Kauno technologijos universitetas

Informatikos fakultetas

Duomenų viliojimo atakų komunikacijos programose aptikimo metodas

Baigiamasis magistro krypties studijų projektas

Modestas Krištaponis

Projekto autorius

Vyresn. Lektorius, dr. Gedeiminas Činčikas

Vadovas

Kaunas, 2025

Kauno technologijos universitetas

Informatikos fakultetas

Duomenų viliojimo atakų komunikacijos programose aptikimo metodas

Baigiamasis magistro krypties studijų projektas

Informacijos ir informacinių technologijų sauga (6211BX008)

Modestas Krištaponis

Projekto autorius

Vyresn. Lektorius, dr. Gedeiminas Činčikas

Vadovas

Kaunas, 2025

Kauno technologijos universitetas

Informatikos fakultetas

Modestas Krištaponis

Duomenų viliojimo atakų komunikacijos programose aptikimo metodas

Akademinio sąžiningumo deklaracija

Patvirtinu, kad:

1. baigiamąjį projektą parengiau savarankiškai ir sąžiningai, nepažeisdama(s) kitų asmenų autoriaus ar kitų teisių, laikydamasi(s) Lietuvos Respublikos autorių teisių ir gretutinių teisių įstatymo nuostatų, Kauno technologijos universiteto (toliau – Universitetas) intelektinės nuosavybės valdymo ir perdavimo nuostatų bei Universiteto akademinės etikos kodekse nustatytų etikos reikalavimų;

2. baigiamajame projekte visi pateikti duomenys ir tyrimų rezultatai yra teisingi ir gauti teisėtai, nei viena šio projekto dalis nėra plagijuota nuo jokių spausdintinių ar elektroninių šaltinių, visos baigiamojo projekto tekste pateiktos citatos ir nuorodos yra nurodytos literatūros sąraše;

3. įstatymų nenumatytų piniginių sumų už baigiamąjį projektą ar jo dalis niekam nesu mokėjęs (-usi);

4. suprantu, kad išaiškėjus nesąžiningumo ar kitų asmenų teisių pažeidimo faktui, man bus taikomos akademinės nuobaudos pagal Universitete galiojančią tvarką ir būsiu pašalinta(s) iš Universiteto, o baigiamasis projektas gali būti pateiktas Akademinės etikos ir procedūrų kontrolieriaus tarnybai nagrinėjant galimą akademinės etikos pažeidimą.

Modestas Krištaponis

Patvirtinta elektroniniu būdu

Autorius Krištaponis, Modestas. Duomenų viliojimo atakų komunikacijos programose aptikimo metodas. Magistro krypties studijų baigiamasis projektas / vadovas Vyresn. Lektorius, dr. Gedeiminas Činčikas; Kauno technologijos universitetas, informatikos fakultetas.

Studijų kryptis ir sritis (studijų krypčių grupė): Informatikos inžinerija, informatikos mokslai.

Reikšminiai žodžiai: duomenų viliojimas, kibernetinė ataka, socialinė inžinerija, mašininis mokymas, teksto analizė, nuorodų analizė, failų analizė.

Kaunas, 2025. 61p.

Santrauka

Modernios ir šiuolaikiškos sistemos nėra pakankamai apsaugotos nuo kibernetinių nusikaltimų, dažnai silpnoji saugumo grandis yra žmogiškasis faktorius. Nusikaltėliai siunčia skirtingas kenkėjiškas žinutes, nuorodas, failus bandant gauti jautrią informaciją iš aukos. Šio baigiamojo magistro projekto tikslas yra ištirti duomenų viliojimo atakas komunikacijos platformose ir sukurti metodą, kuris atpažintų šias atakas. Šiuo projektu siekiama automatizuoti siunčiamų žinučių analizavimą realiu laiku ir užtikrinti, jog duomenų viliojimo atakos būtų atpažintos. Siūlomas metodas analizuoja failus, nuorodas naudodamas nustatytų taisyklių rinkinį ir tekstą naudojant mašininį mokymą atpažinti duomenų viliojimo atakas. Eksperimentiniai rezultatai pademonstruoja metodo aukštą efektyvumą ir tikslumą nustatant duomenų viliojimo kibernetines atakas.

Krištaponis, Modestas. A Method For Detecting Phishing Attacks In Communication Applications. Master's Final Degree Project / supervisor Senior Lect. Dr. Gedeiminas Činčikas; Informatics Faculty, Kaunas University of Technology.

Study field and area (study field group): Informatics Engineering, Computing.

Keywords: phishing attacks, link analysis, text analysis, file analysis, machine learning, communication platforms

Kaunas, 2025. 61p.

Summary

Modern and up-to-date systems are not sufficiently protected against cybercrime, phishing attacks have become a significant threat in online communication platforms. Criminals send different malicious messages, links, files in an attempt to obtain sensitive information from the victim. The aim of this final Master's project is to investigate data gathering attacks on communication platforms and to develop a method to detect malicious content. The aim of this project is to automate the analysis of outgoing messages in real time and to ensure that data solicitation attacks are detected. The proposed method analyses files, links using a set of defined rules and text using machine learning to identify data gathering attacks. Experimental results demonstrate the method's high efficiency and accuracy in detecting phishing cyber attacks.

Turinys

Paveikslų sąrašas

Lentelių sąrašas

Santrumpų ir terminų sąrašas

Santrumpos:

DI – Dirbtinis intelektas

MFA – daugiafaktorinis autentifikavimo mechanizmas (angl. *Multi-factor authentication*)

NLP – Natūralios kalbos apdorojimas (angl. *Natural Language Processor*)

SPF – Siuntėjo politikos sistema (angl. *Sender Policy Framework*)

TF-IDF – Terminų dažnio ir atvirkštinio dokumento dažnio (angl. *Term Frequency-Inverse Document Frequency*)

FN – Klaidingi neigiami rezultatai (angl. *False Negative*)

HTTPS – Saugus hiperteksto perdavimo protokolas (angl. *HyperText Transfer Protocol Secure*)

HTTP – Hiperteksto perdavimo protokolas (angl. *HyperText Transfer Protocol*)

API – Programų sąsaja (angl. *Application Programming Interface*)

SSL – Saugių lizdų lygmuo (angl. *Secure Sockets Layer*)

CA – Sertifikavimo institucija (angl. *Certificate Authority*)

MIME – Daugiatikslės internetinio pašto plėtiniai (angl. *Multipurpose Internet Mail Extensions*)

Įvadas

Šiame magistro darbe nagrinėjama, kaip dėl vis sudėtingesnių kibernetinių grėsmių reikia naujoviškų gynybos mechanizmų. Darbas orientuotas į duomenų viliojimo tipo atakas. Socialinės inžinerijos atakos tapo viena iš labiausiai paplitusių efektyvių kibernetinių grėsmių, ypač bendravimo programose. Šios platformos, plačiai naudojamos tiek profesiniam, tiek asmeniniam bendravimui, tapo pagrindiniais užpuolikų, besinaudojančių patogiomis sąsajomis ir integracijos galimybėmis, taikiniais. Vykdant duomenų viliojimo atakas, naudotojai apgaule priverčiami atskleisti jautrią informaciją, įskaitant įgaliojimus ir finansinius duomenis, ir taip padaroma didelė finansinė žala ir žala reputacijai. Naudotojai gali įrašyti kenkėjišką kodą ar nutekinti savo slaptažodžius, prisijungimus to patys nežinodami. Nepaisant kibernetinio saugumo pažangos, dinamiškas ir prisitaikantis duomenų viliojimo (angl. *phishing*) taktikos pobūdis reikalauja novatoriškų sprendimų, kad būtų galima veiksmingai kovoti su šiomis grėsmėmis. Šiame projekte sprendžiamas itin svarbus poreikis sukurti patikimas, ryšių programoms pritaikytas sukčiavimo aptikimo ir prevencijos sistemas.

Projekto naujumas ir aktualumas

Sudėtingų sukčiavimo metodų, nukreiptų būtent į ryšių platformas, atsiradimas yra esminė dabartinės kibernetinio saugumo apsaugos spraga. Nors atlikta daug tyrimų apie elektroniniu paštu vykdomą sukčiavimą, unikalios realaus laiko pranešimų platformų savybės, įskaitant sutrumpintus nuorodų adresus, neformalius bendravimo modelius ir integraciją su trečiųjų šalių programomis, sukuria naujus atakų vektorius, kuriems reikia naujoviškų aptikimo metodų. Šis projektas skirtas šiai didėjančiai grėsmei spręsti, kuriant specializuotus aptikimo mechanizmus, skirtus šiuolaikinėms ryšių platformoms, naudojama kelių sluoksnių patikra.

Tikslas ir uždaviniai

Darbo tikslas – sukurti pažangų duomenų viliojimo atakos aptikimo metodą.

Uždaviniai:
1. atlikti duomenų viliojimo atakų analizę;
2. išsiaiškinti, kodėl duomenų viliojimo atakos yra kenksmingos;
3. nustatyti daugiavektorinių atakų požymius;
4. išanalizuoti esančius duomenų viliojimo atakų aptikimo sprendimus;
5. panaudoti mašininio mokymo modelį teksto analizei, failų ir nuorodų analizę, kad įvertinti bendrą žinutės kenkėjiškumą;
6. įgyvendinti duomenų viliojimo atakų aptikimo algoritmo prototipą ir atlikti eksperimentinius tyrimus;

Dokumento struktūra

Analitinė literatūros dalis, kurioje apžvelgiami skirtingos socialinės inžinerijos atakų tipai ir aptikimo sistemos. Sprendimo koncepcija – joje aptariama kas bus padaryta, priimti sprendimai ir preliminarūs algoritmai ir jų veikimas. Sprendimo prototipas – aprašomas realizuotas prototipas. Eksperimentas – atliktas eksperimentas su rezultatais ir metodo patobulinimas. Išvados ir naudota literatūra.

1. Analizė

Duomenų viliojimo atakose naudojama žmogaus psichologija, kad manipuliuotų asmenimis ir priverstų juos atskleisti konfidencialią informaciją arba atlikti veiksmus, kurie kelia grėsmę saugumui. Dažniausiai pasitaikantys socialinės inžinerijos atakų tipai yra duomenų viliojimas (angl. *phising*), pretekstas (angl. *pretexting*), kibimas (angl. *baiting*“) ir sekimas (angl. *tailating*). Labiausiai paplitusi duomenų viliojimo forma - apgaulingais el. laiškais ar žinutėmis siekiama įkalbėti aukas pateikti konfidencialius duomenis, pavyzdžiui, slaptažodžius ar finansinę informaciją. Prisidengimas – tai melagingo pasakojimo kūrimas, siekiant įgyti pasitikėjimą ir išgauti informaciją, dažnai prisidengiant autoritetingais asmenimis. Viliojimas traukia aukas viliojančiais pasiūlymais, pavyzdžiui, nemokama programine įranga ar dovanomis, kuriose dažnai būna kenkėjiško kodo. Pasinaudojant pasitikėjimu, pavyzdžiui, sekant paskui žmogų į saugomą pastatą, fiziškai įgyjama neteisėta prieiga prie riboto patekimo zonų. Prevencijos sistemose daugiausia dėmesio skiriama technologinių priemonių ir naudotojų švietimo deriniui. Daugiafaktorinis autentiškumo patvirtinimas (angl. *MFA*) mažina riziką reikalaudamas kelių formų patvirtinimo, o pažangūs el. pašto filtravimo ir kovos su kenkėjiškomis programomis sprendimai blokuoja galimas grėsmes. Reguliarios kibernetinio saugumo mokymo programos didina naudotojų informuotumą, suteikdamos asmenims galimybę atpažinti socialinės inžinerijos taktiką ir jai pasipriešinti. Integruodamos technines apsaugos priemones ir sąmoningą budrumą, organizacijos gali gerokai sumažinti šių atakų poveikį.

1.1. Socialinės inžinerijos atakų tipai

1.1.1. Duomenų viliojimas (angl. *phishing*)

Duomenų viliojimas - tai kibernetinė ataka, kai apgaulės būdu, paprastai per apgaulingus el. laiškus, svetaines ar žinutes, asmenys priverčiami pateikti jautrią informaciją arba atlikti veiksmus, kurie kelia grėsmę jų saugumui. Tai tebėra viena iš labiausiai paplitusių ir veiksmingiausių kibernetinių nusikaltimų formų, nes remiasi žmogaus klaidomis, o ne techninėmis spragomis.

Duomenų viliojimo tipai:

Sukčiavimas elektroniniu paštu. Sukčiavimas elektroniniu paštu yra labiausiai paplitusi sukčiavimo forma. Užpuolikai siunčia el. laiškus, kurie atrodo kaip iš teisėtų šaltinių, pavyzdžiui, bankų, interneto paslaugų ar darbdavių. Šiuose el. laiškuose dažnai būna išreikštas skubos jausmas, skatinantis gavėją paspausti nuorodą arba atsisiųsti priedą, dėl kurio gali būti užkrėsta kenkėjiška programine įranga arba pavogti duomenys.

„*Spear Phishing*“. Skirtingai nei bendroji elektroninio pašto apgaulė, „*spear phishing*“ nukreiptas į konkrečius asmenis ar organizacijas. Užpuolikai suasmenina el. laišką naudodami iš socialinės žiniasklaidos ar kitų šaltinių surinktą informaciją, kad pranešimas būtų įtikinamesnis. Pavyzdžiui, užpuolikas gali apsimesti kolega ar patikimu verslo partneriu.

Banginių medžioklė (angl. *whaling*). „*Whaling*“ - tai „*spear phishing*“ rūšis, kurios taikinys yra aukšto rango organizacijos asmenys, pavyzdžiui, generaliniai direktoriai arba finansų direktoriai. Turinys paprastai pritaikomas taip, kad atrodytų kaip svarbus verslo reikalas, dažnai susijęs su suklastotomis sąskaitomis faktūromis, teisiniais klausimais arba vadovų lygio pranešimais.

Duomenų viliojimas SMS žinutėmis (angl. *smishing*). Duomenų viliojimas SMS žinutėmis - tai apgaulingų SMS žinučių siuntimas, kurios atrodo kaip gautos iš teisėtų šaltinių, pavyzdžiui, bankų ar paslaugų teikėjų. Šiose žinutėse dažnai pateikiama nuoroda į suklastotą svetainę, kurioje aukų prašoma įvesti asmeninę informaciją arba atsisiųsti kenkėjišką programinę įrangą.

Duomenų viliojimas balsu (angl. *voice phishing*). Duomenų viliojimas balsu - tai užpuolikų skambučiai telefonu apsimetant teisėtų institucijų atstovais. Skambinantysis gali teigti, kad iškilo problemų su aukos sąskaita, ir telefonu prašyti pateikti asmeninę informaciją arba atlikti mokėjimą.

Kloninis sukčiavimas. Vykdydami klonuotąją apgaulę, užpuolikai sukuria beveik identišką teisėto el. pašto, kurį anksčiau išsiuntė patikimas šaltinis, kopiją. Klonuotame el. laiške pateikiama kenkėjiška nuoroda arba priedas, o užpuolikas jį siunčia iš suklastoto el. pašto adreso, kad jis atrodytų autentiškas.

1.1.2. Impersonacija

Dažniausiai pasitaikanti socialinės inžinerijos ataka - apsimetinėjimas, kai užpuolikai apsimeta patikimais asmenimis ar subjektais, kad apgaule priverstų aukas atskleisti konfidencialią informaciją arba atlikti veiksmus, kurie kelia grėsmę saugumui [4]. Tai metodas, kuriuo pasinaudojama socialiniams ir profesiniams santykiams būdingu pasitikėjimu, siekiant apeiti technines apsaugos priemones. Dažniausiai pasitaikantys scenarijai - tai užpuolikai, apsimetę IT pagalbos darbuotojais, prašantys prisijungimo duomenų, vadovai, nurodantys pavaldiniams pervesti lėšas, arba paslaugų teikėjai, siekiantys patekti į saugias zonas. Apsimetėlių atakos yra ypač veiksmingos, nes jose naudojamasi žmonių polinkiu paklusti autoritetingiems asmenims ir padėti kolegoms. Tokioms atakoms aptikti reikia patikimų tikrinimo procesų, pavyzdžiui, įdiegti griežtus tapatybės tikrinimo protokolus ir daugiafaktorinį autentiškumo patvirtinimą. Be to, mokymo programos, orientuotos į informuotumo apie apsimetinėjimo taktiką didinimą ir skepticizmo kultūros neprašytų prašymų atžvilgiu skatinimą, gali gerokai sumažinti sėkmingų atakų riziką. Suprasdamos psichologinę manipuliaciją, būdingą apsimetinėjimui, ir taikydamos tiek technologines, tiek edukacines atsakomąsias priemones, organizacijos gali geriau apsisaugoti nuo šios klastingos grėsmės.

1.1.3. Stebėjimas per petį (angl. *shoulder surfing*)

Žiūrėjimas per kompiuterio naudotojo petį, kai jis savo įrenginiuose rašo duomenis, pvz., naudotojo vardus ir slaptažodžius [4]. Kenkėjas išsaugo šią informaciją, kad gautų gauti prieigą prie aukos sistemos.

1.1.4. Telefono arba elektroninio pašto atakos

Telefoniniai ir el. pašto sukčiai - plačiai paplitusi socialinės inžinerijos atakų forma - naudojasi ryšio kanalais, kad apgautų aukas ir priverstų jas atskleisti konfidencialią informaciją arba atlikti kompromituojančius veiksmus. Šios atakos dažnai susijusios su apsimetinėjimu, kai užpuolikai apsimeta teisėtais subjektais, pavyzdžiui, bankais, vyriausybinėmis agentūromis ar patikimomis organizacijomis, kad pasinaudotų aukos pasitikėjimu. Telefoniniai sukčiai, arba „*vishing*“ (angl. *voice phishing*), - tai sukčiai, skambinantys aukoms ir naudojantys įtikinamą taktiką arba skubius grasinimus, kad išviliotų konfidencialią informaciją, pavyzdžiui, kredito kortelių duomenis arba socialinio draudimo numerius. Elektroninio pašto sukčiai, įskaitant „*phishing“* ir „*spear-phishing“*, siunčia apgaulingus elektroninius laiškus, kurie atrodo tarsi iš patikimų šaltinių ir kuriais siekiama

apgauti gavėjus, kad šie paspaustų kenkėjiškas nuorodas, atsisiųstų kenksmingus priedus arba atskleistų asmeninę informaciją. Šių atakų veiksmingumą lemia psichologinė manipuliacija, pasinaudojant baime, smalsumu ar autoritetu. Siekdamos kovoti su telefoniniais ir el. pašto sukčiais, organizacijos turėtų įdiegti patikimas el. pašto filtravimo technologijas, reguliariai rengti informuotumo apie saugumą mokymus ir skatinti skeptišką požiūrį į neprašytus pranešimus. Įdiegus daugiafaktorinį autentifikavimą ir skatinant netikėtų užklausų tikrinimą žinomais, patikimais kanalais, galima dar labiau sustiprinti apsaugą nuo šių sudėtingų socialinės inžinerijos taktikų.

1.1.5. „*Quid pro quo*“

„*Quid pro quo*“ - tai gudri socialinės inžinerijos ataka, kurio metu užpuolikai mainais už slaptą informaciją ar prieigą suteikia tariamą atlygį [4]. Užpuolikai naudojasi žmogiškuoju poreikiu atsilyginti už paslaugas, apsimesdami naudingais specialistais ir dažnai teikdami techninę pagalbą ar atnaujinimus mainais į prisijungimo duomenis ar prieigą prie sistemos. Šiomis strategijomis piktnaudžiaujama noru atsilyginti ir manipuliuojama pasitikėjimu, todėl aukos tampa pažeidžiamos. Norint sumažinti tokių išpuolių poveikį, reikia rengti išsamius informuotumo mokymus, skatinti nepasitikėjimą neprašytais pasiūlymais ir nustatyti griežtus prieigos apribojimus bei tikrinimo procedūras, kad būtų sustiprinta kibernetinio saugumo apsauga.

1.1.6. Viliojimo atakos (angl. *Baiting attacks*)

Viliojimo atakos - apgaulinga socialinės inžinerijos forma - vilioja aukas patraukliais pasiūlymais, pavyzdžiui, nemokamu atsisiuntimu arba USB laikmenomis su kenkėjiška programine įranga. Pasinaudodami žmonių smalsumu, įsilaužėliai pasinaudoja nemokamų daiktų vilionėmis, kad pažeistų sistemas ir pavogtų slaptą informaciją. Įprasta taktika - palikti užkrėstas USB laikmenas viešose vietose arba platinti viliojančias nuorodas el. paštu ar socialinėje žiniasklaidoje. Norint užkirsti kelią tokioms atakoms, būtina rengti išsamius saugumo sąmoningumo mokymus, kad naudotojai būtų supažindinti su nežinomų daiktų priėmimo rizika. Organizacijos taip pat turėtų vykdyti griežtą išorinių įrenginių naudojimo politiką ir skatinti nedelsiant pranešti apie įtartiną veiklą, kad sumažintų kibernetinio saugumo pasekmes, kylančias dėl viliojimo atakų.

1.1.7. Viliojimo atakos (angl. *Grooming*)

Vienas iš naujausių socialinės inžinerijos metodų yra viliojimas, kai, naudojant psichologinius metodus ir informacines technologijas, gaunama konfidenciali informacija apie potencialias pedofilijos aukas [8]. Viliojimo metodai daugiausia dėmesio skiria informacinių technologijų, pavyzdžiui, SMS, el. pašto ir telefono, naudojimui. Pokalbis yra labiau spąstai, kuriais siekiama privilioti aukas.

1.1.8. Pretekstas (angl. *pretexting*)

Pretekstas - tai išgalvoto scenarijaus sukūrimas, kad auka būtų priversta atskleisti slaptą informaciją arba atlikti veiksmus, kurie kelia grėsmę saugumui. Pavyzdžiui, užpuolikas gali apsimesti IT specialistu ir prašyti prisijungimo duomenų, prisidengdamas techninės problemos sprendimu. Šis metodas labai priklauso nuo pasitikėjimo ir patikimumo, todėl be elgsenos ar konteksto analizės jį aptikti sudėtinga [8]. Aptikimo sistemos dažnai naudoja anomalijų aptikimo ir elgsenos analizės metodus, kad tokiuose scenarijuose nustatytų neatitikimus.

Įsilaužėliai gauna viešą informaciją iš interneto svetainių, socialinės žiniasklaidos ir telefonų knygų, kad susietų informaciją apie veiklą, kurioje auka dalyvavo arba turi galimybę dalyvauti. Pretekstas - tai sukčiavimo technika, kuri remiasi abipusiu bendravimu; vyksta pokalbis su auka. Pokalbiai gali vykti siūlant darbą ar pagalbą, prašant asmeninės informacijos arba patvirtinant, kad gautas prizas.

1.1.9. Profilio klonavimas (angl. *Profile Cloning*)

Šis metodas naudoja viešai prieinamą informaciją, pvz., socialinę žiniasklaidą, ir veikia kaip asmuo, kurio profilis klonuotas, kad gautų svarbios informacijos iš kitų žmonių. Vieša informacija gali būti vaizdai, vaizdo įrašai, pilni vardai ir net pokalbių stiliai, gauti iš socialinės žiniasklaidos. Jei įsilaužėlis nori gauti informacijos iš aukos, jis bandys rasti būdą, kaip pateikti įsilaužėliui reikalingą informaciją [8]. Įsilaužėliai gali naudotis *IHD* koncepcija, t. y. ieškoti žmonių, kurie turi didesnę įtaką aukai. Pirma įsilaužėliai naudoja profilio klonavimą, tada pradeda susisiekti su auka, kad gautų konfidencialios informacijos, naudodami keletą netiesioginių socialinės inžinerijos technikų, pavyzdžiui, duomenų viliojimas arba pretekstas.

