, pomoć pri analizi zlonamjernog koda ili automatizaciju odgovora na incidente. Iako su ta dva područja povezana, važno ih je razlikovati jer se sigurnosni zahtjevi, model prijetnji i metode evaluacije bitno razlikuju [1]. Ovaj rad usmjeren je na sigurnost samih ML sustava, njihovih ulaza, izlaza, opskrbnog lanca i mehanizama uporabe.

Dodatnu složenost donosi širenje velikih jezičnih modela i generativne umjetne inteligencije u svakodnevne alate i poslovne procese. U tim sustavima prirodni jezik nije samo medij prijenosa sadržaja, nego i mehanizam upravljanja ponašanjem modela. Time se briše granica između podataka i uputa, što otvara prostor za *prompt injection*, *jailbreak* i druge oblike manipulacije koji nisu tipični za ranije, uže definirane ML aplikacije [3, 4].

Važno je naglasiti da sigurnost strojnog učenja ne prestaje trenutkom puštanja modela u rad. U produkcijskom okruženju model ostaje dio šireg socio-tehničkog sustava u kojem se mijenjaju ulazni podaci, korisnički obrasci, vanjske integracije i poslovna pravila. Zbog toga sigurnosna analiza mora obuhvatiti cijeli razvojno-operativni ciklus strojnog učenja (MLOps): prikupljanje i verzioniranje podataka, treniranje, validaciju, implementaciju, nadzor i upravljanje promjenama modela. Tek tada je moguće govoriti o stvarnoj otpornosti sustava, a ne samo o sigurnosti jedne verzije modela u laboratorijskim uvjetima.

Cilj ovog rada je ponuditi preglednu klasifikaciju sigurnosnih prijetnji nad sustavima strojnog učenja, analizirati najvažnije tehnike napada, raspraviti obrambene pristupe i smjestiti problem u širi tehnički i regulatorni kontekst. Valja pritom naglasiti da sigurnost strojnog učenja obuhvaća širi skup prijetnji, napada i obrambenih pristupa nego što ih je moguće detaljno obraditi u jednom preglednom radu ograničenog opsega. Stoga se u ovom radu ne nastoji dati iscrpan katalog svih poznatih metoda, nego se naglasak stavlja na najvažnije i najreprezentativnije klase napada i obrana koje pružaju dobar temelj za razumijevanje sigurnosti suvremenih sustava strojnog učenja.

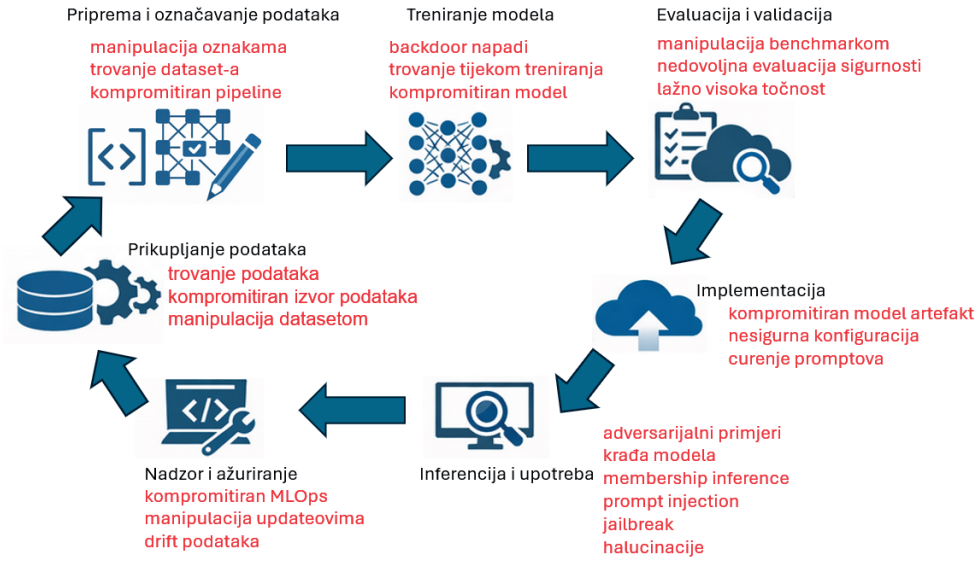
Rad je strukturiran tako da najprije uvodi pojmovni okvir i taksonomiju prijetnji (Poglavlje 2), zatim zasebno razmatra napade u fazi treniranja (Poglavlje 3) i napade u fazi inferencije (Poglavlje 4). Potom analizira sigurnost velikih jezičnih modela (Poglavlje 5), predstavlja dvije studije slučaja iz novije literature (Poglavlje 6), razmatra obrambene pristupe (Poglavlje 7) i regulatorni okvir (Poglavlje 8). Naposljetku, sažima glavne otvorene izazove (Poglavlje 9).

2. Pojmovni okvir i taksonomija prijetnji

U raspravi o sigurnosti strojnog učenja važno je najprije razgraničiti sigurnosne incidente od općih problema pouzdanosti modela. Nije svaka pogrešna odluka modela posljedica napadača: mnoge poteškoće proizlaze iz loše kvalitete podataka, pomaka distribucije, pristranih oznaka, neodgovarajuće metrike ili neprikladne evaluacije. Takvi problemi mogu imati ozbiljne posljedice, ali ih metodološki treba razlikovati od situacija u kojima napadač svjesno i ciljano manipulira ulazima, podacima, okruženjem ili izlazima modela radi ostvarivanja vlastitog cilja [2].

Prva temeljna podjela odnosi se na fazu životnog ciklusa sustava. Napadi u fazi treniranja ciljaju podatke, oznake, hiperparametre, procese učenja ili opskrbni lanac modela. Najpoznatiji primjeri su trovanje podataka i *backdoor* napadi, kod kojih napadač pokušava oblikovati buduće ponašanje modela prije njegove uporabe. Napadi u fazi inferencije izvode se nad već istreniranim modelom te iskorištavaju način na koji model reagira na ulazne upite. U tu skupinu ubrajaju se adversarijalni primjeri, krađa modela, *membership inference*, *prompt injection* i *jailbreak* [2-7].

Sigurnosne prijetnje u sustavima strojnog učenja mogu se pojaviti u različitim fazama njihova životnog ciklusa, od prikupljanja podataka do rada modela u produkciji. Slika 1 prikazuje pregled glavnih faza životnog ciklusa ML sustava te tipične sigurnosne prijetnje povezane s pojedinim fazama.



Slika 1: Životni ciklus sustava strojnog učenja i tipične sigurnosne prijetnje povezane s pojedinim fazama

Druga korisna klasifikacija polazi od sigurnosnih ciljeva, odnosno CIA trijade. Napadi na povjerljivost nastoje izvući informacije o modelu ili podacima na kojima je treniran. Napadi na integritet pokušavaju izazvati pogrešne odluke, zaobići detekciju ili promijeniti ponašanje sustava u korist napadača. Napadi na dostupnost ciljaju degradaciju performansi ili onemogućavanje uporabe modela. Ove se kategorije u praksi preklapaju: krađa modela, primjerice, ugrožava povjerljivost, ali istodobno može olakšati kasnije napade na integritet [2, 6].

Treća važna os klasifikacije odnosi se na znanje i sposobnosti napadača. *White-box* napadač poznaje arhitekturu, parametre i obrambene mehanizme. *Gray-box* napadač raspolaže parcijalnim znanjem o modelu ili skupu podataka. *Black-box* napadač vidi samo ulazno-izlazno ponašanje sustava, primjerice preko aplikacijskog sučelja (API-ja). Upravo je to razlikovanje važno jer brojni napadi, iako izvorno razvijeni za *white-box* okruženje, ostaju učinkoviti i u *black-box* scenarijima zbog transferabilnosti ili curenja informacija kroz izlaze modela [5, 6, 8].

Za potrebe ovog rada korisno je promatrati taksonomiju prijetnji kao višedimenzionalnu matricu, a ne kao jednostavnu listu napada. Isti napad može se opisati prema više osi: prema fazi životnog ciklusa, sigurnosnom cilju, znanju protivnika i operativnom učinku. Takav pristup pogodniji je za pregledni rad jer omogućuje povezivanje tehničkih detalja napada s obranom, upravljanjem rizikom i regulatornim zahtjevima.

3. Napadi u fazi treniranja

U suvremenim sustavima dodatni problem predstavlja opskrbni lanac modela. Organizacije često koriste unaprijed naučene modele, vanjske skupove podataka, otvorene repozitorije i gotove biblioteke za fino podešavanje. Time se povećava produktivnost, ali i otvara mogućnost da kompromitacija nastane prije nego što organizacija uopće započne vlastito treniranje. Sigurnost treniranja zato uključuje provjeru podrijetla artefakata, reproducibilnost procesa učenja, praćenje promjena u skupovima podataka te postupke kojima se može dokazati kako je određena verzija modela nastala i na kojim je ulazima temeljena.

Napadi u fazi treniranja posebno su opasni zato što kompromitaciju ugrađuju u model prije njegove produkcijske uporabe. Takvi napadi mogu ostati neotkriveni dulje vrijeme jer se njihovi učinci ne moraju manifestirati odmah, a model može zadržati dobru opću točnost na standardnim testnim skupovima. Napadač time iskorištava osnovnu činjenicu da model svoju buduću logiku odluke gradi iz podataka i iz procesa optimizacije [2, 9].

3.1 Trovanje podataka

Trovanje podataka (engl. *data poisoning*) obuhvaća namjerno umetanje ili manipulaciju trening uzoraka s ciljem da model nakon treniranja donosi pogrešne odluke. Već su rani radovi pokazali da i relativno mali broj pažljivo konstruiranih uzoraka može znatno utjecati na performanse klasifikatora [9]. U praksi razlikujemo neselektivno trovanje, kojem je cilj opća degradacija modela, i ciljano trovanje, kojem je cilj promjena ponašanja za točno određene uzorke, klase ili scenarije uporabe. Ciljano trovanje posebno je opasno kada napadač želi zaobići detekciju prijevare, biometrijsku provjeru ili sigurnosni klasifikator samo u uskom skupu situacija.

3.2 *Backdoor* napadi

Posebno važna podvrsta trovanja podataka su *backdoor* napadi. Kod njih model zadržava prividno dobro ponašanje na benignim ulazima, ali pokazuje zlonamjerno ili pogrešno ponašanje kada se pojavi određeni okidač. Okidač može biti vizualni uzorak, specifična riječ, sekvenca tokena ili neko drugo prepoznatljivo obilježje ulaza. Upravo zato *backdoor* napadi predstavljaju problem za standardnu evaluaciju: model može izgledati ispravno na testnom skupu, a ipak sadržavati skrivenu ranjivost. To otežava i tehničku provjeru i odgovornost organizacije koja model uvodi u uporabu [2, 10].

Rizik trovanja dodatno raste u okruženjima gdje podaci dolaze iz više izvora i gdje je provjera kvalitete ograničena. To vrijedi za *crowdsourcing*, automatizirano prikupljanje podataka s interneta, repozitorije gotovih skupova podataka, ali i za federirano učenje. U federiranom učenju pojedini klijenti treniraju lokalne modele nad vlastitim podacima, a zatim agregiraju promjene parametara. Takva arhitektura ima prednosti sa aspekta privatnosti, ali otvara i nove sigurnosne rizike: zlonamjerni klijent može pokušati otrovati zajednički model ili ubaciti *backdoor* uz ograničenu vidljivost centralnog koordinatora u njegove lokalne podatke [11].