Klonuojant profilį sukuriama netikra paskyra naudojant pavogtą informaciją ir paveikslėlius iš teisėto naudotojo profilio. Užpuolikai naudoja šį kloną bendraudami su aukos kontaktais, dažnai prašydami slaptos informacijos, pinigų arba prieigos prie riboto naudojimo sistemų. Profilio klonavimo aptikimas grindžiamas algoritmais, kurie stebi, ar nėra profilių dublikatų, analizuoja draugų tinklus ir pažymi neįprastą paskyros veiklą. Į prevencijos sistemas gali būti įtraukti realiuoju laiku naudotojams siunčiami įspėjimai, kai aptinkamas įtartinas paskyros dubliavimas.

1.1.10. Atvirkštinė socialinė inžinerija (angl. *Reverse Social Engineering*)

Įsilaužėliai sukuria sąlygas, kad auka pakliūtų į atvirkštinės socialinės inžinerijos spąstus, sabotuoti sistema ir apsimesti vieninteliu galimu technikos specialistu. Tuomet įsilaužėliai tiesiogiai ar netiesiogiai kreipiasi į aukas siūlydami pagalbą [8]. Pavyzdžiui, jie priversdavo sistemą neveikti, kai auka norėdavo prie jos prisijungti, tada siūlydavo pagalbą pataisyti sistemą, kad ji veiktų taip, kaip turėtų, kad įsilaužėliai galėtų laisvai įdiegti bet kokią programą, kurią galima naudoti administracinei prieigai per aukos kompiuterį perimti.

1.1.11. Vandens telkinys (angl. *Water-Holing*)

„*Water-holing*“ reiškia, kad įsilaužėliai nustato svetaines, kuriose dažnai lankosi jų tikslinė auditorija, ir pažeidžia šias svetaines įvesdami kenkėjišką kodą [8]. Kai naudotojai apsilanko užkrėstoje svetainėje, jie netyčia parsisiunčia kenkėjišką programinę įrangą arba pateikia užpuolikui konfidencialią informaciją. Šis metodas ypač veiksmingas prieš organizacijas, turinčias bendrų interneto išteklių. Aptikimas apima tinklo duomenų srauto stebėjimą, ieškant neįprastos veiklos, interneto programų ugniasienių diegimą ir reguliarų lankomų svetainių saugumo auditą. Prevencinės priemonės apima saugios naršymo praktikos užtikrinimą ir atnaujintos programinės įrangos, skirtos pažeidžiamumui sumažinti, palaikymą.

1.2. Socialinės inžinerijos atakų prevencijos metodai

Viena iš naudotojo tapatybės vagystės formų - profilio klonavimas, kai teisėti prisijungimo duomenys naudojami pavogtam profiliui kopijuoti [8]. Siekiant sumažinti profilių klonavimo bandymų skaičių, reikalingas mokomasis modelis. Šis modelis turėtų apimti tokius dalykus, kaip mokymasis apie

profilio nustatymus, draugystės linijų tikrinimas ir vengimas spausti nežinomas nuorodas socialiniuose tinkluose.

Sukčiavimo atakų aptikimo metodas, kuriame daugiausia dėmesio skiriama greitam atsako laikui ir dideliam tikslumui. Šis metodas sujungia nuorodų juodąjį, baltąjį sąrašą ir paieškos sistemų metodus. Į juodąjį sąrašą įtraukti nuorodų adresai, kurie, kaip įtariama, yra sukčiavimo nuorodos, o į baltąjį sąrašą įtraukti nuorodų adresai, kurie laikomi teisėtais nuorodų adresais. Paieškos variklių metodais iš nuorodų gaunama papildoma informacija, pavyzdžiui, domenų pavadinimai, svetainių pavadinimai ir užklausų rezultatai.

Požymių išskyrimo metodas, kurį neuroniniai tinklai gali naudoti socialinės inžinerijos atakoms aptikti. Požymių nustatymas atliekamas telefono skambučiams arba skambintojams, siekiant nustatyti, ar šie požymiai apima socialinės inžinerijos atakas, ar ne [8]. Nustatyti šie požymiai: balso identifikavimas, konfidencialios informacijos prašymas, slaptažodžių prašymas, programų diegimas. Šis požymis gaunamas remiantis pokalbių, kurie buvo identifikuoti kaip socialinės inžinerijos atakos, duomenų rinkiniu [9] . Gauti požymiai yra minimalūs ir nepatikrinti realiomis sąlygomis. Todėl, norint plėtoti šį tyrimą, būtina nustatyti papildomus požymius, pavyzdžiui, pokalbio laiką, pokalbių dažnumą ir trukmę.

Socialinės inžinerijos išpuolius galima nustatyti taikant dirbtinių neuronų tinklo (ANN) ir natūralios kalbos apdorojimo (angl. *NLP*) metodus [8]. ANN naudojamas NLP proceso rezultatams kategorizuoti, t. y. ar tai yra socialinės inžinerijos ataka, o NLP naudojamas pokalbio turiniui iššifruoti ir gramatinėms klaidoms patikrinti [10]. Didelis tikslumas pasiekiamas derinant NLP ir ANN metodus, tačiau šis derinys nebuvo įvertintas subalansuotame duomenų rinkinyje su daugiau atributų ir realių situacijų.

1.2.1. El. Pašto filtravimas

El. pašto filtrai, kuriuose naudojamas mašininis mokymasis, gali atpažinti ir blokuoti apgaulingus el. laiškus, kol jie dar nepasiekė naudotojų. Šie filtrai analizuoja el. laiško turinį, siuntėjo reputaciją ir priedų elgseną, kad aptiktų įtartinus el. laiškus ir patalpintų juos į karantiną

1.2.2. Aplikacijų sąsajų apsauga (angl. *endpoint protection*)

Šios programos gali aptikti kenkėjišką programinę įrangą, kuri gali būti siunčiama naudojant socialinės inžinerijos taktiką, pvz., kenkėjiškus priedus ar nuorodas apgaulinguose el. laiškuose, ir užkirsti jai kelią.

1.2.3. Kelių faktorių autentifikacija (angl. *multi-factor authentication*)

Kelių faktorių autentifikacija reikalauja kelių patikros formų, pavyzdžiui, slaptažodžio ir biometrinio veiksnio arba vienkartinio kodo, siunčiamo į mobilųjį įrenginį, įsilaužėliams gerokai sunkiau gauti prieigą, net jei gavo naudotojo prisijungimo duomenis socialinės inžinerijos būdu.

1.2.4. Internetinių svetainių filtravimo sprendimai

Šiomis priemonėmis galima blokuoti prieigą prie žinomų kenkėjiškų svetainių, kurios gali būti naudojamos sukčiavimo atakoms. Jos analizuoja URL adresus realiuoju laiku, kad užtikrintų, jog naudotojai nesilankytų pažeistose ar apgaulingose svetainėse.

1.2.5. Siuntėjo patikrinimo sistema (angl. *SPF*)

Siuntėjo politikos sistema (SPF) - tai el. pašto autentiškumo patvirtinimo protokolas, skirtas užkirsti kelią el. pašto klastojimui, suteikiant domenų savininkams galimybę nurodyti, kurie pašto serveriai yra įgalioti siųsti el. laiškus jų vardu. Naudojant DNS TXT įrašus, SPF leidžia gaunančiajam pašto serveriui patikrinti siuntėjo IP adresą pagal domeno savininko paskelbtą autorizuotų adresų sąrašą [12]. Šis tikrinimo procesas padidina el. pašto saugumą, nes sumažina sėkmingų sukčiavimo atakų, kurių metu dažnai naudojami suklastoti el. pašto adresai, siekiant apgauti gavėjus, tikimybę. Kai gaunamas el. laiškas, pašto serveris atlieka SPF patikrinimą ir, atsižvelgdamas į siuntėjo IP ir autorizuotų adresų sąrašo atitikimą, priskiria būseną. SPF veiksmingumas dar labiau padidėja, kai jis derinamas su kitais protokolais, tokiais kaip DKIM (*DomainKeys Identified Mail*) ir DMARC (*Domain-based Message Authentication, Reporting, and Conformance*), taip sukuriant patikimą el. pašto vientisumo ir autentiškumo apsaugos sistemą vis priešiškesnėje skaitmeninėje aplinkoje.

1.2.6. Švietimas

Veiksmingiausia strategija, kaip išvengti socialinės inžinerijos atakų yra būsimų aukų švietimas ir mokymas [8]. Todėl aukos turi turėti pakankamai žinių, kad apsisaugotų nuo socialinės inžinerijos bandymų.

1.2.7. Socialinės inžinerijos politika

Sukurta keturių lygių valdymo politiką, kad būtų užkirstas kelias socialinei inžinerijai [8]. Rizikos valdymas naudojamas pirmuoju lygmeniu siekiant užkirsti kelią socialinės inžinerijos išpuoliams. Antrajame lygyje paaiškinamos pareigos ir jų vykdymo metodai. Trečiajam lygiui yra parengtas socialinės inžinerijos atakų prevencijai skirtų vertinimų rinkinys ir gairės, kaip konkrečiai deklaruojama, kad veikla baigta. Ketvirtame lygyje galima rasti sistemą, kurią galima įtraukti į sistemą, kad būtų galima sustabdyti socialinės inžinerijos atakas ir palaikyti skaitmeninį tokių atakų įrodymą.

Organizacijos, įgyvendinančios kibernetinio saugumo taisykles, ypač apsaugančias nuo socialinės inžinerijos atakų, susiduria su sunkumais dėl žmogiškųjų klaidų įgyvendinant politiką. Viena iš priežasčių yra ta, kad parengtose taisyklėse trūksta aiškių nurodymų, kaip sustabdyti socialinės inžinerijos atakas ar į jas reaguoti. Tai logiška, nes norint užkirsti kelią šioms socialinės inžinerijos atakoms reikia daugiau žmonių sąveikos švietimo, o ne technologijų naudojimo forma. Organizacijai prireiks laiko ir pinigų, nes ji turės stebėti kiekvieno nario judėjimą.

1.3. Socialinės inžinerijos atakų atpažinimas naudojant DI

1.3.1. Elgsenos analizavimas ir anomalijų aptikimas

Elgsenos analizė apima nuolatinį naudotojų ir sistemų veiklos stebėjimą, siekiant nustatyti įprasto elgesio modelių atskaitos tašką. Nukrypimai nuo šių nustatytų modelių, vadinamieji anomalijos, gali rodyti galimus saugumo pažeidimus arba kenkėjišką veiklą. Anomalijų aptikimui naudojami sudėtingi algoritmai, dažnai grindžiami mašininiu mokymusi, kad būtų galima pastebėti neįprastą elgesį, kuris gali būti neaptinkamas taikant tradicinius parašais pagrįstus metodus. Pavyzdžiui, Naudotojo ir subjekto elgsenos analizės sistemoje naudojamas mašininis mokymasis, kad būtų galima analizuoti įvairias elgsenos metrikas, pavyzdžiui, prisijungimo laiką, prieigos modelius ir duomenų naudojimą, siekiant nustatyti anomalijas, rodančias grėsmes iš vidaus arba įtartinas paskyras.

Integravus dirbtinį intelektą į anomalijų aptikimą, dar labiau padidėja jo veiksmingumas, nes galima atlikti analizę realiuoju laiku ir sumažinti klaidingų teigiamų rezultatų skaičių, todėl saugumo komandos gali sutelkti dėmesį į tikrąsias grėsmes. Apskritai elgesio analizės ir anomalijų aptikimo sistemų diegimas gerokai sustiprina organizacijos saugumo padėtį, nes aktyviai nustato sudėtingas kibernetines grėsmes ir į jas reaguoja.

1.3.2. Duomenų viliojimo atpažinimas

Atliekant dirbtinio intelekto analizę sukčiavimo aptikimo srityje naudojami mašininio mokymosi algoritmai, kurie mokosi iš paženklintų arba nepaženklintų el. laiškų duomenų, kad atskirtų sukčiavimo ir teisėtus el. laiškus. Prižiūrimi mokymosi algoritmai yra apmokyti iš pažymėtų duomenų rinkinių, kad galėtų klasifikuoti el. laiškus pagal tokius požymius kaip teksto modeliai ir siuntėjo informacija. Nekontroliuojamo mokymosi algoritmai aptinka el. pašto duomenų anomalijas, kurios gali rodyti bandymus sukčiauti. Be to, dirbtinis intelektas naudoja natūralios kalbos apdorojimą, kad iš el. pašto turinio išgautų semantinę prasmę ir nustatytų apgaulingą kalbą. Anomalijų aptikimo algoritmai pažymi neįprastą naudotojo elgseną, taip padidindami bendrą aptikimo veiksmingumą. Nuolat mokantis ir prisitaikant, dirbtinio intelekto valdomos sistemos atlieka svarbų vaidmenį stiprinant kibernetinio saugumo apsaugą nuo kintančių sukčiavimo grėsmių.

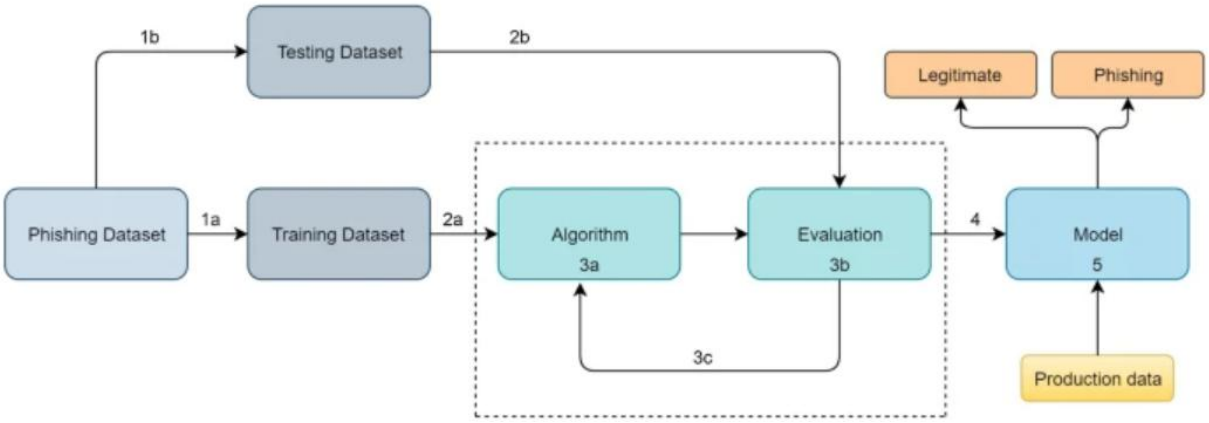
1.3.3. Elektroninio pašto žinučių analizavimas naudojant mašininį mokymą

Mašininis mokymasis (angl. *Machine learning, ML*) atlieka svarbų vaidmenį nustatant ir užkertant kelią verslo elektroninio pašto (angl. *BEC*) atakoms, nes analizuoja istorinius elektroninio pašto duomenis ir nustato su žinomomis grėsmėmis susijusius dėsningumus. Mašininio mokymo modeliai, apmokyti pagal tokius požymius, kaip įtartini el. pašto adresai, kalbos modeliai ir el. pašto antraščių anomalijos, gali aptikti „*BEC“* atakų požymius. Realiuoju laiku atliekama el. laiško turinio, struktūros, siuntėjo reputacijos ir kontekstinės informacijos analizė dar labiau pagerina aptikimą, nustatant įtartiną veiklą, pavyzdžiui, netikėtus sąskaitos pakeitimus ar skubius lėšų prašymus [12].

Mašininis mokymas taip pat padeda nustatyti pažeistas paskyras, nes stebi naudotojų elgseną ir pažymi nukrypimus, pavyzdžiui, nepažįstamas prisijungimo vietas arba neįprastą veiklos laiką. Pavyzdys - „*Barracuda Networks*“ aptikimo sistema „*BEC-Guard*“ [12], kuri grėsmėms nustatyti naudoja prižiūrimą mokymąsi didelėje el. pašto duomenų bazėje. Ji integruoja debesijos el. pašto paslaugų teikėjų *API*, kad išmoktų organizacijos bendravimo modelius, ir pasiekia aukštą 98,2 % tikslumą, o klaidingai teigiamų atvejų skaičius neviršija vieno iš penkių milijonų el. laiškų.

1.4. Mašininio mokymo technikos

Norint apmokyti mokymusi pagrįstos aptikimo sistemos mašininio mokymosi modelį, duomenyse turi būti požymių, susijusių su sukčiavimo ir teisėtų tekstų klasėmis. Apgaulingoms atakoms aptikti naudojami įvairūs klasifikatoriai, aptikimo tikslumas yra didelis, kai naudojami patikimi mašininio mokymo (angl. *ML*) metodai [15]. Siekiant sumažinti požymių skaičių, naudojami keli požymių atrankos metodai. **1 pav.** [15] parodyta, kaip veikia mašininio mokymosi modelis. Įvesties duomenų rinkinys pateikiamas kaip įvestis mašininio mokymosi modelio mokymui, kad būtų galima prognozuoti sukčiavimo ataką arba teisėtą srautą.



1 pav. Mašininio modelio veikimas

1.4.1. Sprendimų medis (angl. *Decision Tree)*

Sprendimų medžio algoritme naudojamas duomenų klasifikavimo procesas grindžiamas taisyklių seka. Gautas modelis turi medžio pavidalo struktūrą. Neatsiejama sprendimų medžio algoritmo savybė yra automatinis nereikšmingų ir nereikalingų požymių pašalinimas. Mokymosi procesą sudaro atskiri etapai: požymių atranka, medžio generavimas ir medžio genėjimas [2]. Per sprendimų medžio modelio mokymo etapą algoritmas sistemingai atskirai atrenka tinkamiausius požymius ir generuoja antrinius mazgus, pradedant nuo šakninio mazgo. Pabrėžiama, kad sprendimų medis pats savaime tarnauja kaip pagrindinis klasifikatorius. Pažangūs algoritmai, pavyzdžiui, atsitiktinių sprendimų miškas ir ekstremalaus gradiento didinimo algoritmas (angl. *XGBoost*), peržengia šio paprastumo ribas ir į savo sistemas įtraukia kelis sprendimų medžius [2].

1.4.2. Atsitiktinio miško klasifikatorius (angl. *Random Forest)*

Atsitiktinio miško klasifikatorius [1] yra sudėtinis modelis, sudarytas iš daugelio sprendimų medžių, kurie veikia kartu. Kiekvienas sprendimų medis tarnauja kaip sprendimų priėmimo įrankis, priimantis į medį panašią struktūrą sprendimams priimti ir rezultatams prognozuoti. Kiekvienas atsitiktinio miško modelio medis tam tikram duomenų rinkinio įrašui prognozuoja konkrečią klasę, o daugiausiai kolektyvinių balsų surinkusi klasė pasirenkama kaip viso modelio išvestis. ,,Norint užtikrinti optimalų veikimą, algoritmą reikia mokyti naudojant gerai reprezentatyvius duomenis, be to, labai svarbu išlaikyti mažą koreliaciją tarp kiekvieno medžio prognozių.

1.4.3. C4.5 metodas (angl. *C4.5)*

Ross Quinlan [22] sukurtas C4.5 algoritmas yra ID3 algoritmo plėtinys ir yra plačiai naudojamas sprendimų medžiams konstruoti mašininio mokymosi srityje. Tai prižiūrimo mokymosi algoritmas, kuris sudaro sprendimų medį iš mokymo duomenų aibės, naudodamasis informacijos entropijos koncepcija [23]. Algoritmas rekursyviai parenka požymį, kurio informacijos prieaugio santykis yra didžiausias, kaip skaidymo kriterijų kiekviename medžio mazge.

Šiuo metodu siekiama minimizuoti entropiją ir maksimizuoti gautų poaibių homogeniškumą. C4.5 įdiegta keletas patobulinimų, palyginti su ID3, pavyzdžiui, darbas su ištisiniais atributais, trūkstamomis reikšmėmis ir sprendimų medžio apkarpymas siekiant išvengti per didelio pritaikymo. Algoritme taikoma „skaldyk ir valdyk“ strategija, kai mokymo duomenys palaipsniui skaidomi į

mažesnius poaibius pagal pasirinktus požymius, kol įvykdoma baigiamoji sąlyga, pavyzdžiui, visi poaibio atvejai priklauso tai pačiai klasei arba pasiekiamas minimalus atvejų skaičius. C4.5 sėkmingai taikomas įvairiose srityse, įskaitant medicininę diagnostiką, kredito rizikos vertinimą ir klientų skaičiaus mažėjimo prognozavimą, nes sugeba generuoti interpretuojamas ir žmogui suprantamas sprendimų priėmimo taisykles [24].

Tačiau šis algoritmas gali susidurti su sunkumais dirbant su didelės apimties duomenų rinkiniais ir gali būti jautrus triukšmingiems ar nereikšmingiems požymiams.

1.4.4. K artimiausių kaimynų algoritmas (angl. *K-Nearest Neighbors, KNN*)

K artimiausių kaimynų algoritmas (angl. *k-nearest neighbors, KNN*) yra paprastas, tačiau galingas prižiūrimas mašininio mokymosi metodas, naudojamas klasifikavimo ir regresijos užduotims atlikti [21]. KNN veikia pagal principą, kad tam tikroje požymių erdvėje panašūs duomenų taškai yra arti vienas kito. Jis klasifikuoja naują duomenų tašką nustatydamas K artimiausių kaimynų iš mokymo duomenų aibės ir priskirdamas bendriausią etiketę (klasifikavimo atveju) arba vidutinę jų vertę (regresijos atveju). Atstumas tarp duomenų taškų paprastai matuojamas naudojant tokias metrikas, kaip Euklido atstumas, Manheteno atstumas arba kosinusinis panašumas, priklausomai nuo duomenų rinkinio pobūdžio [21].

Vienas iš pagrindinių KNN privalumų yra neparametrinis pobūdis, t. y. jis nedaro prielaidų apie duomenų pasiskirstymą. Dėl to ji yra veiksminga įvairiose srityse, įskaitant modelių atpažinimą, rekomendavimo sistemas ir anomalijų aptikimą. Tačiau KNN turi nemažai trūkumų, įskaitant dideles skaičiavimo sąnaudas prognozavimo metu, nes norint nustatyti artimiausius kaimynus reikia ieškoti visame duomenų rinkinyje [21]. Be to, KNN jautriai reaguoja į nereikšmingus požymius ir duomenų mastelio keitimą, todėl dažnai prireikia išankstinio apdorojimo metodų, pavyzdžiui, požymių atrankos ir normalizavimo. Nepaisant šių trūkumų, KNN išlieka plačiai naudojamas algoritmas dėl savo paprastumo, aiškumo ir efektyvumo mažų matmenų duomenų rinkiniuose su aiškiomis klasterių struktūromis.

1.5. Mašininio mokymo algoritmų naudojimo nauda

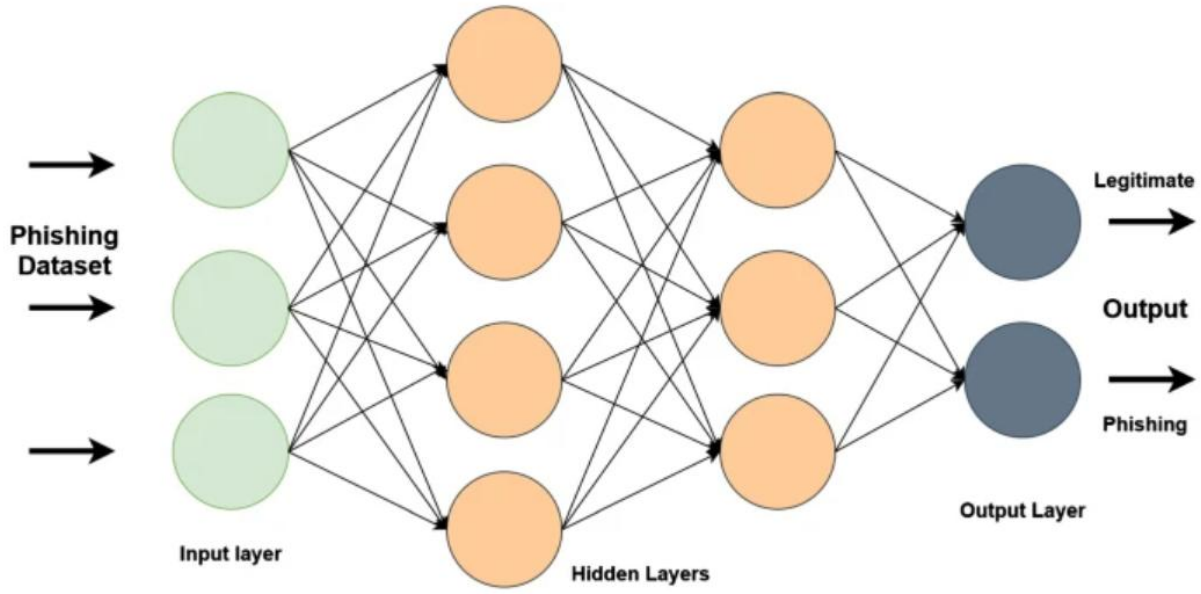
Nustatyta, kad dirbtinio intelekto sistemoms teikiama informacija gerokai pagerina jų gebėjimą atremti atakas. Kibernetinį saugumą padeda užtikrinti mašininio mokymosi technologijų pažanga. Tai tapo įmanoma, kai sistema mokosi realiuoju laiku ir kartu apsaugo naudotoją [11]. Kibernetinio saugumo srityje dirbtinis intelektas siūlo automatines funkcijas, kurios greitai reaguoja, užtikrindamos saugumą. Dėl to sistemos geba nustatyti pagrindinius pavojus ir kurti strategijas, kaip užkirsti kelią šioms atakoms. Lyginant su žmonėmis, reagavimo laikas skatina tokį saugumą.

Šiuolaikiniame kibernetiniame saugume dirbtinis intelektas turi didelių privalumų [11]. Visų pirma, lyginant su tradiciniais metodais, dirbtiniu intelektu valdomos sistemos reaguoja gerokai greičiau, todėl grėsmės gali būti greičiau pašalintos. Antra, dirbtinis intelektas gali nuolat mokytis ir tobulėti. Pasitelkdamos mašininį mokymąsi, AI sistemos gali analizuoti ankstesnes atakas, nustatyti dėsningumus ir atitinkamai pritaikyti savo gynybą, taip ilgainiui didindamos jos veiksmingumą. Be to, dirbtinis intelektas puikiai aptinka naujas ir sudėtingas atakas, kurių gali nepastebėti žmonių analitikai. Dėl dinamiško grėsmių aplinkos pobūdžio būtina aktyvi gynyba, o dirbtinio intelekto gebėjimas nustatyti kylančias grėsmes yra labai svarbus [11]. Galiausiai, dirbtinis intelektas gali

veiksmingai apdoroti ir analizuoti didžiulius duomenų rinkinius, o tai yra labai svarbus gebėjimas visapusiškai saugumo stebėsenai ir pažeidžiamumų valdymui.

1.6. Giliojo mokymo technikos (angl. *Deep-learning*)

Remiantis naujausiais giliojo mokymo (angl. *DL*) metodų pasiekimais, giliųjų neuronų tinklai (angl. *NN*) turėtų veikti geriau nei įprasti mašininio mokymosi (angl. *ML*) algoritmai, kai kalbama apie sukčiavimo svetainių klasifikavimą. Tačiau įvairių mokymosi parametrų konfigūracija turi didelę įtaką giliųjų neuroninių tinklų naudojimo rezultatams [15]. **2 pav.** parodyta, kaip veikia giliojo mokymosi modeliai. Norint prognozuoti bandymą sukčiauti arba legalų srautą, į neuronus paduodama įvesties duomenų rinkinys, kuriam suteikiami svoriai [15].



2 pav. Giliojo mokymo veikimas

1.6.1. Išankstiniai neuroniniai tinklai (angl. Feed-forward deep neural-network)

Tai pagrindinė gilaus mokymosi architektūra, kuriai būdingas vienakryptis informacijos srautas iš įvesties į išvesties sluoksnius be jokių grįžtamojo ryšio jungčių. Šiuos tinklus sudaro keli tarpusavyje sujungtų dirbtinių neuronų sluoksniai: įvesties sluoksnis, į kurį patenka neapdoroti duomenys, vienas ar daugiau paslėptųjų sluoksnių, kurie apdoroja informaciją naudodami svertines jungtis ir aktyvavimo funkcijas, ir išvesties sluoksnis, kuriame pateikiamos galutinės prognozės. Kiekvienas neuronas apskaičiuoja svertinę savo įėjimų sumą, taiko netiesinę aktyvavimo funkciją ir perduoda rezultatą kitam sluoksniui [16]. Tinklas mokosi per procesą, vadinamą grįžtamuoju skleidimu, kai jis koreguoja savo svorius ir nuokrypius, kad sumažintų prognozuojamų ir faktinių išėjimų skirtumą, naudodamas tokius optimizavimo algoritmus kaip stochastinis gradientinis nusileidimas (angl. *Stochastic Gadirent Descent - SGD*). FNN puikiai atlieka modelių atpažinimo ir funkcijų aproksimacijos užduotis, todėl tinka įvairioms taikomosioms programoms, įskaitant vaizdų klasifikavimą, regresijos problemas ir požymių mokymąsi. Tačiau jų veiksmingumas priklauso nuo tinkamo architektūros dizaino, kruopštaus hiperparametrų derinimo ir pakankamo mokymo duomenų kiekio, kad būtų išvengta tokių problemų kaip perteklinis pritaikymas ir vietinių minimumų konvergencija.

1.6.2. Pasikartojantys neuroniniai tinklai (angl. *Recurrent neural-network, RNN*)

Pasikartojantys neuroniniai tinklai (angl. *Recurrent neural network - RNN*) - tai dirbtinių neuroninių tinklų klasė, kuri sulaukė didelio dėmesio gilaus mokymosi srityje dėl savo gebėjimo efektyviai apdoroti nuoseklius duomenis [17]. *RNN* apima grįžtamojo ryšio kilpą, kuri leidžia jiems išlaikyti paslėptą būseną laikui bėgant, todėl jie gali fiksuoti ir mokytis iš laiko priklausomybių, esančių įvesties sekose. Dėl tokios architektūros *RNN* ypač gerai tinka užduotims, susijusioms su laiko eilutėmis, pavyzdžiui, kalbos modeliavimui, kalbos atpažinimui ir mašininiam vertimui [16]. *RNN* pasiekė geriausius rezultatus sprendžiant įvairias seka į seką užduotis ir tapo neatsiejama daugelio natūralios kalbos apdorojimo ir kalbos atpažinimo sistemų dalimi. Tačiau, nepaisant jų sėkmės, *RNN* vis dar susiduria su tokiomis problemomis kaip skaičiavimo sudėtingumas, sunkumai fiksuojant labai ilgalaikes priklausomybes ir didelio mokymo duomenų kiekio poreikis [17]. Vykdomais moksliniais tyrimais siekiama pašalinti šiuos trūkumus ir dar labiau padidinti RNN galimybes apdoroti sudėtingus nuosekliuosius duomenis.

1.6.3. Konvoliuciniai neuroniniai tinklai (angl. *Convolutional neural-network, CNN*)

Konvoliuciniai neuroniniai tinklai (angl. *Convolutional Neural Networks - CNN*) yra specializuotos gilaus mokymosi architektūros, pirmiausia skirtos tinkliniams duomenims apdoroti, ypač puikiai tinkančios vaizdų ir vaizdo įrašų analizei. Skirtingai nuo tradicinių neuroninių tinklų, *CNN* naudoja konvoliucinius sluoksnius, kuriuose naudojami slankiojantys filtrai (branduoliai), kad būtų galima automatiškai aptikti erdvines požymių hierarchijas - nuo paprastų kraštų ir tekstūrų ankstyvuosiuose sluoksniuose iki sudėtingų modelių ir objektų gilesniuose sluoksniuose. Architektūrą paprastai sudaro pakaitomis kintantys konvoliuciniai ir jungiamieji sluoksniai, o galutiniam klasifikavimui - visiškai sujungti sluoksniai. Konvoliuciniai sluoksniai atlieka požymių išskyrimą naudodami išmoktus filtrus, kurie dalijasi svoriais visoje įvesties erdvėje, todėl gerokai sumažėja parametrų skaičius, palyginti su visiškai sujungtais tinklais. Sujungimo sluoksniai (paprastai maksimalus arba vidutinis sujungimas) sumažina požymių žemėlapių imtį, todėl užtikrinamas vertimo invariantiškumas ir sumažinamas skaičiavimo sudėtingumas [17].

1.6.4. FNN, CNN, RNN palyginimas

RNN gerai tinka nuosekliems duomenims apdoroti, todėl yra geras pasirinkimas analizuojant el. pašto turinį ir nuorodų adresus sukčiavimo aptikimo tikslais. Jie gali užfiksuoti teksto kontekstą ir priklausomybes, todėl gali nustatyti įtartinus modelius ir kalbą, dažniausiai naudojamą sukčiavimo el. laiškuose. Tačiau *RNN* gali būti sunku užfiksuoti ilgalaikes priklausomybes, o ilgoms sekoms gali būti reikalingos didelės skaičiavimo sąnaudos.

FFN, kurie yra paprasčiausi iš visų trijų architektūrų, gali būti naudojami sukčiavimui aptikti, išgaunant svarbius požymius iš el. laiško turinio, antraščių ir metaduomenų. Remdamiesi šiomis savybėmis jie gali išmokti klasifikuoti el. laiškus kaip apgaulingus ar teisėtus. Tačiau *FFN* aiškiai neatsižvelgia į nuoseklų teksto pobūdį ir gali būti ne tokios veiksmingos fiksuojant kontekstinę informaciją, palyginti su *RNN*.

CNN, iš pradžių sukurtus vaizdams apdoroti, taip pat galima taikyti apgaulingam apsimetinėjimui aptikti, el. laiško turinį arba nuorodos adresą traktuojant kaip simbolių seką. *CNN* gali išmokti identifikuoti vietinius modelius ir išgauti svarbius požymius naudodami konvoliucinius filtrus. Įrodyta, kad jie veiksmingai fiksuoja erdvines priklausomybes ir identifikuoja diskriminacinius

modelius tekste. *CNN* gali apdoroti kintamo ilgio įvesties duomenis ir, palyginti su *RNN*, yra skaičiavimo požiūriu efektyvesni.

Apibendrinant galima teigti, kad sukčiavimo aptikimui galima naudoti visas tris architektūras, kurių kiekviena turi savų privalumų ir trūkumų. *RNN* tinka nuosekliems duomenims apdoroti ir kontekstui fiksuoti, *FFN* yra paprastos ir gali mokytis iš išskirtų požymių, o *CNN* puikiai atpažįsta vietinius modelius ir veiksmingai apdoroja kintamo ilgio įvestis. Architektūros pasirinkimas priklauso nuo konkrečių reikalavimų, turimų duomenų ir skaičiavimo išteklių. Taip pat galima apsvarstyti galimybę taikyti hibridinį metodą, derinantį kelias architektūras, kad būtų išnaudotos jų papildomos stiprybės ir pagerintas sukčiavimo aptikimo efektyvumas.

1.7. Giliojo mokymo palyginimas su mašininiu mokymu

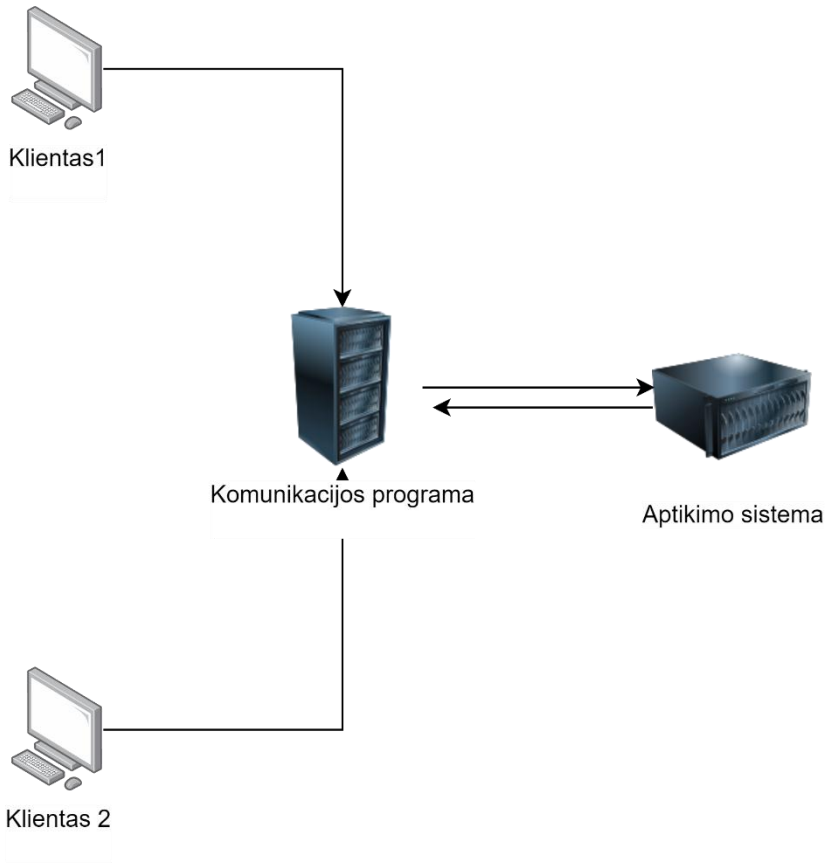
Gilusis mokymas yra naudojamas, kai turimi dideli duomenų kiekiai, duomenų kiekiai nėra struktūrizuoti, reikia aukščiausio galimo tikslumo, turima prieiga prie milžiniškų kompiuterinės technikos ir apskaičiavimo mašinų. Priešingai, mašininiam mokymui galima naudoti mažesnius duomenų kiekius, duomenys yra struktūrizuoti, tikslumas gali būti šiek tik mažesnis turimi limituoti kompiuteriniai resursai.

1.8. Analizės išvados

1. Yra daug skirtingų socialinės inžinerijos atakų tipų, aukos yra apgaunamos naudojantis el. laiškais, SMS žinutėmis, apsimetinėjant tam tikrais asmenimis norint gauti jautrią informaciją;
2. Norint apsaugoti žmones nuo galimų atakų yra keli sprendimo būdai, įvedant saugos politiką, vedant mokymus arba bandant apsaugoti sistemą technologiškai, naudojant filtravimo taisykles, mašininio mokymo algoritmus, trečių šalių servisus;
3. Mašininio mokymo algoritmas „*Random Forest*“ tinkamas dėl jo pobūdžio, kuris sujungia kelis sprendimų medžius, kad sumažintų perteklinį pritaikymą ir pagerintų suvienodinimą. Jis gerai apdoroja didelės apimties duomenis, yra atsparus triukšmui ir pateikia požymių svarbos metrikas, todėl tinka sudėtingiems modeliams nustatyti duomenų viliojimo atakas.

2. Sprendimo koncepcija

2.1. Sistemos diegimo infrastruktūra



3 pav. Duomenų viliojimo atakos atpažinimo sistemos infrastruktūra

Demonstruojama norimos įgyvendinti sistemos infrastruktūra. Pats anomalijų aptikimo modelis apdoroja gaunamus žinutes ir paruošia jų analizei. Nuorodų analizės modelis analizuoja ar nuorodos yra kenksmingos. Pridėtų failų modulis analizuoja pridėtus failus, Apmokytas dirbtinio intelekto modelis atlieka siunčiamą žinutę ir nusprendžia ar žinutė kenksminga. Jei aptinkamos anomalijos, sistema informuoja apie gaunamą galimą kenksmingą duomenų srautą.

2.2. Sistemos architektūra

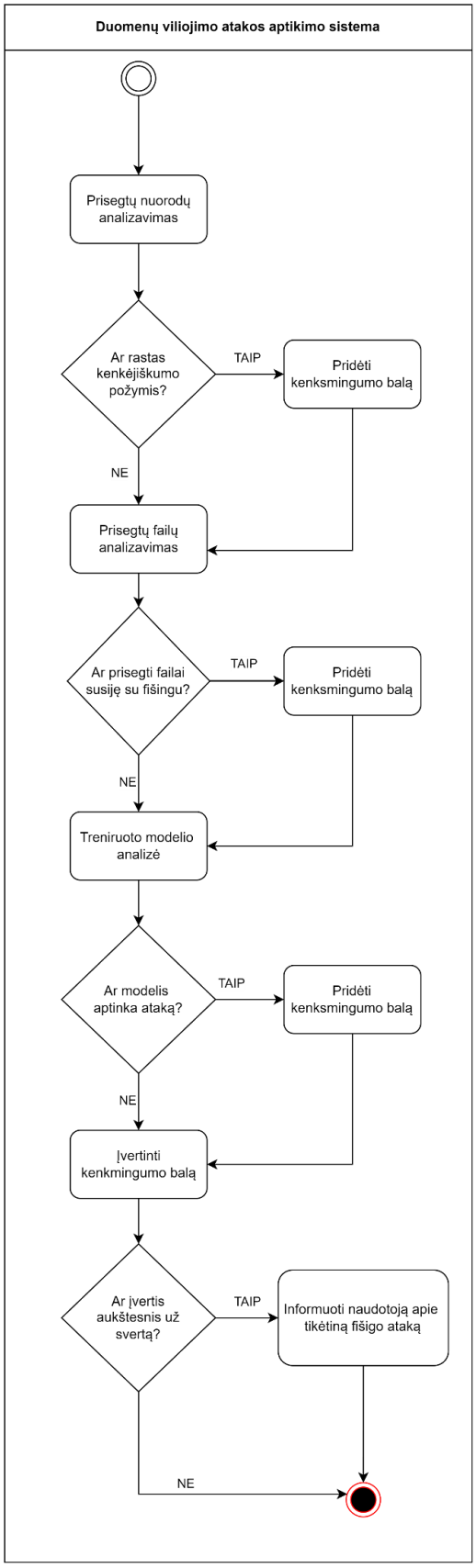
Duomenų viliojimo atakos aptikimo sistema atsakinga už realiuoju laiku atliekamas atakas. Šis modulis analizuoja sukonfigūruotas komunikacijos programinės sąsajos pranešimus tarp komunikacijos programos naudotojų.

2.3. Duomenų šaltiniai

Duomenų rinkimas iš žinučių siuntimo paslaugų, pavyzdžiui, „*Discord*“, atlieka labai svarbų vaidmenį renkant svarbius duomenis apie pranešimus. Žinučių perdavimo platformos dažnai tampa užpuolikų taikiniu dėl jų paplitimo ir vartotojų pasitikėjimo šiais kanalais gaunamais pranešimais. Veiksmingas duomenų rinkimas apima prieigą prie žinučių, siuntėjo informacijos, priedų ir įterptų nuorodų. Tokiose platformose kaip „*Discord*“ pateikiamos *API*, leidžiančios programuotojams gauti struktūrizuotus duomenis, įskaitant naudotojo identifikatorių, laiko žymas, pranešimų turinį ir

metaduomenis, kuriuos galima saugiai saugoti analizei. Surinkti duomenys yra pagrindas aptikti kenkėjišką veiklą, pavyzdžiui, nustatyti sukčiavimo raktinius žodžius, patvirtinti siuntėjo autentiškumą ir analizuoti įtartinus priedus ar nuorodų adresus.

2.4. Duomenų viliojimo atakų aptikimas



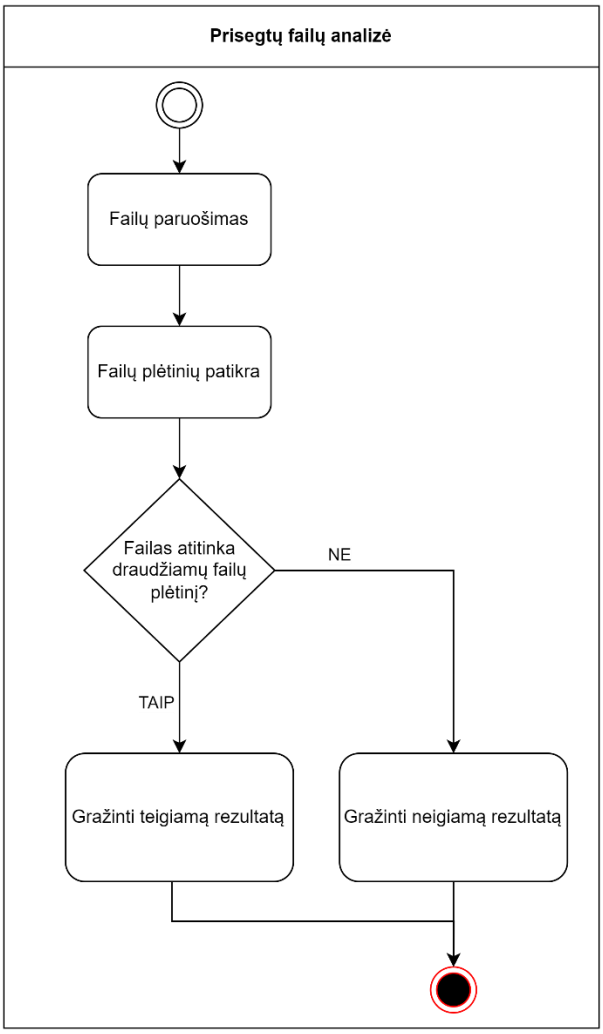
4 pav. Duomenų viliojimo atakos aptikimo sistema

UML diagramoje (**4 pav.**) procesas prasideda nuo nuorodų. Metodas tikrina ar nuorodos yra kenksmingos, kenksmingumas apskaičiuojamas pagal tam tikras taisykles. Kenksmingumo balas sumuojamas. Jei žinutė neturi nuorodos, einama prie failų analizės, jei nėra failų, tikrinamas žinutės tekstas. Taip pat sukuriamas taisyklių sąrašas failų analizei, kiekviena taisyklė turi savo įverčio balą. Jei randamas bent vienos taisyklės atitikmuo, balas pridedamas ir toliau žinutė persiunčiama modelio analizei. Treniruotas modelis įvertina tekstą ir nustato kenksmingumo tikimybę. Galiausiai metodas patikrina visus įverčius pagal nustatytą svertą, jei svertas žemesnis už įvertį, žinutė pažymima kaip ataka, jei ne, žinutė nepažymima.

2.4.1. Prisegtų failų analizė

Prisegtų failų analizė yra labai svarbi sukčiavimo aptikimo sistemų sudedamoji dalis, nes daugiausia dėmesio skiriama potencialiai kenkėjiškų failų, pridedamų prie pranešimų, nustatymui. Analizuojant šiuos failus tikrinami jų metaduomenys, failo tipas, turinys ir elgsena, siekiant aptikti anomalijas ar kompromitavimo požymius. Pirmasis žingsnis - patvirtinti failo tipą, palyginant jo plėtinį su *MIME* tipu įsitikinant, kad jis atitinka deklaruojamą tipą. Reikėtų pažymėti įtartinus failus, pavyzdžiui, vykdomuosius failus (.exe), scenarijus (angl. *scripts*) (.js, .vbs) arba dokumentuose įterptas makrokomandas, kad juos būtų galima atidžiau išnagrinėti. Metaduomenys, pavyzdžiui, failo dydis, sukūrimo data ir pakeitimų istorija, taip pat gali padėti nustatyti galimą klastojimą ar padirbinėjimą.

Pažangūs failų turinio analizės metodai apima statinę ir dinaminę analizę. Statinė analizė analizuoja failą jo nevykdant, naudodamasi tokiomis priemonėmis kaip *YARA* taisyklės, ieškodama žinomų kenkėjiškų šablonų ar parašų. Naudojant šifravimo algoritmus, tokius kaip *SHA-256,* galima sukurti unikalų failo pirštų atspaudą, kurį vėliau galima palyginti su grėsmių žvalgybos duomenų bazėmis, tokiomis kaip „*VirusTotal*“, ir nustatyti žinomą kenkėjišką programinę įrangą.