3.3 Kompromitirani opskrbni lanac modela

Osim trovanja podataka, napad se može provesti i kroz kompromitirani opskrbni lanac modela. U praksi se često koriste unaprijed istrenirani modeli, *checkpoint* preuzeti s javnih repozitorija, biblioteke trećih strana i skripte za fino podešavanje. Svaki od tih elemenata može postati točka kompromitacije. Time sigurnost strojnog učenja postaje slična sigurnosti opskrbnog lanca u klasičnom softverskom inženjerstvu, ali s dodatnim problemom slabe transparentnosti modela: zlonamjerno ponašanje ne mora biti vidljivo u kodu, nego može biti utjelovljeno u parametrima modela i u njegovoj latentnoj reprezentaciji [12].

Iz perspektive obrane napadi u fazi treniranja posebno su zahtjevni jer njihovo otkrivanje često traži kombinaciju provjere podrijetla podataka, analize anomalija, robusnog agregiranja, posebnih metoda skeniranja modela i strožeg upravljanja opskrbnim lancem. Drugim riječima, problem nije samo u samom modelu, nego u cjelokupnom procesu kojim model nastaje.

4. Napadi u fazi inferencije

Napadi u fazi inferencije često su operativno privlačniji od napada na treniranje jer ne traže izravan pristup internim resursima organizacije. Dovoljan je pristup aplikaciji, aplikacijskom sučelju (API-ju) ili nekom drugom sučelju preko kojega model prima upite i vraća rezultate. To znači da su upravo javno izloženi servisi, *chatbotovi*, sustavi preporuke i sigurnosni klasifikatori pod najvećim pritiskom. U takvim scenarijima granica između zlouporabe aplikacije i napada na model postaje tanka, pa se obrana mora graditi istodobno na razini modela, servisa i upravljanja pristupom.

Napadi u fazi inferencije ne mijenjaju model izravno, nego iskorištavaju njegovo ponašanje tijekom uporabe. To ih čini osobito relevantnima za produkcijske sustave dostupne preko aplikacijskog sučelja, korisničkih sučelja ili ugrađenih inteligentnih funkcija u većim aplikacijama. U takvim scenarijima napadač često nema pristup parametrima modela, ali ipak može manipulativno oblikovati ulaze ili kroz upite prikupljati informacije o sustavu [5-7].

4.1 Adversarijalni primjeri

Adversarijalni primjeri predstavljaju jedan od najpoznatijih oblika inferencijskih napada. Riječ je o ulazima namjerno perturbiranima tako da model napravi pogrešku, iako je promjena često mala i ljudskom promatraču teško uočljiva. Već su prvi radovi pokazali da duboki modeli mogu biti iznenađujuće osjetljivi na takve perturbacije te su popularizirali jednostavne metode (npr. FGSM [5]) za njihovu konstrukciju. U računalnom vidu posljedice mogu biti pogrešno prepoznavanje objekata, zaobilaženje sigurnosnih klasifikatora ili narušavanje rada autonomnih sustava. U širem smislu, adversarijalni primjeri pokazuju da visoka točnost modela na standardnim testnim podacima nije isto što i robusnost u neprijateljskom okruženju.

Važno obilježje adversarijalnih primjera jest transferabilnost. Napad konstruiran na jednom modelu često djeluje i na drugi model iste ili srodne domene, čak i kada napadač nema uvid u unutarnje parametre cilja. Ta pojava značajno povećava praktičnu prijetnju *black-box* napada, jer napadač može trenirati ili koristiti *surrogate* modele te napad prenijeti na produkcijski sustav [2, 5]. Iz toga proizlazi važna obrambena implikacija: oslanjanje na tajnost arhitekture nije dovoljno jamstvo sigurnosti.

4.2 Krađa modela

Krađa modela (engl. *model extraction* ili *model stealing*) u pravilu se promatra kao napad u kojem protivnik, koristeći pristup predikcijskom sučelju ili drugim obilježjima izvođenja sustava, pokušava rekonstruirati vrijedna svojstva ciljnog modela. Posljedice takvog napada nisu samo gospodarske, u smislu gubitka intelektualnog vlasništva, nego i operativne: jednom kada raspolaže dovoljno vjernim zamjenskim modelom, napadač može detaljnije analizirati granice odluke, razvijati prilagođene adversarijalne primjere ili planirati napade na privatnost [6].

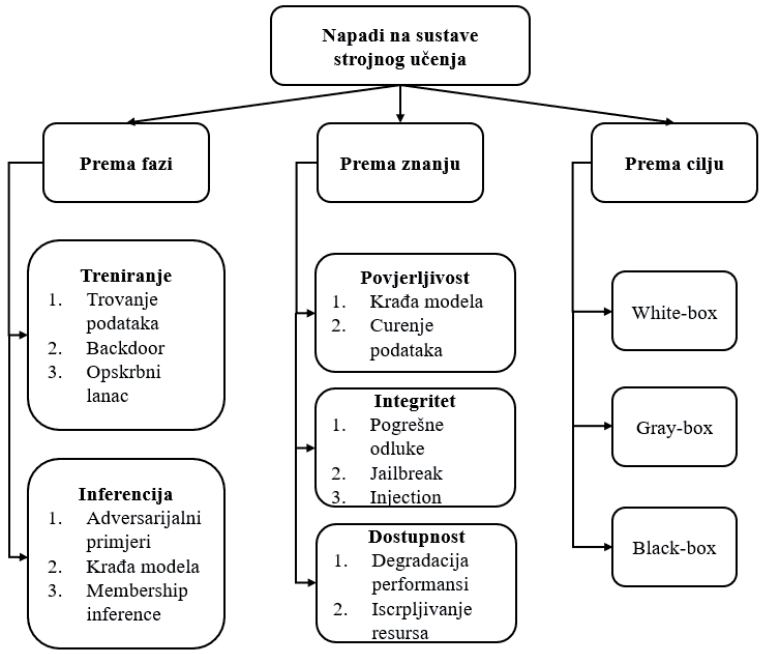
U tom je kontekstu korisno razlikovati krađu modela u širem smislu od krađe vjernosti ponašanja (engl. *fidelity stealing*) u užem smislu. U prvom slučaju protivnik nastoji rekonstruirati funkcionalna, strukturna ili parametarska svojstva modela, dok je u drugom primarni cilj izgraditi zamjenski model koji što vjernije oponaša ulazno-izlazno ponašanje izvornog sustava. Ta razlika nije samo terminološka. U mnogim praktičnim scenarijima napadaču nije nužno potreban isti model kao izvornik; dovoljno je da dobije model koji se ponaša dovoljno slično da omogući daljnje napade ili zaobiđe tržišnu prednost vlasnika izvornog modela. Štoviše, krađa modela ne odvija se isključivo kroz javno izloženi API. Model se može rekonstruirati i analizom elektromagnetskog zračenja i vremenskih obilježja izvođenja [13]. Nadalje, parametri ReLU mreža se mogu izvući u polinomnom vremenu upitima nad modelom, oslanjajući se na pristup inspiriran diferencijalnom kriptoanalizom [14].