5 pav. Prisegtų failų analizė

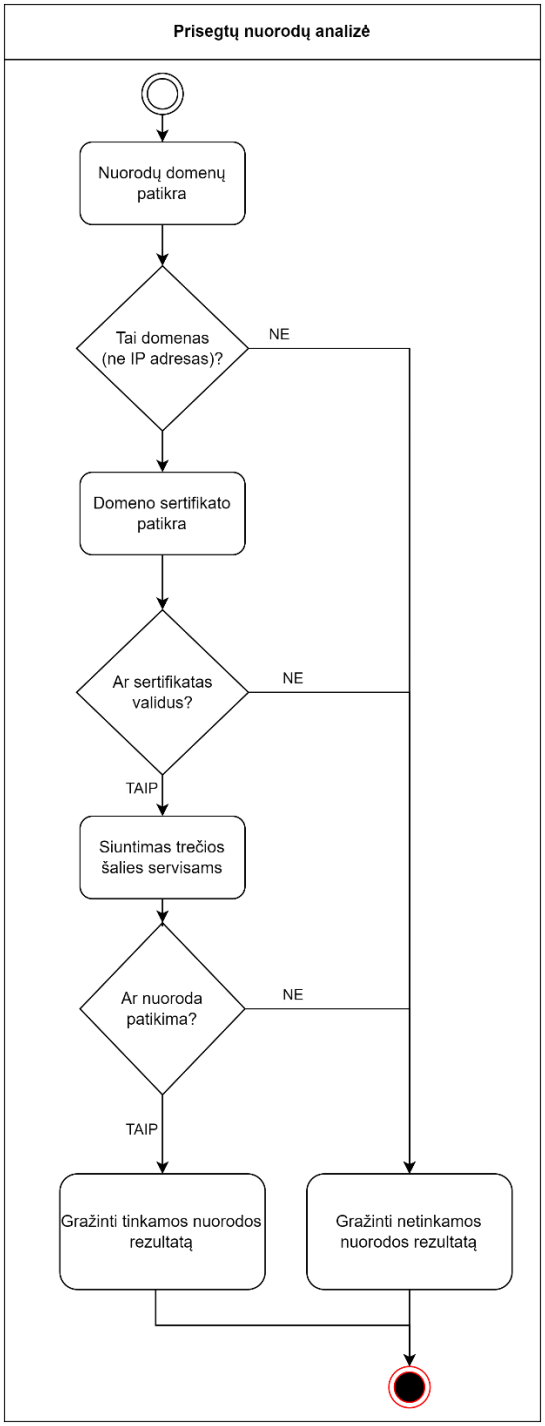
Būtina atidžiai tikrinti pridėtus failus, ypač atkreipiant dėmesį į failų plėtinius. Vienas iš pagrindinių aspektų yra kruopštus failų plėtinių tikrinimas, nes kenkėjai dažnai maskuoja kenksmingą turinį naudodami apgaulingus plėtinius arba keisdami failų plėtinius, kad jie atrodytų nekenksmingi. Visų pirma tokie failų plėtiniai kaip .exe, .bat, .vbs ir .js paprastai siejami su vykdomaisiais failais, scenarijais ir paketiniais failais, kuriuose gali slypėti kenkėjiška programinė įranga arba kuriuos įvykdžius gali būti inicijuojami kenkėjiški veiksmai. Priešingai, teisėti failai paprastai turi gerai žinomus plėtinius, atitinkančius jiems skirtą formatą, pavyzdžiui, .docx - „*Microsoft Word*“ dokumentai arba .pdf - „*Adobe PDF*“ failai.

2.4.2. Nuorodų analizė

Nuorodų analizė yra labai svarbus metodas nustatant sukčiavimo atakas, nes kenkėjiškos nuorodos dažnai naudojamos siekiant nukreipti naudotojus į apgaulingas svetaines, skirtas slaptai informacijai pavogti. Nuorodų analizė apima jų struktūros, metaduomenų ir susijusių domenų tikrinimą, siekiant nustatyti sukčiavimo požymius. Pradedama nuo nuorodos analizės, siekiant išskirti pagrindinius komponentus, pavyzdžiui, protokolą, domeno pavadinimą, užklausos parametrus ir fragmentus. Įtarimą keliančius požymius, pavyzdžiui, tipus (pvz., „g00gle.com“), per didelį nuorodos ilgį ar neįprastus simbolius, galima pažymėti naudojant šablonų atitikimo algoritmus ir euristiką. Be to,

sutrumpintų nuorodų (pvz., bit.ly) nustatymas išplečiant nuorodą padeda atskleisti tikrąją nuorodos paskirties vietą.

Kitas svarbus nuorodų analizės aspektas - domeno reputacijos patikrinimai. Integruojant grėsmių žvalgybos paslaugas, sistema gali nustatyti nuorodos adresus, susijusius su žinomomis sukčiavimo kampanijomis arba į juodąjį sąrašą įtrauktais domenais. Be to, analizuojant domenų sertifikatus dėl jų galiojimo, išdavėjo patikimumo ir galiojimo pabaigos datų, galima gauti papildomų įžvalgų apie svetainės teisėtumą. Norint atlikti nuodugnesnį patikrinimą, naudojant *DNS* paiešką ir atvirkštinius *IP* patikrinimus galima atskleisti domenų ryšius ir galimai nustatyti ryšius su kenkėjiškomis infrastruktūromis.



6 pav. Prisegtų nuorodų analizė

Šioje diagramoje bandoma atskirti domeno ir *IP* nuorodas, nes pirmoji paprastai reiškia interneto adresą, susijusį su tam tikra organizacija ar subjektu, o antroji - skaitmeninę nuorodą, atitinkančią įrenginio tinklo adresą. Be to, vertinant domeno nuorodų teisėtumą, reikia patikrinti, ar yra skaitmeniniai sertifikatai, kurie kriptografiškai užtikrina svetainės autentiškumą ir saugumą. Be to, naudojant trečiųjų šalių paslaugas, kuriose įdiegti patikimi grėsmių aptikimo mechanizmai, galima padidinti kenkėjiškų nuorodų identifikavimo veiksmingumą, realiuoju laiku analizuojant ir kategorizuojant potencialiai pavojingus nuorodų adresus.

2.4.3. Dirbtinio intelekto analizė

2.4.3.1. Duomenų apdorojimas

Įvesties etape procesas pradedamas gaunant naudotojo pranešimus, po to atliekamas kruopštus pirminis apdorojimas, kad tekstas būtų paruoštas analizei. Tai apima teksto konvertavimą į mažąsias raides, specialiųjų simbolių ir nuorodų adresų pašalinimą bei turinio suskirstymą į prasmingus komponentus. Šie veiksmai užtikrina, kad modelis apdorotų švarius ir normalizuotus duomenis, kuriuose nėra triukšmo ir nereikšmingų elementų.

2.4.3.2. Požymių išskyrimas

Iš anksto apdorotame tekste atliekamas požymių išskyrimas, kai taikomi pažangūs metodai, pavyzdžiui, *TF-IDF* (terminų dažnio ir atvirkštinio dokumento dažnio) vektorizavimas, siekiant tekstinius duomenis paversti skaitiniais atvaizdais. Šiame etape atliekama n-gramų analizė, kurioje fiksuojami nuoseklūs žodžių modeliai ir kontekstas, kurie yra labai svarbūs nustatant į kalbą įterptas sukčiavimo taktikas. Gauta požymių matrica yra mašininio mokymosi pagrindas, apimantis svarbiausius pranešimų lingvistinius požymius. Pranešimų ir veiksmų sistema

2.4.3.3. Klasifikavimas

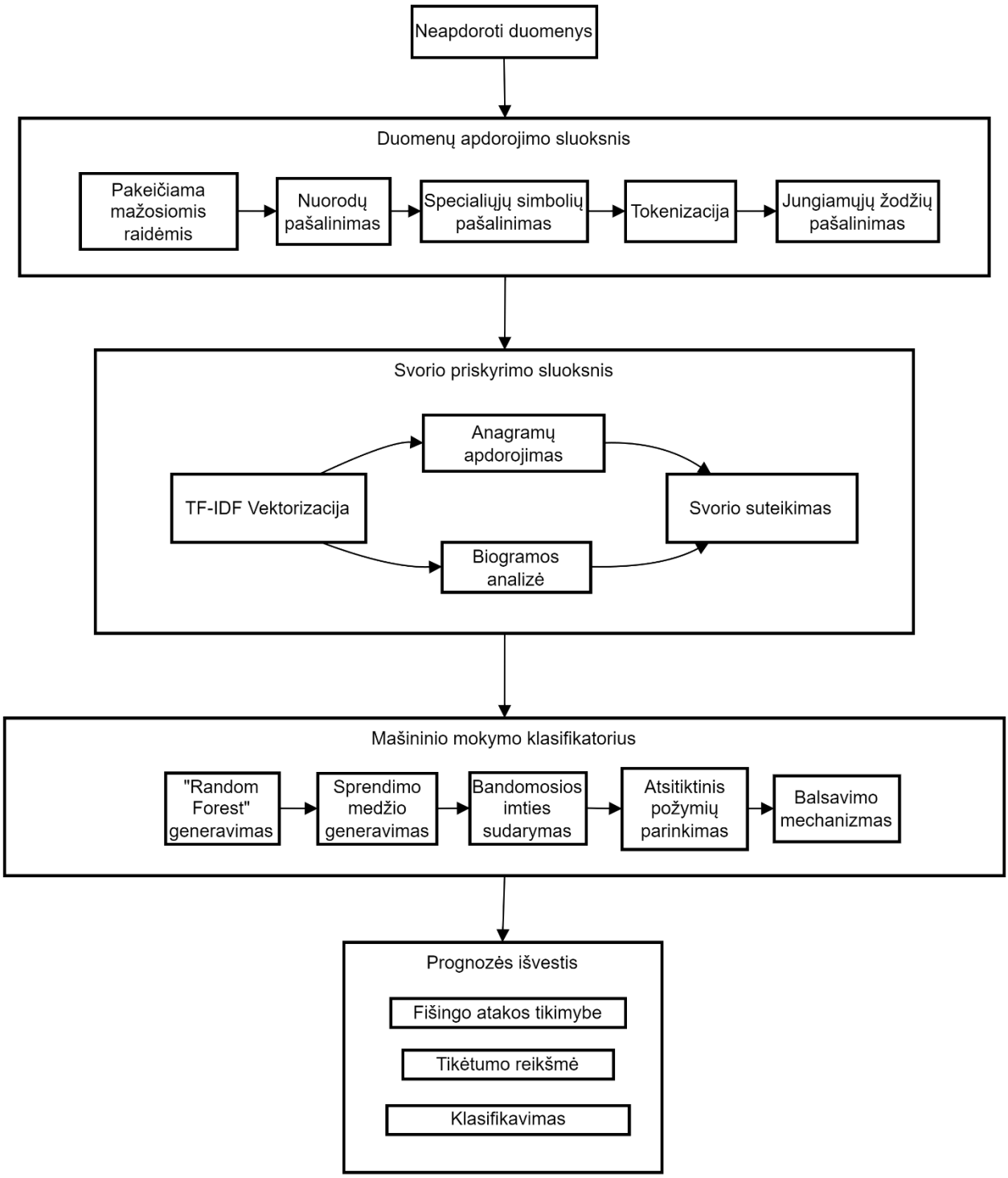
Klasifikavimo etape naudojamas atsitiktinio miško klasifikatorius (angl. *Random Forest*) dėl jo patikimumo ir gebėjimo apdoroti sudėtingus duomenų modelius. Šis ansamblinio mokymosi metodas naudoja kelis sprendimų medžius, remiasi balsų dauguma ir tikimybiniais skaičiavimais, kad užtikrintų tikslias prognozes. Klasifikatorius pateikia dvejetainį sprendimą - sukčiavimas arba teisėtas - kartu su patikimumo balu. Šis tikimybinis metodas ne tik pagerina interpretaciją, bet ir leidžia suprasti prognozės niuansus.

2.4.3.4. Išvestis ir moduliavimas

Galutinėje išvestyje pateikiamas klasifikavimo rezultatas, išsamus patikimumo balas ir pagrindiniai našumo rodikliai, padedantys skaidriai įvertinti modelio veiksmingumą. Galutinė diagrama pabrėžia darbo eigos modulinę struktūrą, parodydama pagrindinius transformacijos etapus nuo pirminio teksto apdorojimo iki prognozavimo. Be to, architektūra palaiko pažangius pirminio apdorojimo metodus, sprendžia klasių disbalanso problemas ir siūlo išplėtimą, kad būtų galima integruoti į įvairias platformas. Šis išsamus dizainas užtikrina modelio pritaikomumą ir mastelio keitimą, todėl jis yra patikimas sprendimas aptikti sukčiavimo atvejus dinamiškoje bendravimo aplinkoje.

2.4.3.5. Veiksmai aptikus

Patvirtinus bandymą sukčiauti, galima automatizuotai atlikti tokius veiksmus kaip siuntėjo blokavimas, pranešimo rodymo uždraudimas arba naudotojo pažymėjimas



7 pav. Mašininio mokymo modelio analizė

2.5. Duomenų viliojimo atakos aptikimo modelio apmokymas

Atsitiktinio miško metodas atrodo ypač tinkamas pasirinkimas dėl jam būdingų privalumų sprendžiant daugialypę užduotį. Jis yra universalus mokymosi metodas, garsėjantis gebėjimu apdoroti

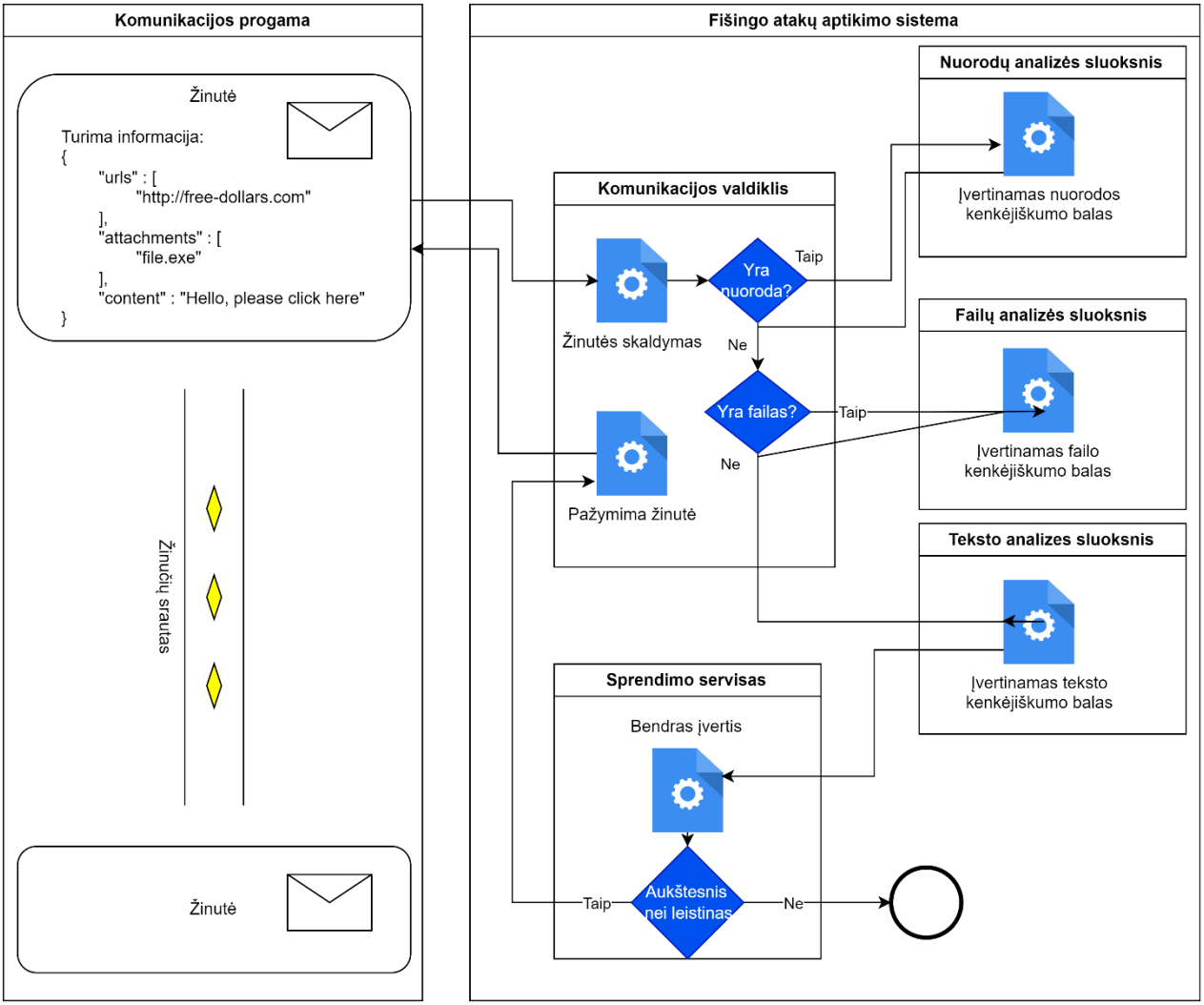
daug dimensijų duomenis, sumažinti perteklinį pritaikymą ir pateikti patikimas prognozes. Nustatant kenkėjiškus pranešimus, „*Random Forest*“ išsiskiria tuo, kad gali sujungti įvairias savybes, gautas kruopščiai tikrinant siuntėjo informaciją, analizuojant turinį, aptinkant kenkėjiškus failus ir nustatant kenkėjiškas nuorodas. Sudarydamas sprendimų medžių modelį, apmokytą pagal duomenų poaibius, „*Random Forest*“ naudoja kolektyvinę šių medžių informaciją, kad pateiktų pagrįstas prognozes, kartu sumažindamas šališkumo ir dispersijos riziką. Be to, dėl jam būdingo lygiagretinimo ir mastelio keitimo jis tinka dideliems duomenų kiekiams, būdingiems kibernetinio saugumo užduotims, apdoroti. Be to, „*Random Forest*“ suteikiamas interpretavimo gebėjimas palengvina pagrindinių požymių, padedančių klasifikuoti kenkėjiškus pranešimus, nustatymą, taip padidinant skaidrumą ir suteikiant kibernetinio saugumo analitikams galimybę gauti naudingų įžvalgų. Taigi „*Random Forest*“ yra išskirtinis algoritmas mašininio mokymosi metodų sąraše, skirtas kenkėjiškiems pranešimams aptikti, ir pasižymi dideliu tikslumu, patikimumu ir interpretacijos galimybėmis, kurios yra būtinos siekiant apsisaugoti nuo kylančių kibernetinių grėsmių.

2.6. Sprendimo koncepcijos išvados

1. Sukurtas sprendimo konceptas, leidžiantis lengvai prijungti metodą prie komunikacijos programos;
2. Metodas analizuoja komunikacijos platformoje esančias nuorodas, failus ir tekstą;
3. Aprašytas mašininio mokymo modelio apmokymas ir integravimas naudojant "*Random Forest*" algoritmą;

3. Sprendimo prototipas

3.1. Sistemos architektūra



8 pav. Duomenų viliojimo atakos atpažinimo serviso architektūra

3.2. Duomenų gavimo sluoksnis

Daugiausia dėmesio skiriame galimai kenkėjiškų žinučių iš populiarios bendravimo platformos „*Discord*“ rinkimui ir analizei. Duomenų rinkimo procesas apima „*Discord*“ žinučių stebėjimą per „*Discord*“ *API* naudojant boto realizaciją. Botas renka keletą svarbiausių kiekvieno pranešimo duomenų: pranešimo turinį, įterptas nuorodas, failų priedus, naudotojo informaciją (ID ir naudotojo vardą), laiko žymą ir kanalo kontekstą.

Šie neapdoroti duomenys sudaro mūsų sukčiavimo aptikimo sistemos pagrindą.

3.3. Nuorodų analizė

Šioje dalyje aprašomas nuorodų analizės serviso veikimas, skirto aptikti potencialiai kenkėjiškus ir apgaulingus nuorodų adresus, architektūra ir algoritmai. Servisas naudoja nuorodų adresus iš „*RabbitMQ*“ pranešimų eilės, atlieka kelis žingsnius kiekvienos nuorodos analize, perduoda

rezultatus tolimesniam apdorojimui kitiems servisams. Nuorodų analizė susideda iš struktūros analizės ir patvirtinimo, domeno komponentų išskyrimo ir analizės, nuorodos adresų *SSL* sertifikatų patikrinimo, įtartinų raktinių žodžių ir šablonų patikrinimo ir bendros domeno reputacijos nustatymo. Derindama šiuos signalus paslauga gali veiksmingai ir tiksliai nustatyti nuorodos, keliančius pavojų saugumui.

3.3.1. Nuorodų analizės serviso architektūra

Nuorodų analizės servisas įgyvendintas *Python* programavimo kalba ir sukurtas taip, kad veiktų kaip atskiras mikroservisas, gaunantis nuorodas iš *RabbitMQ* pranešimų eilės.

Serviso paleidimo metu inicializuoja savo modelių duomenų bazes vidinėje atmintyje. Kai į paskirtą įvesties eilę gaunamas pranešimas su nuorodų adresais, servisas peržiūri kiekvieną nuorodos adresą ir atlieka analizę. Tada apibendrinti pranešimo analizės rezultatai perduodami failų analizės servisui arba teksto analizės servisui, priklausomai nuo to, ar pranešime taip pat buvo failų priedų.

Faktinė nuorodos analizės logika yra įtraukta į atskirą metodą, kad pranešimo tvarkymas būtų paprastas. Tai leidžia analizės etapams vystytis nepriklausomai nuo eilės įgyvendinimo.

3.3.2. Nuorodų analizės metodika

3.3.2.1. Nuorodos analizė ir patvirtinimas

Pirmasis žingsnis - neapdorotos nuorodos eilutės išskaidymas į sudedamąsias dalis (schema, domenas, kelias, užklausos eilutė ir t. t.) naudojant „*urllib.parse“* biblioteką. Taip ne tik išgaunamos tolesnei analizei reikalingos dalys, bet ir patvirtinama, kad nuoroda yra tinkamai suformuota. Netinkamai suformuoti nuorodų adresai iš karto kelia įtarimų.

3.3.2.2. Domenų analizė

Domenas yra vienas iš stipriausių signalų nustatant, ar nuorodos adresas yra įtartinas. Paslauga naudoja „*tldextract“* biblioteką, kad išskaidytų domeną į subdomeną, pagrindinį domeną ir priesagą (*TLD*). Tada ji tikrina:

- Neįprastai ilgus domenus. Jei domene daugiau nei 50 simboliu, pažymima, jog tai potenciali grėsmė;
- Per didelis subdomenų skaičius. Jei subdomenų kiekis yra didesnis už 2, nuoroda žymima kaip potenciali grėsmė

3.3.2.3. SSL patikra

HTTPS nuorodų adresų atveju servisas bando patikrinti *SSL* sertifikatą naudodama „*ssl*“ ir „*socket*“ modulius. Ji tikrina, ar sertifikatas galioja domenui, ar jį išdavė patikimas *CA* ir ar jo galiojimo laikas nėra pasibaigęs arba netrukus baigsis. Negaliojantys sertifikatai yra aiškus galimo sukčiavimo arba kenkėjiškos programinės įrangos požymis.

3.3.2.4. Nuorodų šablonų atitikimas

Toliau servisas tikrina visą nuorodos adresą pagal apibrėžtą reguliariųjų išraiškų rinkinį, ieškodama šablonų, būdingų sukčiavimo ir kenkėjiškos programinės įrangos nuorodų adresams. Pavyzdiniai šablonai yra šie:

- Klaidinantys prisijungimo, saugumo, bankininkystės raktiniai žodžiai;
- Per didelis nuorodos kodavimo naudojimas. Kodavimo atpažinimui naudojama reguliarioji išraiška kuri ieško „%“ ženklų kiekio, jis žymi nuorodos kodavimą, jei tokių ženklų randama daugiau nei trys, nuoroda žymima kaip potenciali grėsmė;
- Vietoj domenų vardų naudojami IP adresai;
- Neįprasti prievadų numeriai. Jei prievadai neatitinka 80, 433, 8080, 8443, jie žymimi kaip potenciali grėsmė.

Šie šablonai inicijuojami iš konfigūracijos failo paleidimo metu ir gali būti lengvai atnaujinami.

3.3.2.5. Domeno reputacijos patikrinimas

Galiausiai servisas naudojasi trečiųjų šalių domenų reputacijos paslaugomis, kad patikrintų, ar domenas nebuvo susijęs su kenkėjiška veikla. Šiuo metu ji naudoja „*Google Safe Browsing API*“, jei pateikiamas *API* raktas, tačiau architektūra leidžia lengvai pridėti kitas paslaugas, pavyzdžiui, „*VirusTotal*“.

3.3.2.6. Rizikos vertinimas ir persiuntimas

Remdamasi analizės etapų rezultatais, paslauga kiekvienam nuorodos adresui priskiria bendrą rizikos balą. Jei balas yra žemesnis už nustatytą „įtartiną“ ribą, ji uždeda vėliavėlę. Ji taip pat nustato atskirą vėliavėlę, jei nuorodos rizika viršija aukštesnę „kenkėjišką“ ribą, dėl kurios gali būti imtasi papildomų veiksmų.

Paslauga prie pradinio pranešimo prideda nuorodos analizės rezultatus ir bet kokią failų analizės informaciją, pažymi įtartinus arba kenkėjiškus subjektus ir praturtintą pranešimą perduoda kitam servisui. Nuorodos kenkėjiškumo balo taisyklės pateiktos **1 lentelėje**. Jei atitikusių taisyklių suma viršija vienetą, ji suapvalinama iki vieneto.

1 lentelė Nuorodos taisyklių reikšmingumo įverčiai

Taisyklė	**Taisyklės įvertis**
1. Domeno simbolių skaičius didesnis nei 50 simbolių	0.3 balo
2. Įtartinų raktinių žodžių atitikimas	0.4 balo
3. Subdomenų kiekis didesnis nei 2	0.2 balo
4. Įtartinas nuorodos šablonas	0.6 balo
5. Užkoduota nuoroda naudojant specialiuosius simbolius	0.4 balo
6. Naudojamas IP adresas vietoj domeno	0.7 balo
7. Nuorodoje randamas neleistinas prievado numeris	0.5 balo
8. Nenaudojamas SSL sertifikatas	1 balas
8.1. Sertifikatas nevaldūs	0.9 balo
8.2. Sertifikatas pasibaigęs	0.9 balo
8.3. Nepavyko gauti SSL sertifikato	1 balas
9. Rizikingumo įvertinimas iš „Google Safe Browsing API“	[0, ..., 1] balo

3.4. Prisegtų failų analizė

3.4.1. Failų analizės serviso architektūra

Failo analizės servisas įgyvendinta kaip atskiras „*Python*“ mikroservisas, kuris naudoja failų priedus iš specialios „*RabbitMQ*“ pranešimų eilės. Aukšto lygio architektūra yra tokia:

Paleidimo metu servisas inicializuoja savo konfigūraciją, įskaitant įtartinų failų plėtinių apibrėžimą ir ryšį su „*VirusTotal*“ *API*. Kai gaunamas pranešimas su failų priedais, paslauga iteruoja kiekvieną priedą, atsisiunčia failą į saugią laikinąją vietą ir perduoda jį analizės moduliui. Tada visų priedų apibendrinti analizės rezultatai perduodami teksto analizės servisui, kad būtų toliau tikrinamas turinys.

3.4.2. Failų analizės metodika

Failų analizės procesas susideda iš kelių etapų, kuriais išrenkami svarbūs failų atributai ir įvertinama galima saugumo rizika.

3.4.2.1. Failų tipo aptikimas

Pirmasis žingsnis - nustatyti failo tipą naudojant „*python-magic“* biblioteką. Ji pateikia ir *MIME* tipą, ir aprašomojo tipo eilutę, kurie naudingi nustatant, ar failas yra vykdomasis, ar kitaip įtartinas. Dažniausiai pasitaikantys pavojingi failų tipai yra *PE* vykdomieji failai, scenarijai ir vykdomieji archyvai.

3.4.2.2. Metaduomenų išskyrimas

Toliau servisas išgauna pagrindinius failo metaduomenis, įskaitant:

- Originalų failo pavadinimą;
- Failo dydį
- *SHA-256* maišos funkciją

Failo maišos funkcija yra ypač svarbi, nes jis yra unikalus failo identifikatorius ir leidžia atlikti užklausas išorinėse kenkėjiškų programų duomenų bazėse. Maišos funckija apskaičiuojama naudojant „*hashlib“* biblioteką ir buferinį skaitymą, kad būtų galima efektyviai apdoroti didelius failus.

3.4.2.3. Rizikos veiksnių analizė

Naudojantis surinktais failų atributais, servisas atlieka keletą euristinių patikrinimų, kad nustatytų galimus rizikos veiksnius:

- Įtartini failų plėtiniai, susiję su kenkėjiškomis programomis, pvz.: .exe .bat .cmd .ps1.
- Vykdomosios programos, kurių failų dydis yra mažas, o tai gali reikšti supakuotą arba užmaskuotą kenkėjišką programinę įrangą.
- Įtartini *MIME* tipai, neatitinkantys failo plėtinio
- Dinaminio vykdymo galimybės, pvz., *PE* failai, scenarijai

Šie rizikos veiksniai sudaromi į sąrašą ir naudojami pažymint failą kaip potencialiai įtartiną. Tai leidžia nustatyti didelės rizikos failų prioritetus tolesnei analizei.

3.4.2.4. Integracija su „*VirusTotal*“

Dėl failų, kurie pažymėti kaip įtartini, servisas užklausia „*VirusTotal*“ duomenų bazę, naudodama failo maišos funkciją. *VirusTotal* apibendrina skenavimo rezultatus iš daugiau nei 70 antivirusinių variklių ir pateikia išsamią informaciją apie grėsmes. Servisas naudoja „*VirusTotal API v4*“, kad patikrintų, ar žinoma failo maišos funckija, ir gautų tikrinimo rezultatus. Jei vienas ar daugiau variklių aptinka failą kaip kenkėjišką, analizės rezultatuose jis pažymimas kaip kenkėjiškas.

3.4.2.5. Rezultatų saugojimas ir greitis

Siekiant išvengti perteklinės analizės ir optimizuoti našumą, paslauga palaiko anksčiau analizuotų failų talpyklą, kaip raktą naudodama *SHA-256* maišos funkciją. Jei gaunamas failas atitinka talpyklos maišos funkciją, ankstesni analizės rezultatai grąžinami iš karto, neanalizuojant failo iš naujo.

3.4.3. Integracija ir mastelio keitimas

Naudojant pranešimų eilių architektūrą, failų analizės servisą galima horizontaliai plėsti, kad būtų galima apdoroti didelius failų priedų kiekius. Keli paslaugos egzemplioriai gali vartoti iš tos pačios *RabbitMQ* eilės, todėl galima lygiagrečiai apdoroti failus. Naudojant talpyklą dar labiau sumažinamas pasikartojantis darbas tarp failų.

3.4.4. Rizikos vertinimas

Rizika įvertinama pagal kiekvieną **2 lentelėje** pateiktų taisyklių. Taisyklių atitinkančių įverčių kiekis sumuojamas ir siunčiamas sekančiam servisui. Jei suma viršija vienetą, ji suapvalinama iki vieneto.

2 lentelė Failo taisyklių įverčių vertė

Taisyklė	**Taisyklės įvertis**
1. Įtartinas failo plėtinys	0.4 balo
2. Jei tai paleidžiamasis failas	0.3 balo
3. Įtartinas MIME failo tipas	0.3 balo
4. „VirtusTotal“ įvertis	[0...1] balo
5. Rastas paslėptas failas	0.4 balo

3.5. Turinio analizė

3.5.1. Turinio analizės serviso architektūra

Teksto analizės servisas įgyvendintas kaip atskira mikroservisas, kuris naudoja pranešimus iš specialios „*RabbitMQ*“ pranešimų eilės.

Paleidimo metu servisas įkelia sukonfigūruotą mašininio mokymosi modelį ir inicializuoja savo modelių duomenų bazes. Gavęs pranešimą, servisas ištraukia atitinkamą turinį ir perduoda jį per analizės vamzdyną. Analizės rezultatai, įskaitant modelio prognozes, nustatytus modelius ir kontekstinius rizikos veiksnius, apibendrinami ir perduodami kitai vamzdyno paslaugai, paprastai galutinei analizės paslaugai, kuri sujungia kelių analizės etapų rezultatus.

Paslauga sukurta taip, kad palaikytų įvairių tipų mašininio mokymosi modelius, įskaitant tradicinius modelius, pavyzdžiui, „*Random Forest*“, ir gilaus mokymosi modelius, pavyzdžiui, transformatorius. Modelio tipą galima konfigūruoti atsižvelgiant į našumo reikalavimus ir turimus išteklius.

3.5.2. Analizės metodika

Teksto analizės procesas susideda iš kelių etapų, kuriuose, siekiant nustatyti potencialias grėsmes, naudojamos ir mašininio mokymosi prognozės, ir taisyklėmis pagrįstas modelių atitikimas.

3.5.2.1. Modelio prognozavimas

Pirmajame analizės etape pranešimo turinys perduodamas per įkeltą mašininio mokymosi modelį. Paslauga palaiko dviejų tipų modelius:

Atsitiktinis miškas. Tradicinis mašininio mokymosi modelis, apmokytas pagal pažymėtą duomenų rinkinį, sudarytą iš sukčiavimo ir teisėtų pranešimų. Modelis naudoja teksto žodžių maišelio atvaizdavimą ir sukuria tikimybės balą, nurodantį tikimybę, kad pranešimas yra kenkėjiškas.

Transformatorius. Gilaus mokymosi modelis, pagrįstas transformatoriaus architektūra, iš anksto apmokytas naudojant didelį tekstinių duomenų korpusą ir pritaikytas aptikti apgaulingus pranešimus. Šis modelis gali fiksuoti sudėtingesnius semantinius ir sintaksinius teksto modelius ir dažnai užtikrina didesnį tikslumą nei tradiciniai modeliai.

Modelio prognozavimo etape grąžinamas tikimybės balas ir binarinė žinutės klasifikacija kaip įtartinos arba neįtartinos, remiantis konfigūruojama riba.

3.5.2.2. Teksto atitikimas statinėms taisyklėms

Be modelio prognozių, servisas atlieka teksto analizę, kad nustatytų žinomus kenkėjiško turinio rodiklius. Tai apima:

Duomenų viliojimo frazės. Servisas tvarko duomenų bazę, kurioje pateikiamos įprastos frazės ir raktiniai žodžiai, dažnai naudojami bandymuose sukčiauti, pavyzdžiui, „reikia imtis skubių veiksmų“ arba „patikrinkite savo paskyrą“. Jei žinutėje randama tokių frazių, padidinamas įtarumo balas.

Nuorodų šablonai. Pranešimai, kuriuose yra neįprastai daug nuorodų adresų arba specifinių modelių, susijusių su sukčiavimu (pvz., IP adresai), pažymimi kaip potencialiai pavojingi.

Piniginės simboliai. Dažnas valiutos simbolių ar su pinigais susijusių raktažodžių paminėjimas gali rodyti finansinį sukčiavimą ar bandymą sukčiauti.

Modelių atitikimo etape pateikiamas nustatytų modelių ir susijusių rizikos veiksnių sąrašas.

3.5.2.3. Konteksto analizė

Paskutiniame analizės etape vertinamas pranešimo kontekstas, siekiant nustatyti papildomus rizikos veiksnius. Tai apima:

Skubumo rodikliai. Žodžiai ir frazės, kurios sukuria skubos ar spaudimo jausmą, pavyzdžiui, „nedelsiant imtis veiksmų“ arba „riboto laiko pasiūlymas“, dažnai naudojami bandymuose sukčiauti, siekiant priversti naudotoją suklysti.

Grasinanti kalba. Pavyzdžiai: „paskyros sustabdymas“ arba ‚neautorizuota prieiga‘, gali būti bandymas įbauginti naudotoją, kad jis atskleistų jautrią informaciją.

Atlygio pažadai. Įtarimą keliantys pasiūlymai dėl prizų, apdovanojimų ar išskirtinių pasiūlymų yra įprasta taktika, naudojama sukčiavimo ir apgaulės pranešimuose.

Kontekstinės analizės etape pagal šių rodiklių buvimą ir dažnumą priskiriami rizikos balai.

3.5.3. Mašininio mokymo modelio mokymas ir diegimas

Teksto analizės servisas, atlikdama prognozes, remiasi iš anksto apmokytais mašininio mokymosi modeliais. Modelių mokymo procesas apima šiuos etapus:

7. Duomenų rinkimas;
8. Pirminis teksto apdorojimas;
9. Modelio mokymas;
10. Modelio įvertinimas;
11. Modelio diegimas;

3.5.3.1. Duomenų rinkimas

Duomenų rinkiniai imami iš „Kaggle“ svetainės. Joje sistemos naudotojai aprašo skirtingus duomenų rinkinius skirtingiems panaudojimo atvejams. Norint padidinti duomenų kiekį apie kenksmingas žinutes apie 20 % duomenų sugeneruota naudojant dirbtinį intelektą.

Duomenų viliojimo atakos atpažinimo modeliui naudojant mašininį mokymą būtina turėti klasifikuotus duomenis. Rasta daug skirtingų duomenų. Tačiau kiekvieno rasto duomenų rinkinio imtis labai maža, mažiau 5000 duomenų eilučių.

Todėl pasirinkta susirinkti populiariausius duomenų rinkinius, ir juos sujungti į vieną.

3 lentelė Duomenų rinkiniai

Duomenų rinkinys	**Failo pavadinimas**	**Metai**	**Klasės**
Email Spam or Not [25]	spam_dataset.csv	2024	*Spam / Ham*
SMS Spam Detection Dataset [26]	spam_sms.csv	2024	*Spam / Ham*
SMS Spam Collection Dataset [27]	spam.csv	2017	*Spam / Ham*
Spam SMS Classification Using NLP [28]	Spam_SMS.csv	2024	*Spam / Ham*

3.5.3.2. Duomenų apdorojimas

Duomenų agregavimo klasė skirta keliems duomenų rinkiniams sujungti ir iš anksto apdoroti. Kaip parametrą ji priima išvesties kelią, kuris nurodo, kur bus išsaugotas sujungtas duomenų rinkinys. Klasė apibrėžia standartinę sujungto duomenų rinkinio schemą, kurią sudaro du stulpeliai: „*content*“ (eilutė) - pranešimo turinys ir „*is_phishing*“ (sveikasis skaičius) - sukčiavimo žyma.

Duomenų agregavimo klasė palaiko kiekvieno duomenų rinkinio atvaizdavimo konfigūraciją, nurodydama, kaip duomenų rinkinio stulpeliai turėtų būti atvaizduoti į standartinę schemą. Tai leidžia tvarkyti duomenų rinkinius su skirtingais stulpelių pavadinimais ir formatais. Atvaizdavimai

apibrėžiami standartizuotame žodyne, kurio raktai yra duomenų rinkinio failų pavadinimai, o reikšmės - žodynai, kuriuose pateikiami stulpelių atvaizdavimai ir etikečių transformacijos.

Duomenų apdorojimo funkcija kaip parametrus priima failo kelią ir pasirinktinę numatytąją etiketę. Jis nuskaito duomenų rinkinį naudodamas „*pandas*“ biblioteką ir taiko atitinkamą atvaizdavimo konfigūraciją stulpelių pavadinimams ir etiketėms standartizuoti. Jei duomenų rinkiniui nenurodytas joks atvaizdavimas, metodas bando automatiškai aptikti atitinkamus stulpelius, remdamasis bendraisiais pavadinimų susitarimais.

Neapibrėžtų duomenų funkcija apdoroja duomenų rinkinius be iš anksto nustatytų atvaizdavimų. Jis nustato turinio ir etikečių stulpelius ieškodamas konkrečių raktinių žodžių stulpelių pavadinimuose. Tada metodas bando konvertuoti etiketes į standartinį formatą (0 - saugus, 1 - apgaulingas) pagal numanomą etikečių tipą (loginis, skaitinis arba tekstinis).

Duomenų apvalymo funkcija yra atsakinga už kiekvieno pranešimo tekstinio turinio valymą ir normalizavimą. Ji pašalina pradinius ir (arba) galinius baltuosius ženklus, kelis tarpus suveda į vieną tarpą ir sutrumpina pernelyg ilgas eilutes iki didžiausio 10 000 simbolių ilgio.

Duomenų rinkinių sujungimo funkcija kaip parametrą priima duomenų rinkinių konfigūracijų sąrašą. Kiekviena konfigūracija yra žodynas, nurodantis duomenų rinkinio failo kelią, stulpelių atitikmenis (neprivaloma) ir numatytąją etiketę (neprivaloma). Metodas iteruoja per konfigūracijas, apdoroja kiekvieną duomenų rinkinį.

Apdorojus visus duomenų rinkinius, jie sujungiami į vieną sujungtą duomenų rinkinį. Pašalinamos pasikartojančios eilutes pagal turinio stulpelį, išsaugodamos natūralų klasių pasiskirstymą. Galiausiai sujungtas duomenų rinkinys įrašomas į inicializacijos metu nurodytą išvesties kelią.

Naudoti duomenų rinkiniai:

- Spam_sms.csv: Duomenų rinkinys, kuriame yra *SMS* žinučių, pažymėtų kaip šlamštas arba šlamštas.
- train.csv: duomenų rinkinį, kuriame yra *SMS* žinučių su šlamšto etiketėmis.
- Spam_SMS_1.csv: Kitas *SMS* žinučių duomenų rinkinys su šlamšto etiketėmis.
- sms_spam.csv: *SMS* žinučių, paženklintų *spam* arba *ham*, duomenų rinkinys.
- spam_dataset.csv: Žinučių su žymomis „*spam/not_spam*“ duomenų rinkinys.

Kiekvienas duomenų rinkinys turi skirtingą schemą ir formatą, todėl, norint suvienodinti stulpelius ir etiketes, reikia specialių atvaizdavimų. Atvaizdavimai apibrėžiami bendrame sąraše, kuriame nurodomas failo kelias, stulpelių atvaizdavimai ir etikečių transformacijos kiekvienam duomenų rinkiniui.

Duomenų agregavimo klasė sėkmingai sujungė penkis „*phishing“* duomenų rinkinius į vieną standartizuotą duomenų rinkinį. Gautame duomenų rinkinyje yra 6836 eilutės, kuriose yra 5719 saugūs pranešimai (0 etiketė) ir 1117 apgaulingų pranešimų (1 etiketė). Sujungtą duomenų rinkinį sudaro du stulpeliai: „*content*“ (eilutė), kurioje pateikiamas pranešimo tekstas, ir „*is_phishing*“ (sveikasis skaičius), nurodantis sukčiavimo etiketę.

Sujungimo proceso metu duomenų agregavimo klasė taikė nurodytus atvaizdavimus, kad suvienodintų kiekvieno duomenų rinkinio stulpelių pavadinimus ir etiketes. Ji taip pat tvarkė trūkstamas reikšmes ir valė teksto turinį, kad būtų užtikrintas duomenų vientisumas. Klasė

automatiškai nustatė duomenų rinkinių be iš anksto nustatytų atvaizdavimų turinio ir etikečių stulpelius, taip parodydama savo lankstumą tvarkant įvairias duomenų struktūras.

Galutinis sujungtas duomenų rinkinys gerai tinka mašininio mokymosi modeliams, skirtiems sukčiavimui aptikti, mokyti. Jame pateikiamas didelis ir įvairus apgaulingų ir saugių pranešimų rinkinys, leidžiantis kurti patikimas aptikimo sistemas. Standartizuotas duomenų rinkinio formatas taip pat palengvina integraciją su įvairiomis mašininio mokymosi bibliotekomis ir įrankiais.

Neapdorotas žinučių tekstas išvalomas, simbolizuojamas ir transformuojamas į modelio mokymui tinkamą formatą. Tai gali apimti sustabdytų žodžių pašalinimą, kamieninių žodžių išbraukimą ir teksto konvertavimą į skaitines reprezentacijas, pavyzdžiui, žodžių įterpimą.

3.5.3.3. Modelio mokymas

Duomenų rengimas

Modelio treniravimo klasė pradedama nuo duomenų rinkinio įkėlimo ir paruošimo mokymui. Duomenų rinkinį sudaro *SMS* žinutės, pažymėtos kaip apgaulingos (1) arba teisėtos (0). Duomenų parengimo etapai apima šiuos veiksmus:

- Duomenų rinkinio įkėlimas iš *CSV* failo naudojant pandas biblioteką.
- Duomenų valymas pašalinant eilutes, kurių stulpeliuose „*content*“ ir „*is_phishing*“ trūksta reikšmių.
- Duomenų aibės subalansavimas, siekiant išspręsti klasių disbalanso problemą. Atliekama mažesnė daugumos klasės imčių atranka, kad ji atitiktų mažumos klasės imčių skaičių, taip užtikrinant vienodą abiejų klasių atstovavimą mokymo metu.
- Subalansuoto duomenų rinkinio padalijimas į mokymo (70 %), testavimo (15 %) ir patvirtinimo (15 %) rinkinius, naudojant sluoksninę atranką, kad būtų išlaikytas klasių pasiskirstymas.

Požymių išskyrimas

Teksto duomenims paversti į „*Random Forest*“ klasifikatoriui tinkamą formatą naudoja *TF-IDF* (terminų dažnio ir atvirkštinio dokumento dažnio) vektorizavimo metodą. Teksto pranešimams transformuoti į *TF-IDF* požymių matricą naudojamas „*scikit-learn*“ bibliotekos „*TfidfVectorizer*“ vektorizatorius. Vektorizatorius konfigūruojamas naudojant šiuos parametrus:

- *max_features*: Didžiausias galimas atsižvelgti požymių skaičius (nustatyta 5000).
- *ngram_range*: Išskiriamų n-gramų intervalas (nustatyta (1, 2), atsižvelgiant ir į unigramas, ir į bigramas).
- *stop_words*: Stopžodžių, kurių reikia pašalinti iš požymių aibės, sąrašas (nustatyta į „*english*“).

Vektorizatorius pritaikomas mokymo duomenims ir naudojamas tekstiniams pranešimams transformuoti į požymių matricas mokymui, testavimui ir tikrinimui.

Modelio mokymas

Atsitiktinio miško klasifikatorius inicializuojamas šiais parametrais:

- *n_estimators*: medžių skaičius miške (nustatyta, kad jų skaičius yra 200).
- *max_depth* (didžiausias gylis): Didžiausias kiekvieno medžio gylis (nustatytas kaip 20).

- *min_samples_split*: Mažiausias pavyzdžių skaičius, reikalingas vidiniam mazgui suskaidyti (nustatytas kaip 5).
- *min_samples_leaf*: Mažiausias mėginių skaičius, kurio reikia, kad būtų lapinis mazgas (nustatyta reikšmė 2).
- *random_state*: Atsitiktinė sėkla atkuriamumui užtikrinti (nustatyta į 42).
- *n_jobs*: Lygiagrečiai vykdomų užduočių skaičius (nustatyta į -1, naudojami visi turimi procesoriaus branduoliai).

Mokymo procesas vykdomas nustatytą epochų skaičių (šiame įgyvendinime nustatyta 10). Kiekvienoje epochoje atsitiktinio miško klasifikatorius mokomas pagal mokymo aibę. Tada apmokytas modelis vertinamas mokymo, testavimo ir patvirtinimo.

3.5.3.4. Modelio įvertinimas

Įvertinamas apmokyto modelio veikimas naudojant keletą rodiklių:
- Tikslumas: Teisingai suklasifikuotų pavyzdžių dalis.
- Tikslumas: Teisingų teigiamų prognozių dalis tarp visų teigiamų prognozių.
- Atšaukimas: Tikrų teigiamų prognozių dalis tarp visų faktinių teigiamų pavyzdžių.
- F1 rezultatas: tikslumo ir atšaukimo harmoninis vidurkis, kuris yra subalansuotas modelio našumo matas.