Takvi rezultati pokazuju da krađa modela nije ograničena samo na kopiranje funkcionalnosti putem aplikacijskog sučelja, nego obuhvaća i sofisticiranije metode rekonstrukcije arhitekture i parametara, što dodatno proširuje površinu napada nad sustavima strojnog učenja.

4.3 Membership inference

Napadi na privatnost predstavljaju dodatnu inferencijsku dimenziju. *Membership inference* pokušava zaključiti je li neki uzorak bio dio trening skupa. Takav napad može biti posebno osjetljiv kada model sadrži tragove o medicinskim zapisima, financijskim podacima ili drugim osobnim informacijama [7]. U nekim scenarijima mogući su i napadi koji pokušavaju rekonstruirati karakteristike trening skupa ili izvesti zaključke o osjetljivim atributima korisnika. Ovi napadi pokazuju da model može postati kanal curenja informacija, čak i kada izravni pristup originalnim podacima nije moguć.

Slika 2 vizualno sažima višedimenzionalnu taksonomiju napada na sustave strojnog učenja. Slika grupira prijetnje prema fazi životnog ciklusa, primarnom sigurnosnom cilju, te znanju napadača.



Slika 2: Taksonomija napada na sustave strojnog učenja

5. Sigurnost velikih jezičnih modela (LLM)

Sigurnost velikih jezičnih modela ne može se promatrati samo kao proširenje sigurnosti klasičnih ML sustava. Razlog nije samo njihova veličina ili generativna sposobnost, nego činjenica da djeluju u izrazito otvorenom semantičkom okruženju. Ulazi više nisu strogo strukturirani, izlazi nisu unaprijed ograničeni na nekoliko klasa, a model je sve češće povezan s dokumentima, vanjskim servisima i alatima koji mu omogućuju djelovanje izvan pukog generiranja teksta.

Veliki jezični modeli uvode novu klasu sigurnosnih izazova zato što prirodni jezik služi istodobno kao nositelj sadržaja i kao mehanizam upravljanja ponašanjem modela. U klasičnom softveru razdvajanje između podataka i uputa tipično je razmjerno jasno; kod LLM aplikacija ta je granica bitno slabije određena. Posljedica je da tekst koji bi aplikacija trebala tretirati kao običan podatak model može protumačiti kao uputu. Upravo na toj napetosti između sadržaja i instrukcije temelji se velik dio suvremenih sigurnosnih problema LLM sustava [3, 4].

Uz namjerne napade, kod velikih jezičnih modela valja razmotriti i halucinacije, odnosno situacije u kojima model generira netočne, izmišljene ili neosnovane tvrdnje

koje su predstavljene s visokim stupnjem uvjerljivosti. Za razliku od *prompt injection* i *jailbreak* napada, halucinacije u pravilu ne proizlaze iz aktivnog djelovanja napadača, nego iz ograničenja samog modela, nesigurnosti u generiranju, nedostatka pouzdanog uporišta u podacima ili neadekvatnog povezivanja s vanjskim izvorima znanja. Ipak, njihove posljedice mogu biti sigurnosno vrlo relevantne, osobito kada se izlaz modela koristi u odlučivanju, radu s alatima ili sigurnosno osjetljivim domenama.

5.1 *Prompt injection*

Prompt injection nastaje kada napadač u korisnički unos, dokument, web-stranicu, e-poštu ili drugi izvor umetne upute koje preusmjeravaju ponašanje modela izvan predviđene politike aplikacije. Temelj problema jest to što veliki jezični modeli obrađuju instrukcije i podatke unutar istog tekstualnog kanala, pa zlonamjerni sadržaj može biti pogrešno protumačen kao legitimna naredba. Napad može biti izravan, kada napadač komunicira neposredno s modelom, ili neizravan, kada se zlonamjerna uputa nalazi u sadržaju koji model naknadno obrađuje, primjerice tijekom pretraživanja weba, rada s dokumentima ili obrade elektroničke pošte [3, 4, 15].

5.2 *Jailbreak* napadi

Jailbreak napadi usmjereni su na zaobilaženje sigurnosnih ograda modela i navođenje modela da proizvede inače zabranjen ili neusklađen izlaz. U praksi *jailbreak* može koristiti različite strategije, uključujući semantičke obilaske, višekoračne zahtjeve, kodirane upute ili iskorištavanje vanjskog konteksta. Sigurnosna relevantnost takvih napada proizlazi iz činjenice da model može ostati fluentan i funkcionalan, a ipak izgubiti sposobnost sigurnosnog odbijanja štetnih zahtjeva [16, 17].