Šie rodikliai apskaičiuojami naudojant „*scikit-learn“* funkciją „*classification_report()“*, kuri pateikia išsamią modelio našumo kiekvienai klasei išklotinę.

Klasėje taip pat stebimas geriausiai veikiantis modelis visose epochose pagal patvirtinimo tikslumą. Jei randamas naujas geriausias modelis, jis išsaugomas kartu su atitinkamais epochų rezultatais.

3.5.3.5. Modelio rezultatas.

Eksperimento rezultatai parodė, kad „*Random Forest*“ klasifikatorius yra veiksmingas nustatant bandymus sukčiauti. Visuose vertinimo rinkiniuose modelis pasiekė aukštus tikslumo rezultatus, o geriausias patvirtinimo tikslumas siekė 0,9173. Tikslumo ir atšaukimo rodikliai taip pat buvo nuolat aukšti, o tai rodo, kad modelis geba teisingai atpažinti tiek apgaulingus, tiek teisėtus pranešimus su minimaliu klaidingai teigiamų ir klaidingai neigiamų rezultatų skaičiumi.

Metrikų palyginimas pagal epochas atskleidė, kad modelio našumas per mokymo epochas nuolat gerėjo, o geriausi rezultatai paprastai buvo stebimi vėlesnėse epochose. Našumas atspindi modelio mokymosi pažangą ir rodo aiškią mokymo, testavimo ir patvirtinimo tikslumo konvergenciją.

Modelio treniravimo metu pateikiami rezultatai treniruojant modelį keliomis iteracijomis, keičiant būsenas. Modelio treniravimo tikslumas apskaičiuojamas pateikiant treniruotus duomenis „Random Forest“ algoritmui ir suskaičiuojant vidutinę teisingai prognozuojamų rezultatų dalį.

Modelio tinkamai atrinktų duomenų viliojimo atakų prognozės procentinė dalis pagal:
- Treniravimo duomenis – 92,66%
- Testavimo duomenis – 91,73%
- Patikros duomenis – 91,73%

3.6. Bendras rizikingumo balo apskaičiavimas

Norint apskaičiuoti galutinę sukčiavimo tikimybę pagal atakos aptikimo sistemą, reikia taikyti metodą, pagal kurį atsižvelgiama į skirtingus analizės rezultatus. Kai turinys, priedai ir nuorodos nepriklausomai vertinami pagal skalę nuo 0 iki 1. Imamas didžiausias įvertis. Jei didžiausias įvertis didesnis nei 0,6 įvertinimas, tai žinutė žymima kenkėjiška, jei virš 0,4 pažymima.

Jeigu siunčiama informacija yra išanalizuota, tačiau nei vienas analizavimo rezultatas neviršija 0.6 sverto, tada naudojama bendra apskaičiavimo formulė:

$$M = W_U \cdot S_U \cdot A_U + W_F \cdot S_F \cdot A_F + W_T \cdot S_T \cdot A_T \quad (1)$$

Formulės elementų paaiškinimas:
M – Bendras kenkėjiškumo balo įvertis (nuo 0 iki 1)

W_U – Nuorodos kenksmingumo įverčio svoris (konstanta, kiek svarbus nuorodos įvertis 0.25)

S_U – Nuorodos kenksmingumo įvertis (atitikusių nuorodos taisyklių įverčių suma nuo 0 iki 1)

A_U – Nusako ar nuorodos analizė atlikta ar neatlikta (1 – taip, 0 – ne)

W_F – Failo kenksmingumo įverčio svoris (konstanta, kiek svarbus failo įvertis 0.25)

S_F – Failo kenksmingumo įvertis (atitikusių nuorodos taisyklių įverčių suma nuo 0 iki 1)

A_F – Nusako ar failo analizė atlikta ar neatlikta (1 – taip, 0 – ne)

W_T – Teksto kenksmingumo įverčio svoris (konstanta, kiek svarbus teksto įvertis 0.5)

S_T – Teksto kenksmingumo įvertis (modelio įvertis nuo 0 iki 1)

A_T – Nusako ar teksto analizė atlikta ar neatlikta (1 – taip, 0 – ne)

3.7. Sprendimo prototipo išvados

1. Sukurta metodo architektūra, įgyvendintas koncepto prototipas;
2. Nuorodos analizuojamos aprašytomis taisyklėmis ir integruojant "*Google Safe Browsing API*" sąsają;
3. Failai analizuojami aprašytomis taisyklėmis ir integruojant "*VirusTotal API*" sąsają;
4. Surinktas išvalytas ir paruoštas duomenų rinkinys mašininio mokymo algoritmui. Apmokytas modelis naudojant "*Random Forest*" algoritmą;
5. Sukurta formulė apskaičiuoti bendrą kenkėjiškumo balą;

4. Eksperimentinis prototipo tyrimas

Scenarijai skirti pagerinti metodo patikimumą realiomis sąlygomis.

4.1. Tyrimo parametrai

4.1.1. Sistemos komponentų patikimumo parametrai

Nustatoma bendrinės sistemos patikimumo parametrai ir atskiri jos komponentai naudojant įverties parametrus:

- Tikslumą (angl. *precision*);
- Atsišaukimą (angl. *recall*);
- F1 balą (angl. *F1-score*);
- Neigiamus teigiamus (angl. *false positive*);
- Neigiamus neigiamus (angl. *false negative*).

4.1.2. Mašininio mokymo treniravimo parametrus

Tiriama, koks modelio tikslumas keičiant treniravimo parametrus:

- Naudojant balansuotą duomenų rinkinį;
- Naudojant nebalansuotą duomenų rinkinį;
- Keičiant atsitiktinio medžio šakų kiekį;
- Keičiant mokymo iteracijų kiekį;

4.1.3. Atminties resursų naudojimas

Tikrinama, kiek operatyviosios atminties naudojama užkrauti mašininio mokymo modelį, realiu laiku atlikti žinučių analizę matuojant kilobaitais (*kB*)

4.2. Tyrimo metodika

4.2.1. Sistemos komponentų patikimumo skaičiavimo metodika

URL analizės patikimumo įvertinimas.

Etapai, kuriuos reikia padaryti norint teisingai patikrinti nuorodų analizės komponento veikimo tikslumą:

- Duomenų rinkinio paruošimas. Sukurti arba gauti duomenų rinkinį iš žinomų duomenų viliojimo atakų nuorodų ir paprastų nuorodų. Proporcija tarp atakų ir paprastų nuorodų turi būti pasiskirsčiusi tolygiai;
- Testavimo procedūra. Siųsti nuorodas į komunikacijos kanalą ir surinkti metodo gautus klasifikacijos atsakymus apie nuorodas. Kiekvienai nuorodai saugoti tikrą atsakymą ir gautą atsakymą. Apskaičiuoti rezultato gavimo laiką.
- Metrikų apskaičiavimas. Apskaičiuoti tikslumą, atsišaukimą, F1 balą, neigiamus teigiamus ir neigiamus neigiamus įverčius, tam naudoti šias formules:

$$Accuracy\ (tikslumas) = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

$$Precision\ (tikslumas) = \frac{TP}{TP + FP} \quad (3)$$

$$Recall\ (atsišaukimas) = \frac{TP}{TP + FN} \quad (4)$$

$$F1\ score\ (F1\ balas) = \frac{2TP}{2TP + FP + FN} \quad (5)$$

– Problemų analizė. Su kategorizuoti neigiamus teigiamus rezultatus pagal nuorodos charakteristikas (domeną, parametrus)

Teksto analizės patikimumo įvertinimas.

– Duomenų rinkinio paruošimas. Paruošti 2000 žinučių duomenų rinkinį, pusė jų turi būti duomenų viliojimo atakų tekstai.
– Našumo analizė. Sukurti supainiojimo matricas (angl. *confusion matrices*).
– Požymių analizė. Sukurti požymių svarbumo analizę su turimais įrankiais ir nurodyti pagrindinius 10-20 įrašų kurie lemia klasifikacijos davimą.

4.2.2. Mašininio mokymo treniravimo metodika

– Duomenų rinkinio sukūrimas. Rasti arba sukurti duomenų rinkinį atlikti prototipo testavimams;
– Eksperimentiniai parametrai:
  - Naudoti balansuotą ir nebalansuota duomenų rinkinį, palyginti metrikas;
  - Pakeisti skirtingą šakų kiekį atsitiktinio medžio klasifikatoriui, 10, 50, 100, 200, 500.
  - Pakeisti atsitiktinio medžio gylio parametrą, 0, 10, 20, 50, 100;
– Modelio tikslumas apskaičiuojamas naudojant tikslumo, atsišaukimo, F1 balo, neigiamų teigiamų ir neigiamų neigiamų įverčių rezultatus.

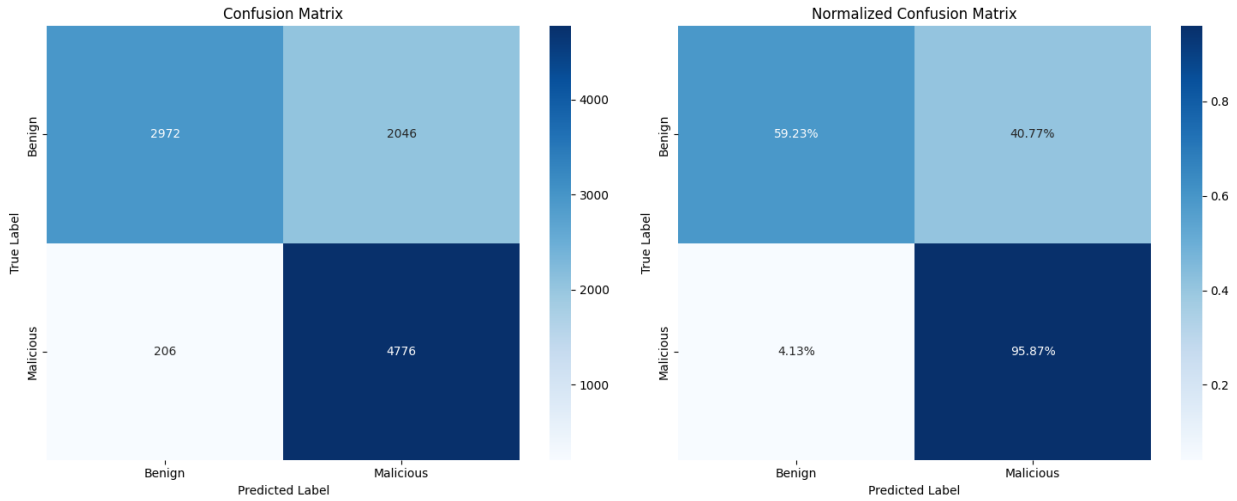
4.2.3. Atminties resursų naudojimo metodika

Atminties resursų sąnaudos renkamos paleidus prototipą ir duodant skirtingą kiekį užklausų sukurtam prototipui. Informacija apie sąnaudas surašoma į lentelę su laiko žymomis.

4.3. Tyrimo rezultatai

4.3.1. Nuorodų analizės patikimumo įvertinimo rezultatai

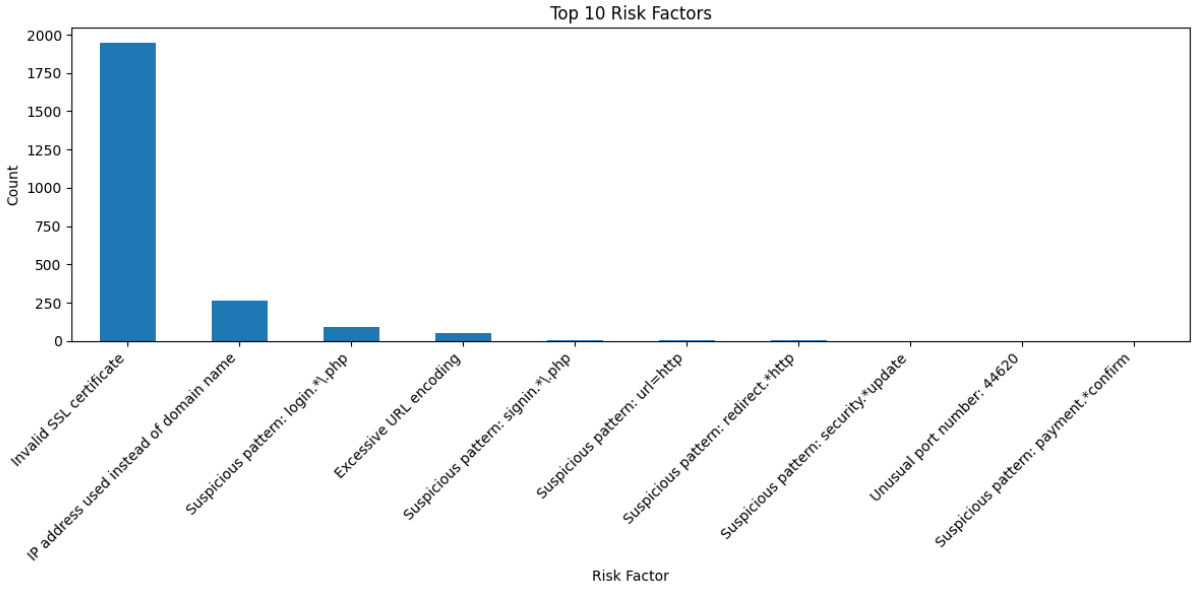
Pirminiai rezultatai naudojant 10000 atsitiktinų duomenų rinkinio reikšmių ir sverto įvertį aukštesnį nei 0.6 (rezultatai pavaizduoti **9** pav)



9 pav. Nuorodų analizės rezultatų supainiojimo matricos

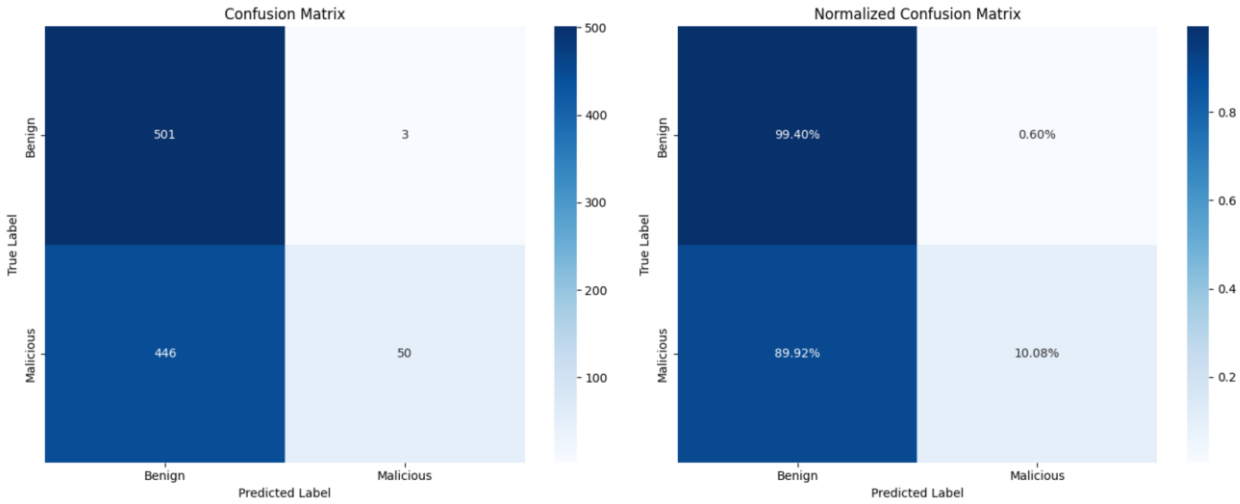
Modelis 0,77 tikslumu pažymi kenkėjiškas nuorodas kaip kenkėjiškas, tačiau taip pat turi aukštą neteisingai įvertintų nekenkėjiškų nuorodų, kurias pažymi kaip kenkėjiškas. Tačiau tai neįrodo, jog metodas veikia neteisingai. Tokį metodo veikimą galima apibūdinti kaip labai griežtą, kadangi nevalidūs SSL sertifikatas nuorodoje traktuojamas kaip stiprus taisyklės pažeidimas ir nuoroda žymima kaip kenksminga.

Didžiausia reikšmę kenkėjiškai pažymėtoms nuorodoms sudarė nevalidūs *SSL* sertifikatai, tai reiškia, nuorodos naudoja SSL, tačiau sertifikatai nevalidūs. To problema gali būti pasenęs duomenų rinkinys, ir sertifikatai yra pasibaigę galioti. Antroje vietoje naudojami *IP* adresai vietoj tikro domeno. Rezultatai pavaizduoti **10** pav.



10 pav. Kenksmingų nuorodų atpažinimo būdų sąrašas pagal kiekį

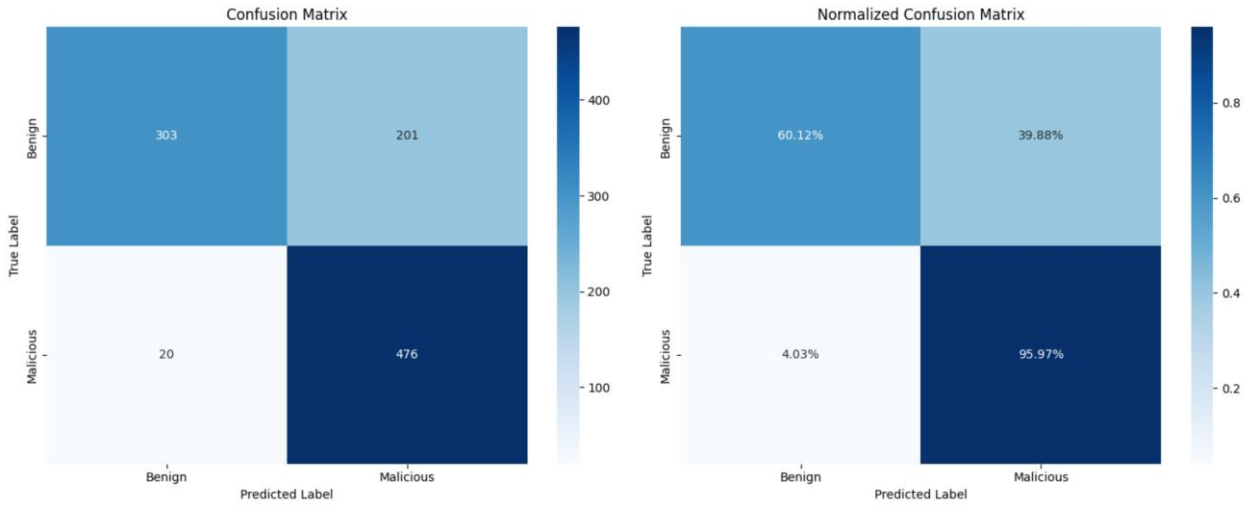
Bandymas išimti *SSL* patikrą iš metodo, tačiau gaunamas dar blogesnis įvertis. Rezultatas pavaizduotas **11** pav.



11 pav. Nuorodų analizės rezultatų supainiojimo matricos (be SSL sertifikato tikrinimo)

Tik 10.08% kenkėjiškų nuorodų pažymimos kaip kenkėjiškos, likusios įvertintos neteisingai. Tai rodo, jog metodas veikia visiškai neteisingai, jis nesugeba atpažinti kenksmingos nuorodos nuo nekenksmingos. Šis bandymas įrodo, jog *SSL* patikra yra būtina.

Norint gauti geriausias taisyklių reikšmes, bandoma keisti kiekvienos taisyklės įverčius ir sverto įvertį. Svertas didinamas nuo 0.2 iki 1 didinant įverčio svertą kas 0.05. Geriausias rastas svertas yra 0.2. Tai reiškia, jog nuorodai atitikus bent vieną taisyklę, ji žymima kaip kenksminga. Gautas aukščiausias tikslumas: 95.97. Rezultatas pavaizduotas **12 pav.**



12 pav. Galutinės nuorodų analizės rezultatų supainiojimo matricos

Rezultatas tik minimaliai geresnis už pirminį eksperimentą.

4.3.2. Mašininio mokymo treniravimo rezultatai keičiant treniravimo parametrus

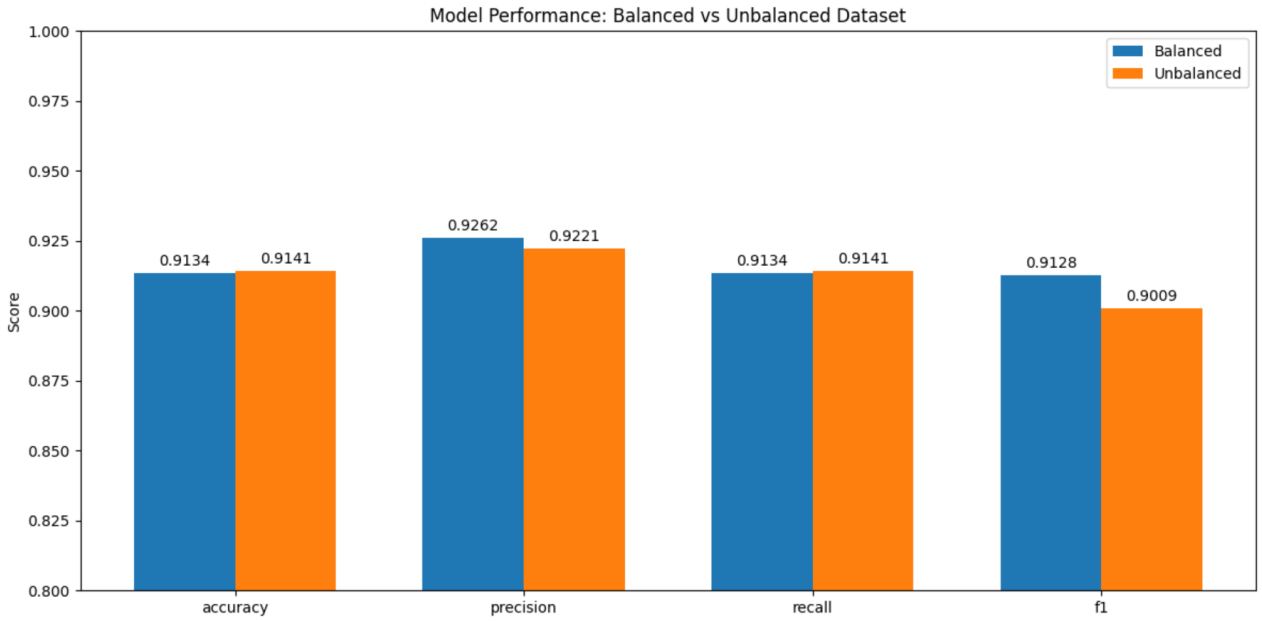
4.3.2.1. Balansuotas ir nebalansuotas duomenų rinkinys

Treniruojant mašininį mokymą buvo naudotas tas pats duomenų rinkinys, kuris buvo sujungtas ir sutvarkytas iš kelių skirtingų. Viso duomenų rinkinyje yra 6836 duomenų eilutės. 5719 nekenkėjiški ir 1117 kenkėjiški tekstai. Duomenų rinkinio pasiskirstymas: 70 % naudojama treniravimui, 15% testavimui, 15% validacijai. Naudojamų duomenų rinkinių inforamcija pateikta **4 lentelėje**.

4 lentelė Duomenų rinkinių palyginimas

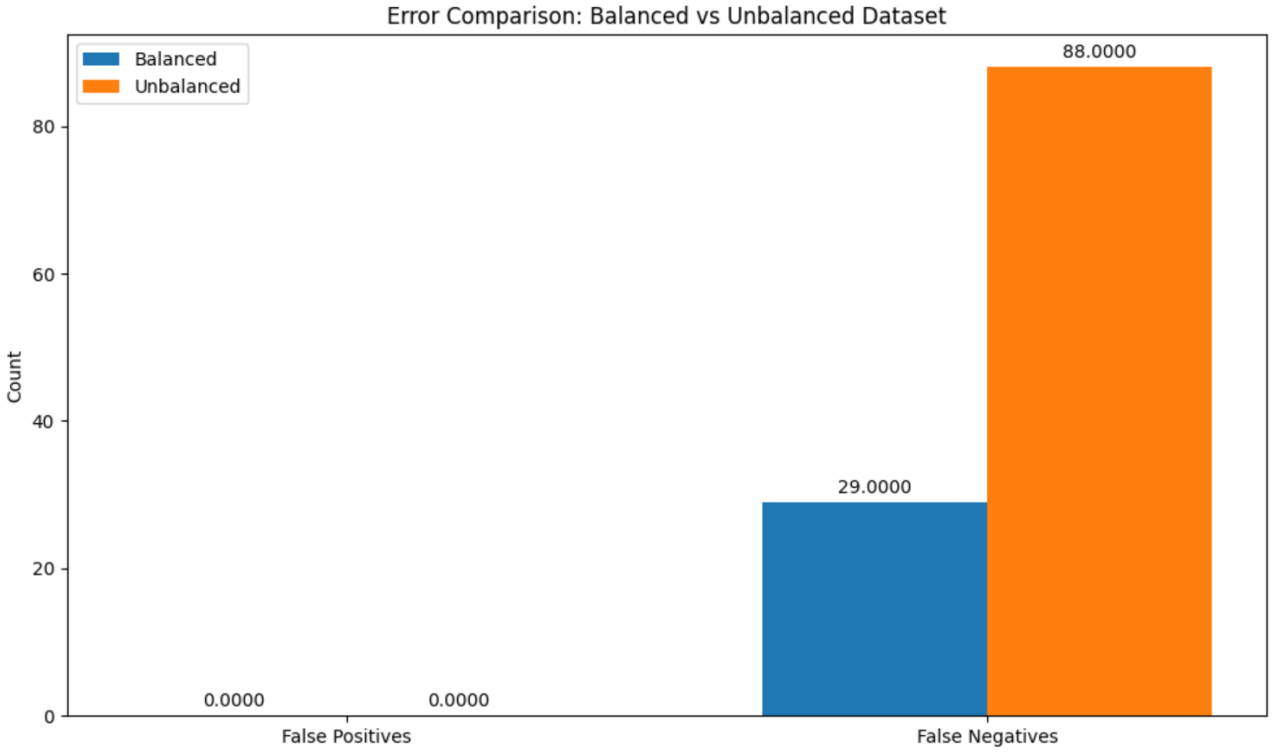
Tipas	**Viso duomenų**	**Kenkėjiški**	**Nekenkėjiški**
Nebalansuotas	6836	1117(16%)	5719(84%)
Balansuotas	2234	1117(50 %)	1117(50 %)

Treniravimo metu apskaičiuojamos metrikos lyginant balansuoto ir nebalansuoto duomenų rinkinio rezultatus. Rezultatas pavaizduotas **13** pav.



13 pav. Įverčių palyginimas naudojant balansuotą ir nebalansuotą duomenų rinkinį

Taip pat palyginamas klaidingai teigiamus ir klaidingai neigiamus rezultatus tarp balansuoto ir nebalansuoto duomenų rinkinio. Rezultatas pavaizduotas **14** pav.

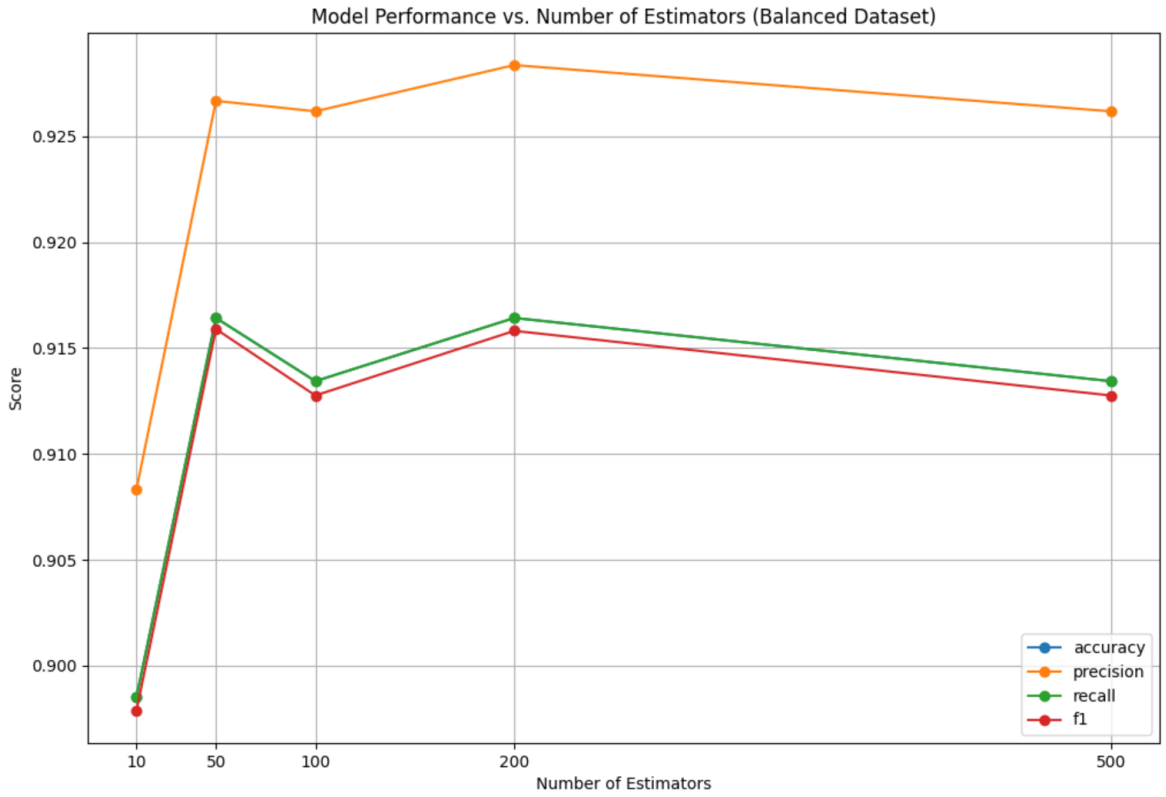


14 pav. Modelio įverčiai keičiant tarp balansuoto ir nebalansuoto duomenų rinkinio

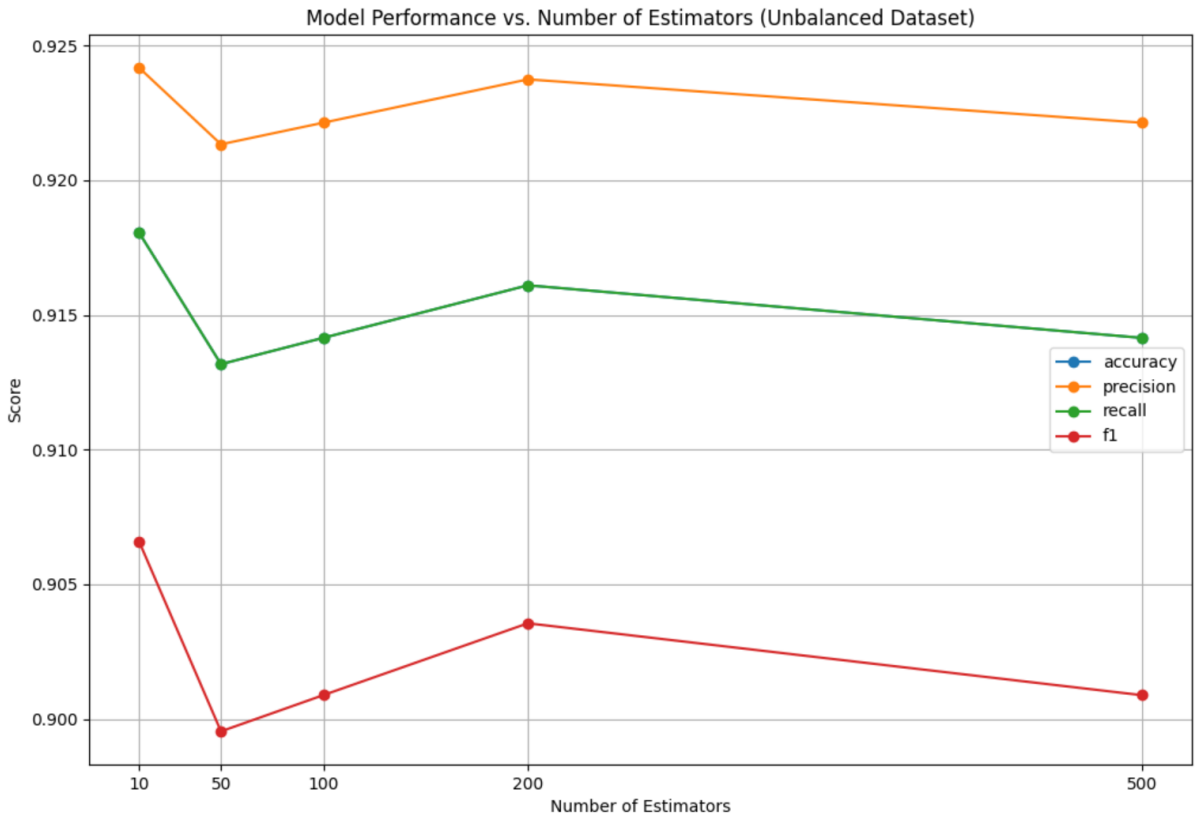
Atliktas atsitiktinio miško klasifikatoriaus, sukonfigūruoto su 200 šakų ir didžiausiu 20 medžio gyliu, vertinimas. Siekiant įvertinti jo veikimą sukčiavimo aptikimo sistemoje naudojant subalansuotus ir nesubalansuotus duomenų rinkinius. Eksperimento rezultatai rodo panašius bendrus tikslumo (subalansuotas - 0,9134, nesubalansuotas - 0,9141), tikslumo (subalansuotas - 0,9262,

nesubalansuotas - 0,9221) ir atšaukimo (angl. *recall*) (subalansuotas - 0,9134, nesubalansuotas - 0,9141) rodiklius. Tačiau pastebimas ryškus F1 balų ir klaidingai neigiamų rezultatų (angl *False-Negavtive, FN*) skaičiaus skirtumas. Modelis, išmokytas pagal subalansuotą duomenų rinkinį, pasiekė geresnį F1 balą (0,9128), palyginti su nesubalansuotu duomenų rinkiniu (0,9009), o tai rodo geresnę tikslumo ir atšaukimo pusiausvyrą. Labai svarbu, kad taikant subalansuotą metodą buvo gauta gerokai mažiau klaidingai neigiamų rezultatų ($FN = 29$) nei taikant nesubalansuotą metodą ($FN = 88$), o abiem atvejais buvo gauta nulinė klaidingai teigiamų rezultatų suma. Atsižvelgiant į tai, kad klaidingų neigiamų rezultatų minimizavimas yra labai svarbus tokiose saugumo programose kaip sukčiavimo aptikimas, reikšmingas klaidingų neigiamų rezultatu sumažėjimas aiškiai rodo didesnį praktinį modelio, parengto pagal subalansuotą duomenų rinkinį, naudingumą, nepaisant šiek tiek mažesnių tikslumo ir atšaukimo rodiklių, palyginti su nesubalansuotu analogišku modeliu.

4.3.2.2. Rezultatai keičiant šakų kiekį

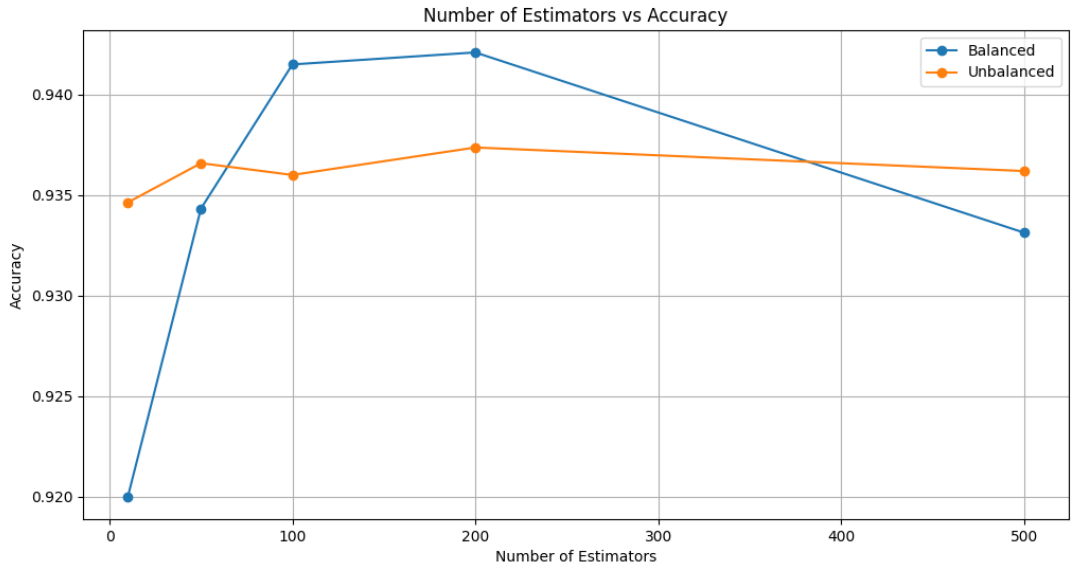


15 pav. Modelio įverčiai keičiant šakų skaičių (balansuotas duomenų rinkinys)

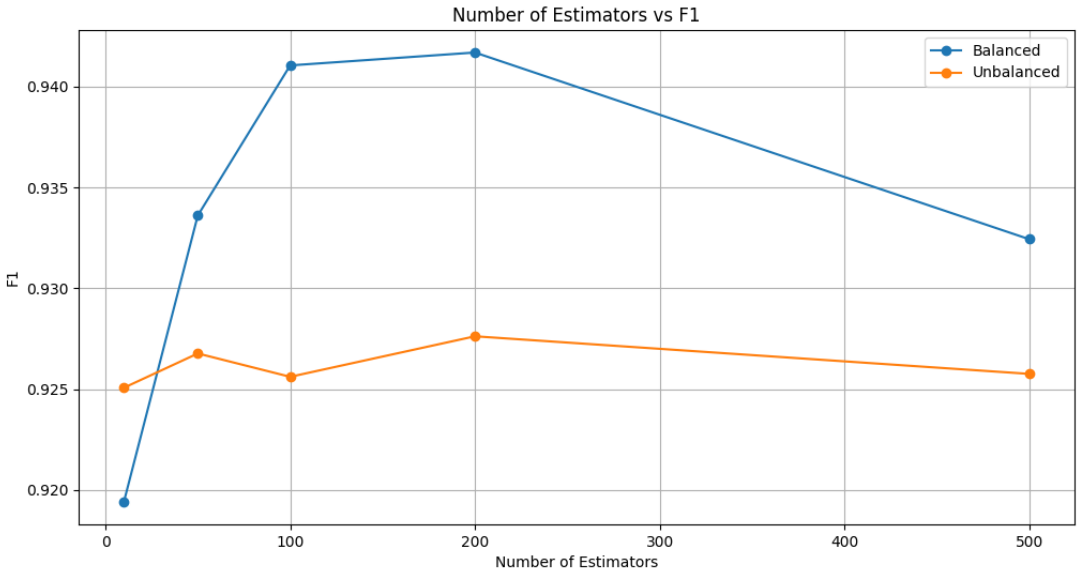


16 pav. Modelio įverčiai keičiant šakų skaičių

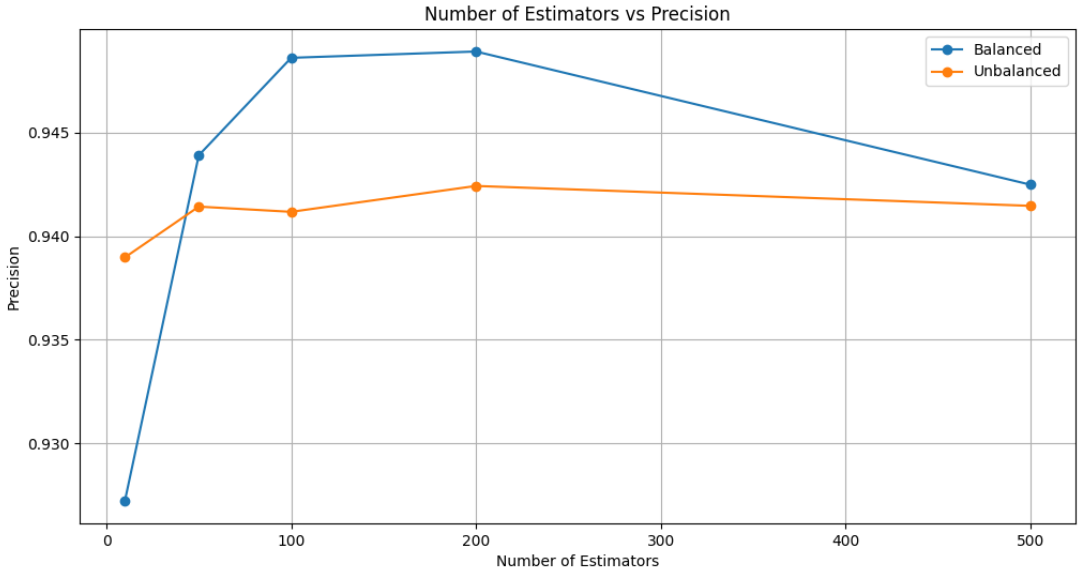
Tiriant šakų kiekio (išbandytos vertės: 10, 50, 100, 200, 500) įtaką, kai neribojamas maksimalus gylis, matyti, kad didinant medžių skaičių paprastai pagerėja našumas iki tam tikros ribos, kurią peržengus grąža sumažėja. Tiek subalansuotų, tiek nesubalansuotų duomenų rinkinių atveju, perėjus nuo 10 prie 50 ar 100 įverčių, paprastai pastebimai padidėja F1 balų skaičius ir sumažėja klaidingų neigiamų rezultatų. Tačiau padidinus įverčių skaičių iki 200 ar 500, našumo rodikliai dažnai pagerėja tik nežymiai arba net šiek tiek sumažėja. Pasirinkta šakų kiekį 200 sutampa su šia didžiausio arba beveik didžiausio našumo įvertinimu, kol pradedama pastebimai mažėti tikslumas.



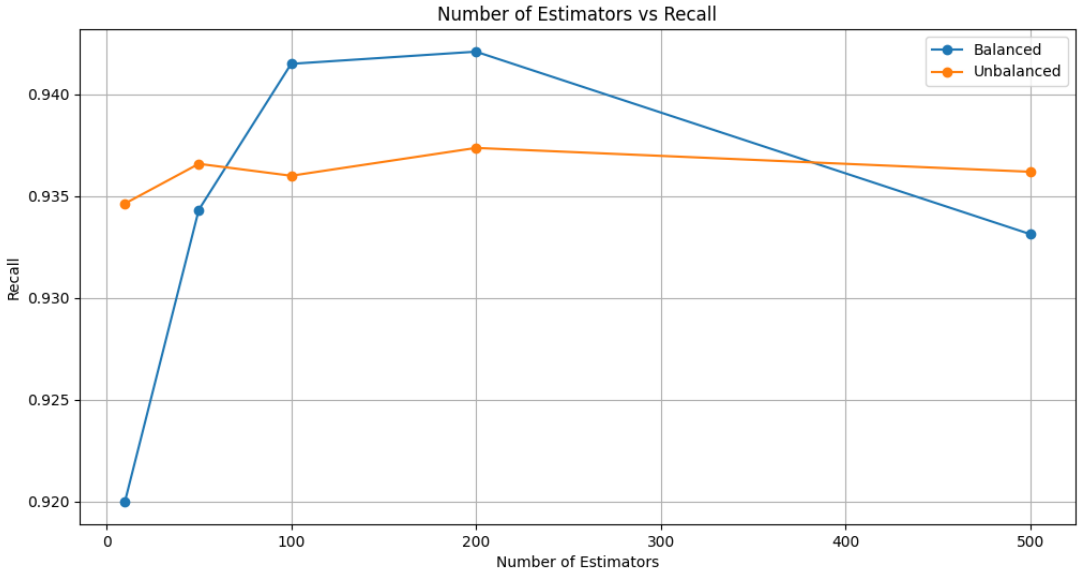
17 pav. Modelio įverčiai tarp balansuoto ir nebalansuoto duomenų rinkinio



18 pav. Modelio įverčiai tarp balansuoto ir nebalansuoto duomenų rinkinio



19 pav. Modelio įverčiai tarp balansuoto ir nebalansuoto duomenų rinkinio



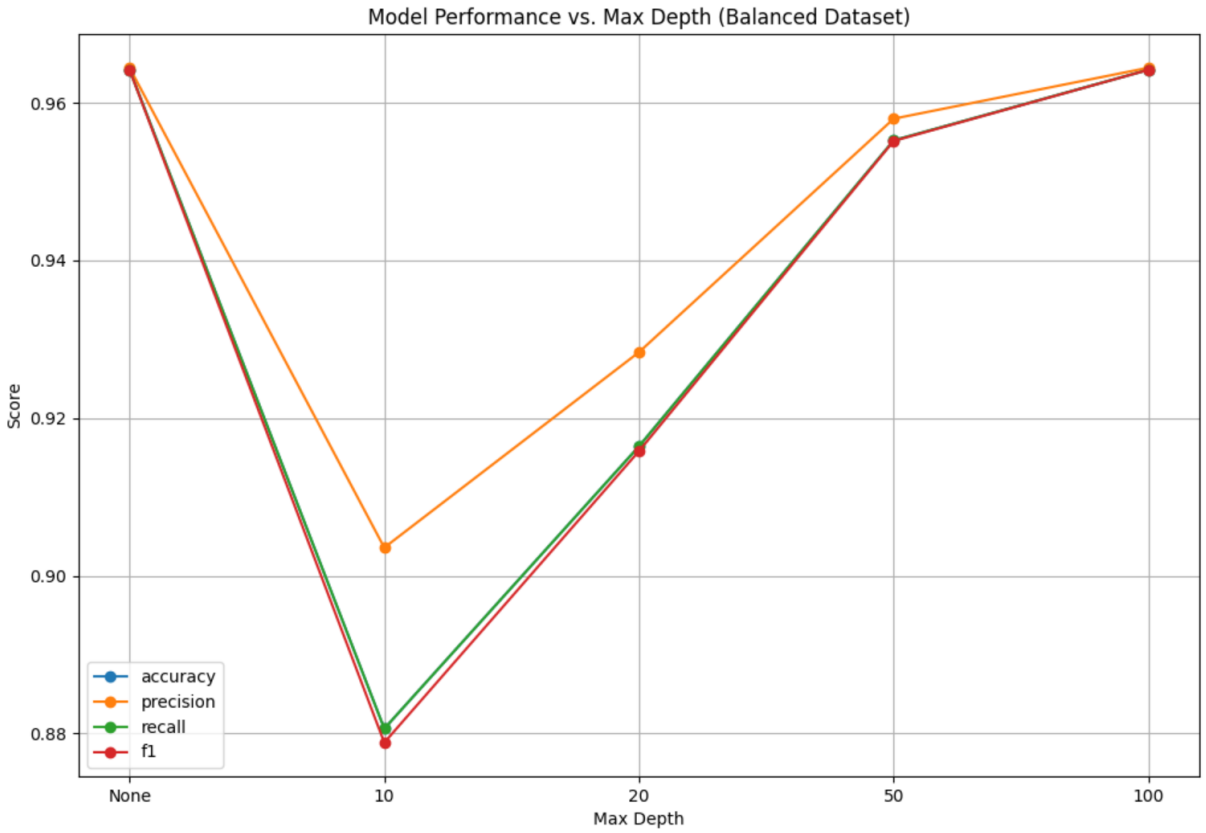
20 pav. Modelio įverčiai tarp balansuoto ir nebalansuoto duomenų rinkinio

Lyginant subalansuotą ir nesubalansuotą duomenų rinkinius pagal skirtingus šakų skaičius, subalansuotas duomenų rinkinys turi pranašumą visuose darytuose įverčių patikrinimuose , o tai yra labai svarbus veiksnys nustatant sukčiavimo atvejus. Nors optimalūs hiperparametrų nustatymai (pvz., medžio gylis nenurodytas šakų skaičius yra 200) leidžia pasiekti didelį tikslumą ir F1 balus abiejuose duomenų rinkiniuose.