Iako se u literaturi i praksi pojmovi *prompt injection* i *jailbreaking* često koriste gotovo sinonimno, među njima postoji važna razlika. *Prompt injection* označava tehniku kojom napadač u kontekst koji model obrađuje umeće dodatne upute s ciljem promjene njegova ponašanja. *Jailbreaking*, naprotiv, označava sigurnosni cilj takve manipulacije, odnosno uspješno zaobilaženje ograničenja koja bi model trebala spriječiti u generiranju nedopuštenog ili neusklađenog izlaza. Drugim riječima, *prompt injection* prije svega opisuje mehanizam napada, a *jailbreak* njegov učinak.

LLM-ovi su dodatno rizični kada su integrirani s alatima i vanjskim akcijama. Model koji može čitati dokumente, slati upite drugim servisima, upravljati kalendarom, generirati kod ili pokretati automatizirane radnje širi površinu napada s razine samog modela na razinu cijele aplikacije. U takvim sustavima pogreška modela više nije samo problem kvalitete teksta, nego može imati neposredne sigurnosne posljedice.

Za razliku od ranijih ML sustava koji su često imali usko definirane ulaze i izlaze, LLM aplikacije djeluju u otvorenijem i semantički bogatijem okruženju. Zato sigurnost velikih jezičnih modela treba promatrati kao spoj robusnosti modela, sigurnosti aplikacijske arhitekture i kontrole nad vanjskim alatima i podacima.

6. Dvije studije slučaja iz novije literature

Cilj ovog poglavlja je prikazati dva konkretna slučaja iz novije literature. Te studije slučaja zajedno ilustriraju dvije važne stvari. Prvo, kod sigurnosti strojnog učenja problem često nije samo otkriti ranjivost, nego i vjerodostojno procijeniti koliko je neka obrana doista učinkovita. Drugo, kod velikih jezičnih modela sigurnosno ponašanje nije nužno ravnomjerno raspodijeljeno kroz sve parametre modela, nego se može koncentrirati u manjem broju unutarnjih mehanizama.

6.1 Studija slučaja: obrane od backdoor napada

Prva studija slučaja odnosi se na obrane od *backdoor* napada. Noviji sistematizacijski rad [10] pokazuje da je upravo način na koji se evaluiraju obrane jedna od najslabijih točaka literature. Analizom velikog broja radova autori pokazuju da se učinkovitost obrana snažno mijenja ovisno o odabranim eksperimentalnim postavkama, modelu prijetnje, arhitekturi i skupu podataka.

Važnost ovog nalaza nadilazi samu domenu *backdoor* napada. On pokazuje da pitanje nije samo koja je obrana najbolja, nego i kako uopće treba vrednovati tvrdnju da je model obranjen. Ako su metrike, pretpostavke i scenariji napada nekonzistentni, tada i usporedba obrana postaje metodološki nesigurna. Posljedica nije samo akademska neurednost, nego i praktičan rizik: obrana koja se u jednoj postavci čini vrlo učinkovitom može u realnijem scenariju pružiti znatno manju razinu zaštite [10].

Iz toga slijedi da obrane od *backdoor* napada treba promatrati kao dio šireg programa sigurnosnog osiguranja kvalitete modela. Potrebno je testirati više scenarija napada, više arhitektura, paziti na kompromis između sigurnosti i korisnosti te jasno navesti koje pretpostavke obrana doista pokriva. Evaluacija tako prestaje biti pomoćna istraživačka aktivnost i postaje sastavni dio same sigurnosne argumentacije.

6.2 Studija slučaja: *jailbreak* napadi koji ciljaju sigurnosno relevantne neurone

Druga studija slučaja odnosi se na *jailbreak* napade koji ciljaju sigurnosno relevantne neurone i eksperte u LLM-ovima. Osnovna premisa je da sigurnosno usklađeni

veliki jezični modeli dio sigurnosnog ponašanja temelje na razmjerno uskom skupu unutarnjih komponenti. Referenca [18] pokazuje da ciljana deaktivacija malog broja neurona može znatno povećati uspješnost *jailbreak* napada.

Ta se ideja dodatno pojačava kod *Mixture-of-Experts* modela. Reference [19] i [20] pokazuju da se sigurnosno relevantna ponašanja, poput odbijanja štetnih zahtjeva, u MoE modelima često koncentriraju u malom broju eksperata ili obrazaca usmjeravanja. Posljedično, napadač može narušiti sigurnosno ponašanje modela bez razmjerno velikog utjecaja na njegovu opću fluentnost i uporabljivost.

Zajednička poruka ovih radova jest da sigurnost LLM-ova nije nužno ravnomjerno raspodijeljena kroz parametre modela. Ako je sigurnosna funkcija koncentrirana u malom broju neurona, eksperata ili obrazaca usmjeravanja, tada obrana ne smije ostati ograničena na filtriranje *promptova* i izlaza. Potrebno je razumjeti i kako je sigurnosna funkcija raspoređena unutar same arhitekture te koliko je takva raspodjela robusna na ciljane intervencije [18-20].

7. Obrambeni pristupi

Praktično gledano, najzreliji pristup obrani jest uvođenje višeslojne strategije. Ona kombinira podatkovne kontrole prije treniranja, robusnije metode učenja, provjere ponašanja modela tijekom evaluacije i nadzor nad produkcijskim radom. Takav pristup odgovara logici obrane u dubini iz klasične kibernetičke sigurnosti, ali je u kontekstu strojnog učenja dodatno složen jer mora obuhvatiti i statističko ponašanje modela, a ne samo pristupne i mrežne kontrole.

Ne postoji univerzalna obrana protiv svih napada na sustave strojnog učenja. Razlog tome nije samo raznolikost napada, nego i činjenica da se površina napada proteže preko podataka, procesa treniranja, modela, aplikacijskog sloja, infrastrukture i organizacijskih postupaka. Zbog toga je korisnije govoriti o višeslojnoj obrani nego o pojedinačnim obrambenim metodama [1, 2].

7.1 Obrane od napada trovanjem

Za napade trovanja korisne su mjere poput provjere kvalitete i podrijetla podataka, detekcije anomalija tijekom treniranja, nadzora nad izvorima podataka i robusnih agregacijskih pravila u distribuiranim postavkama. U federiranom učenju to može uključivati procjenu vjerodostojnosti klijenata, detekciju odstupanja u lokalnim ažuriranjima i arhitekturne mehanizme koji otežavaju unošenje *backdoor* napada [11]. Međutim, važno je naglasiti da ni ove mjere same po sebi nisu dovoljne: često postoji kompromis između privatnosti, skalabilnosti i robusnosti.

7.2 Obrane od adversarijalnih primjera

Protiv adversarijalnih primjera najčešće se ističe adversarijalno treniranje, koje model izlaže napadački generiranim perturbacijama tijekom učenja kako bi povećalo robusnost [5]. Uz to se koriste i detekcija sumnjivih ulaza, preoblikovanje ulaza, certifikacijske metode i robusnije arhitekture. Ipak, iskustvo iz literature pokazuje da su mnoge obrane osjetljive na adaptivne napadače, odnosno da se učinkovitost obrane može značajno smanjiti kada napadač modelira obranu kao dio napadačkog procesa [2]. Stoga je u ovom području posebno važno provoditi evaluaciju protiv jakih i prilagodljivih protivnika.

7.3 Obrane od krađe modela i napada na privatnost

Za krađu modela i napade na privatnost važne su minimizacija izlaza, ograničavanje učestalosti i obrasca upita, praćenje anomalija u uporabi aplikacijskog sučelja, vodeni žigovi ili otisci modela te, u nekim scenarijima, diferencijalna privatnost [6, 7]. Važno je naglasiti da uklanjanje detaljnih vjerojatnosti iz izlaza može otežati određene napade, ali najčešće nije dostatno kao samostalna obrana.

7.4 Obrane za LLM-ove

Kod LLM-ova obrana dodatno mora obuhvatiti aplikacijsku arhitekturu. Dobre prakse uključuju strogo razdvajanje sustavskih uputa od neprovjerenog vanjskog sadržaja, označavanje izvora konteksta, izolirano izvođenje alata, validaciju izlaza prije izvršenja akcija, načelo najmanjih privilegija i protivničko testiranje *promptova* i radnih tokova [3, 4].

Noviji radovi o sigurnosno relevantnim neuronima i ekspertima dodatno sugeriraju da buduće obrane možda moraju postati i arhitekturno svjesne. Ako je sigurnost koncentrirana u uskom broju parametara ili eksperata, tada samo filtriranje ulaza i izlaza možda neće biti dovoljno. Obrana će morati uključiti i metode za analizu unutarnje raspodjele sigurnosnih funkcija te za sprečavanje njihove krhke koncentracije [18-20].

Tablica 1: Glavne skupine prijetnji, tipični ciljevi napadača i reprezentativni obrambeni pristupi

Prijetnja	**Faza**	**Primarni cilj**	**Tipična posljedica**	**Reprezentativne obrane**
Trovanje podataka	Treniranje	Integritet / dostupnost	Smanjenje točnosti modela ili namjerno oblikovanje granice odluke	Provjera podrijetla podataka, detekcija anomalija, robusno treniranje
Backdoor napadi	Treniranje	Integritet	Skriveno zlonamjerno ponašanje aktivirano okidačem	Skeniranje modela, filtriranje sumnjivih uzoraka, višescenarijska evaluacija obrana
Adversarijalni primjeri	Inferencija	Integritet	Pogrešna klasifikacija ili zaobilaženje detekcije	Adversarijalno treniranje, detekcija sumnjivih ulaza, robusna evaluacija
Krađa modela	Inferencija	Povjerljivost	Rekonstrukcija modela i priprema sekundarnih napada	Ograničenje izlaza, ograničavanje učestalosti upita, nadzor API-ja, vodeni žigovi modela
Napadi na privatnost	Inferencija	Povjerljivost	Otkrivanje članstva u trening-skupu ili curenje osjetljivih informacija	Minimizacija izlaza, diferencijalna privatnost, kontrola pristupa i revizija upita
Prompt injection i *jailbreak*	Inferencija / aplikacijski sloj	Integritet / povjerljivost	Zaobilaženje politike modela i manipulacija ponašanjem aplikacije	Razdvajanje uputa i podataka, izolirano izvođenje alata, validacija izlaza, protivničko testiranje

Tablica 1 sažima glavne skupine prijetnji, tipične ciljeve napadača i reprezentativne obrambene pristupe. Njezina je svrha ponuditi brzu referencu koja povezuje taksonomiju, praktične posljedice i obrambene prioritete.

8. Regulatorni okvir

Regulatorni zahtjevi postupno pretvaraju sigurnosne kontrole iz preporučene dobre prakse u element organizacijske dokazivosti i odgovornosti. U europskom kontekstu najvažniji regulatorni dokument je EU AI Act [21]. Riječ je o horizontalnom okviru koji uvodi pristup temeljen na riziku te povezuje tehničku pouzdanost, sigurnost, dokumentaciju i upravljanje tijekom cijelog životnog ciklusa AI sustava. Za sigurnost strojnog učenja važno je to što AI Act ne promatra problem samo kroz prizmu opće kibernetičke sigurnosti, nego šire kroz robusnost, kvalitetu podataka, transparentnost, ljudski nadzor i naknadno praćenje sustava.

Za visokorizične AI sustave posebno su važni zahtjevi koji se odnose na upravljanje rizicima, tehničku dokumentaciju, zapisivanje događaja, kvalitetu podataka i razinu točnosti, robusnosti i kibernetičke sigurnosti [21]. Iako AI Act ne koristi jezik adversarijalnog strojnog učenja kao glavni organizacijski princip, njegovi zahtjevi stvaraju regulatorni pritisak da organizacije ozbiljnije pristupe pitanjima trovanja podataka, validacije modela, upravljanja promjenama i post-market nadzora. To sigurnost ML-a postupno pretvara iz "dobre prakse" u element regulatorne usklađenosti.

Kada ML sustav obrađuje osobne podatke, ključan regulatorni okvir ostaje GDPR [22]. Napadi poput *membership inference* ili rekonstrukcije podataka relevantni su ne samo tehnički nego i pravno, jer pokazuju da model može postati kanal kroz koji se neizravno otkrivaju informacije o pojedincima iz trening skupa. GDPR traži zakonitu osnovu obrade, minimizaciju podataka, sigurnost obrade i odgovarajuće tehničke i organizacijske mjere. U tom smislu privatnost i sigurnost u ML-u nisu odvojene teme, nego se preklapaju na razini modela, izlaza i upravljanja pristupom.

Za organizacije koje razvijaju i uvode ML sustave u praksu regulatorni okvir znači da tehničke odluke trebaju biti dokumentirane i dokazive. Nije dovoljno tvrditi da je model robustan; potrebno je pokazati kako je procijenjen rizik, koje su kontrole provedene, kako se promjene prate nakon implementacije i kako se incidenti dokumentiraju. Takva dokazivost postaje osobito važna kada sustav djeluje u osjetljivim domenama ili kada utječe na prava i interese pojedinaca.

Uz obvezujuće propise važnu ulogu imaju i dobrovoljni okviri poput NIST AI RMF-a [1]. Oni nude strukturiran pristup identifikaciji, procjeni, upravljanju i praćenju AI rizika te pomažu povezati tehničke mjere s organizacijskim procesima. U praksi se regulatorna usklađenost i tehnička sigurnost ne bi smjele promatrati kao odvojeni

projekti. Upravo suprotno, učinkovit program sigurnosti strojnog učenja u pravilu mora kombinirati tehničku robusnost, upravljanje rizikom, dokumentiranje i kontinuirani nadzor.

9. Rasprava i zaključak

Sigurnost strojnog učenja razlikuje se od klasične informacijske sigurnosti po tome što je površina napada djelomično ugrađena u samu logiku učenja. Model uči iz podataka, generalizira statistički i donosi odluke u okruženjima koja se mogu mijenjati ili namjerno oblikovati. Zbog toga sigurnost ML-a nužno povezuje kibernetičku sigurnost, statistiku, softversko inženjerstvo, upravljanje rizikom i regulatorno razumijevanje.

Prijetnje su raznolike: od trovanja podataka i *backdoor* napada u fazi treniranja, preko adversarijalnih primjera, krađe modela i napada na privatnost u fazi inferencije, pa sve do *prompt injection*, *jailbreak* i halucinacija u velikim jezičnim modelima. Upravo ta raznolikost pokazuje da se sigurnost strojnog učenja ne može svesti na jedan model prijetnje ni na jednu obrambenu tehniku. Potrebno je istodobno promatrati model, podatke, aplikacijski sloj i organizacijski kontekst u kojem se sustav koristi.

Zaključno, robusniji i sigurniji sustavi strojnog učenja neće nastati jednom tehnikom, nego kombinacijom tehničkih, organizacijskih i regulatornih mjera kroz cijeli životni ciklus sustava. U budućem razvoju područja posebnu će važnost imati vjerodostojna evaluacija obrana, sigurnost LLM aplikacija povezanih s vanjskim alatima, otpornost na ciljane arhitekturne napade te sposobnost organizacija da sigurnosne tvrdnje učine dokazivima i provjerljivima.

10. Literatura

[1] National Institute of Standards and Technology (NIST): Artificial Intelligence Risk Management Framework (AI RMF 1.0). NIST AI 100-1 (2023), *Dostupno na*: https://nvlpubs.nist.gov/nistpubs/ai/nist.ai.100-1.pdf, *Pristupljeno*: 2026-03-15

[2] Vassilev, A.; Oprea, A., Fordyce, A., Anderson, H., Davies, X., Hamin, M.: Adversarial Machine Learning: A Taxonomy and Terminology of Attacks and Mitigations, NIST AI 100-2e2025 (2025), *Dostupno na*: https://doi.org/10.6028/NIST.AI.100-2e2025, *Pristupljeno*: 2026-03-15

[3] OWASP Foundation: LLM Prompt Injection Prevention Cheat Sheet, *Dostupno na*: https://cheatsheetseries.owasp.org/cheatsheets/LLM_Prompt_Injection_Prevention_Cheat_Sheet.html, *Pristupljeno*: 2026-03-15

[4] OWASP GenAI Security Project: LLM01:2025 Prompt Injection. *Dostupno na*: https://genai.owasp.org/llmrisk/llm01-prompt-injection/, *Pristupljeno*: 2026-03-15

[5] Goodfellow, I. J.; Shlens, J., Szegedy, C.: Explaining and Harnessing Adversarial Examples. *Zbornik 3rd International Conference on Learning Representations, ICLR 2015,* svibanj 2015, (2015). *Dostupno na*: https://arxiv.org/abs/1412.6572, *Pristupljeno*: 2026-03-15

[6] Tramèr, F.; Zhang, F., Juels, A., Reiter, M. K., Ristenpart, T.: Stealing Machine Learning Models via Prediction APIs. *Zbornik 25th USENIX Conference on Security Symposium (SEC'16),* str. 601-618, USENIX Association, SAD (2016). *Dostupno na*: https://arxiv.org/abs/1609.02943, *Pristupljeno*: 2026-03-15

[7] Shokri, R.; Stronati, M., Song, C., Shmatikov, V.: Membership Inference Attacks Against Machine Learning Models. *Zbornik 2017 IEEE Symposium on Security and Privacy (SP),* str. 3-18, San Jose, SAD (2017). *Dostupno na*: https://arxiv.org/abs/1610.05820, *Pristupljeno*: 2026-03-15

[8] Papernot, N.; McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., Swami, A.: Practical Black-Box Attacks against Machine Learning. *Zbornik 2017 ACM on Asia Conference on Computer and Communications Security (ASIA CCS '17),* str. 506-519, Association for Computing Machinery (2017). *Dostupno na*: https://arxiv.org/abs/1602.02697, *Pristupljeno*: 2026-03-15

[9] Biggio, B.; Nelson, B., Laskov, P.: Poisoning Attacks against Support Vector Machines. *Zbornik 29th International Coference on International Conference on Machine Learning (ICML'12),* str. 1467-1474, Omnipress, Madison, SAD (2012). *Dostupno na*: https://arxiv.org/abs/1206.6389, *Pristupljeno*: 2026-03-15

[10] Abad, G.; Krček, M., Koffas, S., Tajalli, B., Arazzi, M., Riaño, R., Xu, X., Liu, Z., Nocera, A., Picek, S.: SoK: The Last Line of Defense: On Backdoor Defense Evaluation. arXiv preprint arXiv:2511.13143 (2025). *Dostupno na*: https://arxiv.org/abs/2511.13143, *Pristupljeno*: 2026-03-15

[11] Abad, G.; Picek, S., Ramírez-Durán, V. J., Urbieta, A.: On the Security & Privacy in Federated Learning. arXiv:2112.05423 (2021). *Dostupno na*: https://arxiv.org/abs/2112.05423, *Pristupljeno*: 2026-03-15

[12] MITRE: ATLAS – Adversarial Threat Landscape for Artificial-Intelligence Systems. *Dostupno na*: https://atlas.mitre.org/, *Pristupljeno*: 2026-03-15

[13] Batina, L.; Bhasin, S., Jap, D., Picek, S.: CSI NN: Reverse Engineering of Neural Network Architectures Through Electromagnetic Side Channel. *Zbornik 28th USENIX Conference on Security Symposium (SEC'19),* str. 515-532, USENIX Association, SAD (2019). *Dostupno na*: https://www.usenix.org/conference/usenixsecurity19/presentation/batina, *Pristupljeno*: 2026-03-15

[14] Canales-Martínez, I. A.; Chávez-Saab, J., Hambitzer, A., Rodríguez-Henríquez, F., Satpute, N., Shamir, A.: Polynomial Time Cryptanalytic Extraction of Neural Network Models. *Zbornik Advances in Cryptology – EUROCRYPT 2024,* Joye, M., Leander, G., str. 3-33, svibanj, 2024, Springer, Cham (2024). *Dostupno na*: https://eprint.iacr.org/2023/1526, *Pristupljeno*: 2026-03-15

[15] Greshake, K.; Abdelnabi, S., Mishra, S., Endres, C., Holz, T., Fritz, M.: Not what you've signed up for: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection. *Zbornik 16th ACM Workshop on Artificial Intelligence and Security (AISec '23*), str. 79-90, Association for Computing Machinery, New York, SAD (2023). *Dostupno na*: https://arxiv.org/abs/2302.12173, *Pristupljeno*: 2026-03-15

[16] Andriushchenko, M.; Croce, F., Flammarion, N.: Jailbreaking Leading Safety-Aligned LLMs with Simple Adaptive Attacks. arXiv preprint arXiv:2404.02151 (2024). *Dostupno na*: https://arxiv.org/abs/2404.02151, *Pristupljeno*: 2026-03-15

[17] Zou, A.; Wang, Z., Carlini, N., Nasr, M., Kolter, J. Z., Fredrikson, M.: Universal and Transferable Adversarial Attacks on Aligned Language Models. arXiv preprint arXiv:2307.15043 (2023). *Dostupno na*: https://arxiv.org/abs/2307.15043, *Pristupljeno*: 2026-03-15

[18] Wu, L.; Behrouzi, S., Rostami, M., Thang, M., Picek, S., Sadeghi, A.-R.: NeuroStrike: Neuron-Level Attacks on Aligned LLMs. arXiv preprint arXiv:2509.11864 (2025). *Dostupno na*: https://arxiv.org/abs/2509.11864, *Pristupljeno*: 2026-03-15

[19] te Lintelo, J.; Wu, L., Picek, S.: Large Language Lobotomy: Jailbreaking Mixture-of-Experts via Expert Silencing. arXiv preprint arXiv:2602.08741 (2026). *Dostupno na*: https://arxiv.org/abs/2602.08741, *Pristupljeno*: 2026-03-15

[20] Wu, L.; Behrouzi, S., Rostami, M., Picek, S., Sadeghi, A.-R.: GateBreaker: Gate-Guided Attacks on Mixture-of-Expert LLMs. arXiv preprint arXiv:2512.21008 (2025). *Dostupno na*: https://arxiv.org/abs/2512.21008, *Pristupljeno*: 2026-03-15

[21] European Parliament and Council of the European Union: Regulation (EU) 2024/1689 of 13 June 2024 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). *Dostupno na*: https://eur-lex.europa.eu/eli/reg/2024/1689/oj/eng, *Pristupljeno*: 2026-03-15

[22] European Parliament and Council of the European Union: Regulation (EU) 2016/679 of 27 April 2016 (General Data Protection Regulation). *Dostupno na*: https://eur-lex.europa.eu/eli/reg/2016/679/oj/eng, *Pristupljeno*: 2026-03-15