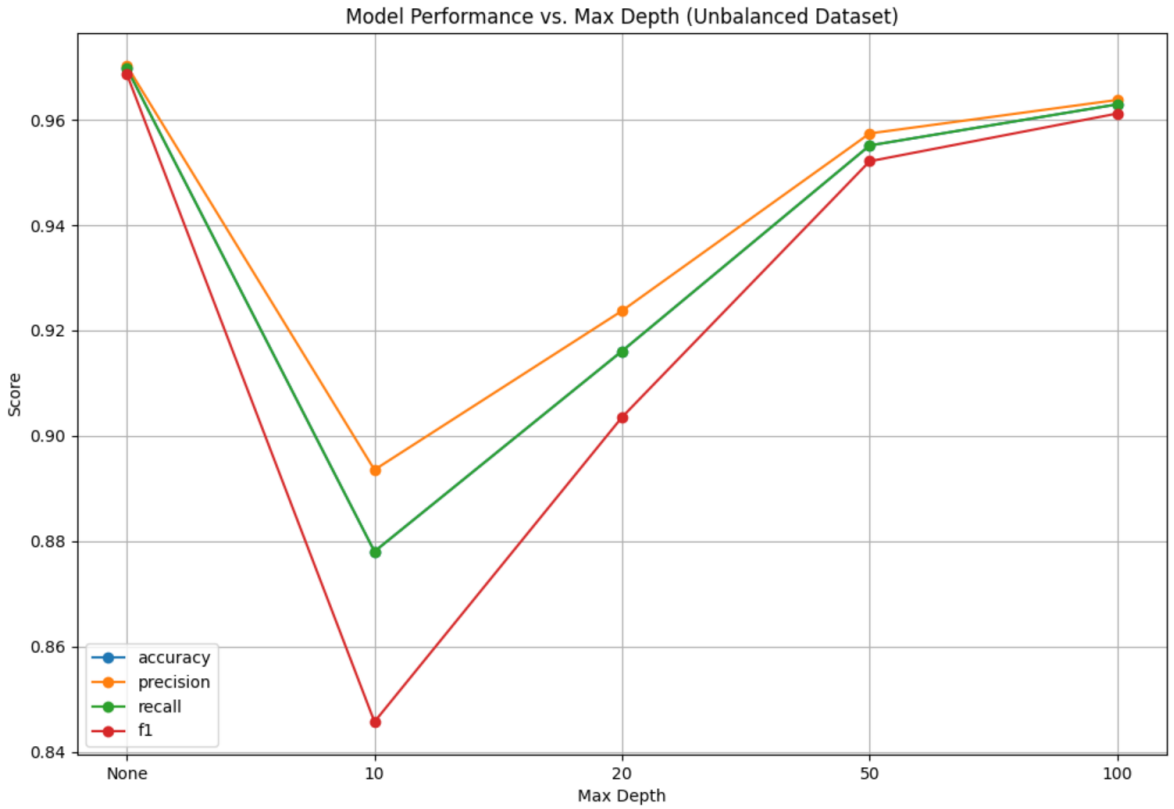
4.3.2.3. Rezultatai keičiant medžio gylį

Atsitiktinio miško pagrindas yra sprendimų medis - pagrindinis struktūrinis elementas, kuris rekursyviai suskirsto duomenis pagal požymių reikšmes, kad būtų galima prognozuoti. Hiperparametras medžio gylis atlieka lemiamą vaidmenį kontroliuojant kiekvieno atskiro miško sprendimų medžio sudėtingumą. Tiksliau, medžio gylis apibrėžia didžiausią medžio lygių arba mazgų, kuriuos jis gali turėti nuo šaknies iki lapo, skaičių. Šis parametras daro tiesioginę įtaką medžio gebėjimui išmokti sudėtingas mokymo duomenų detales ir ryšius.

Mažesnė medžio gylio reikšmė apriboja medžio skirstymų skaičių, todėl medžiai tampa mažesni. Tokie medžiai mažiau geba užfiksuoti sudėtingas požymių sąveikas ir gali lemti paprastesnį modelį. Priešingai, didesnė medžio gylio reikšmė leidžia medžiams augti giliau, todėl jie gali išmokti sudėtingesnius modelius ir galbūt labiau atitikti mokymo duomenis. Nors šis didesnis sudėtingumas gali būti naudingas norint užfiksuoti subtilius duomenų niuansus, jis taip pat padidina per didelio pritaikymo riziką, kai modelis išmoksta triukšmą ir specifines mokymo duomenų savybes, o ne pagrindinius apibendrintus modelius.



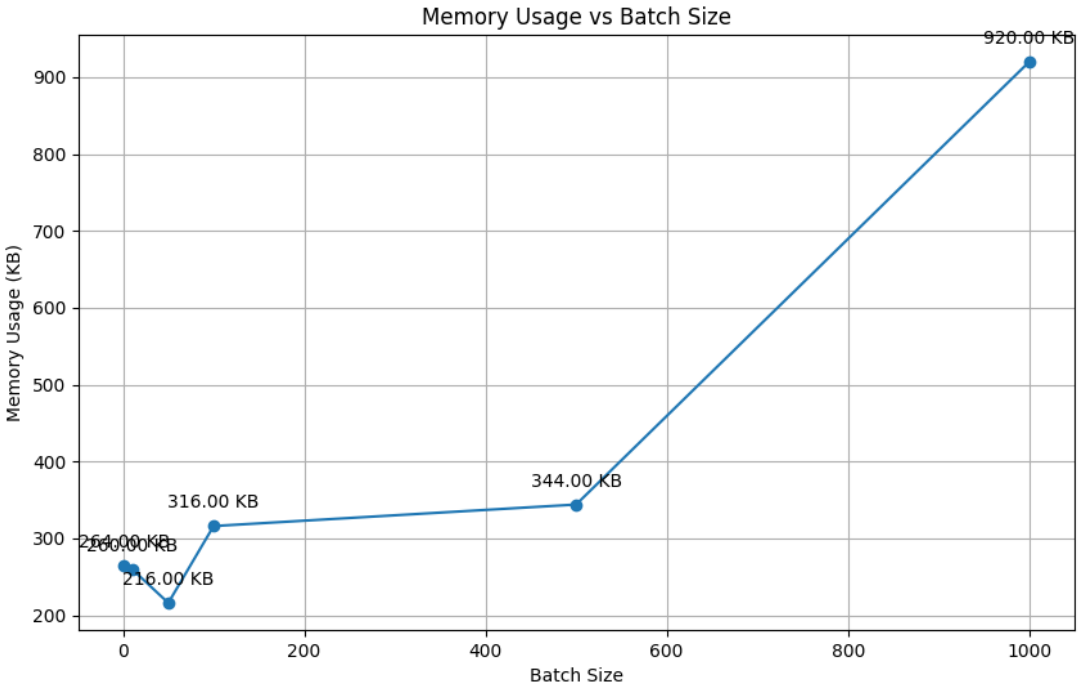
21 pav. Modelio įverčiai keičiant gylio parametrą (balansuotas duomenų rinkinys)



22 pav. Modelio įverčiai keičiant gylio parametrą (nebalansuotas duomenų rinkinys)

Analizė, apimanti atsitiktinio miško klasifikatoriaus hiperparametrų derinimą, atskleidžia skirtingą šakų ir medžio gilumo poveikį modelio veikimui tiek subalansuotiems, tiek nesubalansuotiems duomenų rinkiniams. Nagrinėjant gylio poveikį (išbandytos vertės: nenurodytas, 10, 20, 50, 100), kai šakų reikšmės yra pastovios (pvz., 100 arba 200), pastebima aiški tendencija: medžio gylio apribojimas labai pablogina našumą. Konkrečiai, dėl mažo medžių gylio, pavyzdžiui, 10 arba 20, nuolat mažėja, tikslumas, atšaukiamumas ir F1 rezultatai, palyginti su konfigūracijomis, leidžiančiomis naudoti gilesnius medžius (nenurodytas, 50 arba 100). Leidžiant medžiams augti be aiškaus gylio apribojimo arba nustačius aukštą ribą paprastai gaunami geriausi F1 rezultatai ir mažiausias klaidingų neigiamų rezultatų skaičius. Pradinis medžio gylis pasirinkimas atrodo neoptimalus, palyginti su leidimu augti giliau, remiantis šiuo duomenų rinkiniu.

4.3.3. Sprendimo resursų naudojimo rezultatai.



23 pav. Atminties sąnaudos keičiant užklausų kiekį

Peržiūrėjus duomenis (**23 pav.**), akivaizdu, kad su operacija susijęs atminties naudojimas yra netiesiškai susijęs su partijos dydžiu. Mažesnių partijų dydžių (1, 10 ir 100) atminties naudojimas išlieka santykinai mažas ir šiek tiek kintantis - nuo 200 kilobaitų Šį pradinį kintamumą esant mažesniems partijos dydžiams galima paaiškinti fiksuotomis atminties pridėtinėmis išlaidomis arba nedideliais sistemos išteklių paskirstymo svyravimais operacijos metu.

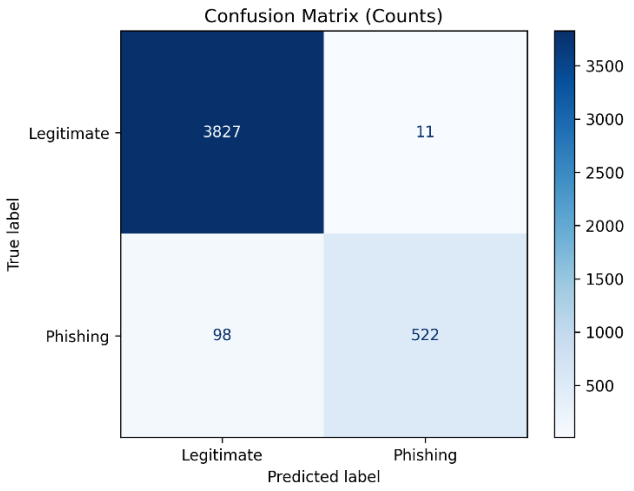
Didinant partijos dydį pastebimas žymus atminties naudojimo padidėjimas. Perėjus nuo 100 iki 500 partijų dydžio, atminties sąnaudos padidėja - iki 344 kilobaitų. Toliau didinant partijos dydį iki 1000, atminties naudojimas žymiai padidėja - iki 936 kilobaitų. Žymus atminties naudojimo padidėjimas esant didesniam partijos dydžiui, ypač nuo 500 iki 1000, rodo, kad operacija apdoroja duomenis dalimis, kurioms reikia daugiau atminties, nes didėja duomenų kiekis vienoje dalyje. Šis pastebėjimas labai svarbus optimizuojant programos našumą ir išteklių naudojimą, ypač kai susiduriama su atminties apribojimais arba apdorojami dideli duomenų kiekiai, rodantis, kad paketų dydžiai, viršijantys tam tikrą ribą, lemia didesnį sistemos atminties poreikį.

4.3.4. Galutinis mašininio mokymo rezultatas pakeitus treniravimo parametrus

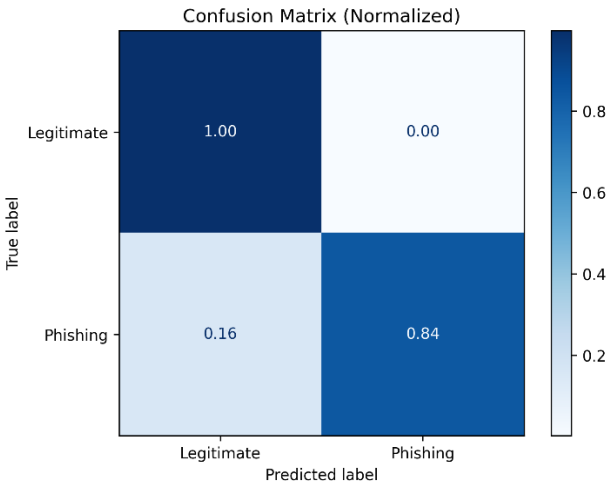
Padarius eksperimentus, buvo nustatyti tinkamiausi ir geriausi parametrai šiam tyrimui yra:

- Duomenų rinkinys: balansuotas;
- Iteracijų skaičius: 5;
- Šakų skaičius: 200;
- Medžio gylis: Nenurodyta (geriausias rezultatas gavus nenurodžius medžio gylio);

Supainiojimo matricos prieš:

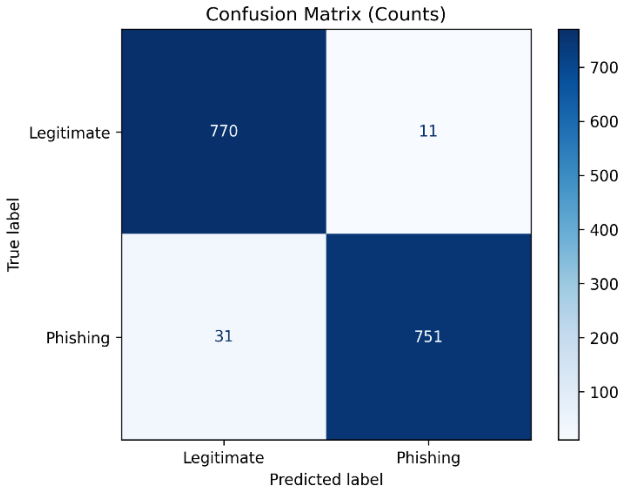


24 pav. Sumaišymo matrica

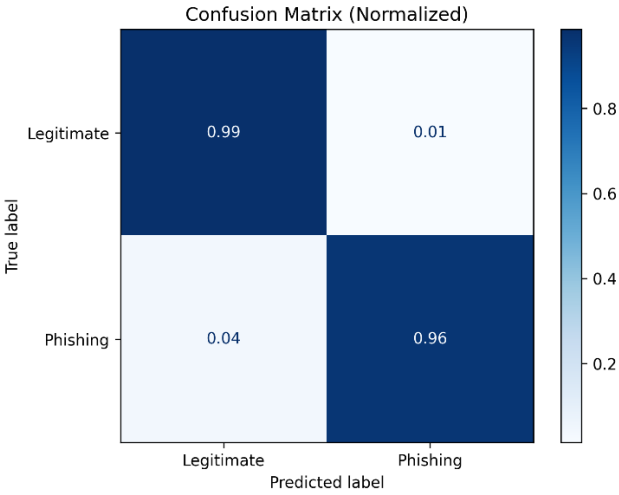


25 pav. Sumaišymo matrica (normalizuota)

Supainiojimo matricos po:



26 pav. Sumaišymo matrica



27 pav. Sumaišymo matrica (normalizuota)

Patobulinus treniravimo rezultatus, matomas ryškus modelio tikslumo pagerėjimas,

- Tikslumas: 0.9644
- Atsakymo įvertis (*angl. Recall*): 0.9642
- F1-įvertis: 0.9642

4.3.4.1. Reikšmingiausios žymos su didžiausiomis įvertimis

5 lentelė Reikšmingiausios žymos su įvertimis

Numeris	Reikšmė	Įvertis	Numeris	Reikšmė	Įvertis
1.	claim	0.035274	11.	text	0.013699
2.	free	0.034269	12.	reply	0.011640
3.	prize	0.023179	13.	service	0.011010
4.	txt	0.022866	14.	send	0.010729
5.	www	0.021926	15.	sms	0.010476
6.	account	0.021836	16.	150p	0.009806
7.	mobile	0.021227	17.	stop	0.009490
8.	50	0.017801	18.	chat	0.008798
9.	uk	0.016919	19.	new	0.008772
10.	urgent	0.014265	20.	won	0.008479

5 lentelėje pateikta informacija apie svarbiausias mokymo žymas (žodžius), kadangi modelis buvo treniruojamas naudojant anglišką duomenų rinkinį, reikšmės yra angliškos. Svarbiausios: *claim, free, prize*, jos turi didžiausius įverčius.

4.3.5. Metodo eksperimento ribotumas

Pasiūlytas metodas turi ribojimų. Visų pirma, mašininio mokymo modelis treniruotas anglų kalba, todėl jis neatpažįsta kitų kalbų teksto. Antra, duomenų rinkinio dydis yra ganėtinai mažas (2234 balansuoto duomenų rinkinio), todėl didesnė tikimybė, jog kompleksinio ar neįprasto teksto modelis neatpažins. Trečia, nuorodų analizė gali netinkamai išanalizuoti sutrumpintas nuorodas, kadangi, tai nėra tikrosios nuorodos į kurias norima nukreipti.

4.4. Eksperimento išvados

1. Atlikti nuorodų analizės testavimai, modelis patobulintas gavus tarpinius testavimo rezultatus;
2. Atlikti mašininio mokymo modelio testavimai treniruojant su skirtingais hiperparametrais, gautas didesnis modelio tikslumas keičiant parametrus;
3. Atlikti resursų sąnaudų testavimai. Apskaičiuota operatyviosios atminties sąnaudos analizuojant skirtingą kiekį žinučių;
4. Išskirta pagrindinės reikšmingiausios žymos apmokius mašininio mokymo modelį;
5. Aprašytas metodo eksperimento ribotumas.

Išvados

1. Išanalizuoti duomenų viliojimo atakų tipai, vieni dažniausiai pasikartojančių, yra duomenų viliojimas el. laiškais, SMS žinutėmis;
2. Išsiaiškinta, jog duomenų viliojimo atakų metu nutekinama jautri informacija, sukeliama turtinė ir neturtinė žala;
3. Nustatyti duomenų viliojimo atakų daugiavektoriniai požymiai - naudojamos skirtingos atakų platformos, atliekamos kelių žingsnių atakos;
4. Išanalizuoti esantys duomenų viliojimo atakų aptikimo sprendimai. Jie gali būti technologiniai, arba žmogiškieji, pavyzdžiui: sukuriant saugos politiką, rengiant mokymus;
5. Sukurtos taisyklės norint įvertinti nuorodų ir failų kenkėjiškumą. Metodas integruoja trečių šalių aplikacijas patikrai. Ištreniruotas ir panaudotas mašininio mokymo modelis naudojant "*Random Forest*" algoritmą. Sukurta formulė įvertinti bendrą žinutės kenkėjiškumą;
6. Įgyvendintas duomenų viliojimo metodas. Ištirtas nuorodų ir teksto kenkėjiškumo atpažinimas ir atlikti eksperimentiniai tyrimai. Mašininio mokymo modelio kenkėjiško teksto atpažinimo tikslumas 0,9644, nuorodų analizės tikslumas 0,77 remiantis duomenų rinkiniu;
7. Parengtas straipsnis "*A method for identifying and assessing phishing attacks in communication messages*" priimtas publikuoti žurnale „*International Journal of Computer (IJC)*" .

Literatūros sąrašas

1. Nuno Oliveira, Isabel Praça, Eva Maia, Orlando Sousa (2021) Intelligent Cyber Attack Detection and Classification for Network-Based Intrusion Detection Systems. https://www.mdpi.com/2076-3417/11/4/1674
2. Hongyu Liu and Bo Lang (2019) Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey. https://www.mdpi.com/2076-3417/9/20/4396
3. Hesham Altwaijry (2012) Bayesian Based Intrusion Detection System. https://link.springer.com/chapter/10.1007/978-94-007-4786-9_3
4. Noor Ammar Odeh, DErar Eleyan, Amna Eleyan (2021) A SURVEY OF SOCIAL ENGINEERING ATTACKS: DETECTION AND PREVENTION TOOLS. https://www.researchgate.net/profile/Derar-Eleyan/publication/355410947_A_SURVEY_OF_SOCIAL_ENGINEERING_ATTACKS_DETECTION_AND_PREVENTION_TOOLS/links/616f06cf3d9af67ad738e640/A-SURVEY-OF-SOCIAL-ENGINEERING-ATTACKS-DETECTION-AND-PREVENTION-TOOLS.pdf
5. Ashraf Abzakh, Ahmad Althunibat (2023) Human Factor and Cybersecurity https://ieeexplore.ieee.org/abstract/document/10225828
6. Mohammad Hijji, Gulzar Alam (2021) A Multivocal Literature Review on Growing Social Engineering Based Cyber-Attacks/Threats During the COVID-19 Pandemic: Challenges and Prospective Solutions https://ieeexplore.ieee.org/abstract/document/9312039
7. Wenni Syafitri, Zarina Shukur, Umi Asma' Mokhtar, Rossilawati Sulaiman Muhammad Azwan Ibrahim (2022) Social Engineering Attacks Prevention: A Systematic Literature Review. https://ieeexplore.ieee.org/abstract/document/9743471
8. Wenni Syafitri, Zarina Shukur, Umi Asma' Mokhtar, Rossilawati Sulaiman Muhammad Azwan Ibrahim (2022) Social Engineering Attacks Prevention: A Systematic Literature Review. https://ieeexplore.ieee.org/abstract/document/9743471
9. H. Sandouka, A. J. Cullen and I. Mann, "Social engineering detection using neural networks" (2009) *Proc. Int. Conf. CyberWorlds* http://ieeexplore.ieee.org/document/5279574/.
10. M. Lansley, F. Mouton, S. Kapetanakis and N. Polatidis, "SEADEr++: Social engineering attack detection in online environments using machine learning", (2020)
11. MF Ansari, PK Sharma, B Dash, Prevention of Phishing Attacks Using AI-Based Cybersecurity Awareness Training (March 2022) DOI:10.47893/IJSSAN.2022.1221 https://www.researchgate.net/publication/362112009_Prevention_of_Phishing_Attacks_Using_AI-Based_Cybersecurity_Awareness_Training
12. Anastasios Papathanasiou, George Liontos, Georgios Paparis, Vasiliki Liagkou and Euripides Glavas (2024) BEC Defender: QR Code-Based Methodology for Prevention of Business Email Compromise (BEC) Attacks https://doi.org/10.3390/s24051676. https://www.mdpi.com/1424-8220/24/5/1676
13. Wenni Syafitri, Zarina Shukur, Umi Asma' Mokhtar, Rossilawati Sulaiman Muhammad Azwan Ibrahim (2022) Social Engineering Attacks Prevention: A Systematic Literature Review. https://ieeexplore.ieee.org/abstract/document/9743471
14. Wenni Syafitri, Zarina Shukur, Umi Asma' Mokhtar, Rossilawati Sulaiman Muhammad Azwan Ibrahim (2022) Social Engineering Attacks Prevention: A Systematic Literature Review. https://ieeexplore.ieee.org/abstract/document/9743471

15. Vrbančič, G., Fister Jr, I., & Podgorelec, V. (2018). Swarm intelligence approaches for parameter setting of deep learning neural network: Case study on phishing websites classification. In *Proceedings of the 8th international conference on web intelligence, mining and semantics*
16. R. Geetha & T. Thilagam (2020) A Review on the Effectiveness of Machine Learning and Deep Learning Algorithms for Cyber Security. https://link.springer.com/article/10.1007/S11831-020-09478-2
17. Samer Atawneh and andHamzah Aljehani (2023) Phishing Email Detection Model Using Deep Learning. https://www.mdpi.com/2079-9292/12/20/4261
18. Asadullah Safi and Satwinder Singh (2023) A systematic literature review on phishing website detection techniques. https://www.sciencedirect.com/science/article/pii/S1319157823000034
19. M. Chathar Singh, P. Sumanth, S. Bala Sathyanarayana, G. Rithika (2022) Phishing email detection using deep learning algorithms. https://www.researchgate.net/publication/360895336_Phishing_email_detection_using_deep_learning_algorithms
20. Valentine Onih (2024) Phishing Detection Using Machine Learning: A Model Development and Integration https://www.researchgate.net/publication/379263290_PHISHING_DETECTION_USING_MACHINE_LEARNING-A_MODEL_DEVELOPMENT_AND_INTEGRATION
21. Padraig Cunningham, Sarah Jane Delany (2020) k-Nearest Neighbour Classifiers: 2nd Edition. https://arxiv.org/abs/2004.04523
22. Steven L. Salzberg (1993) C4.5: Programs for Machine Learning by J. Ross Quinlan. Morgan Kaufmann Publishers https://link.springer.com/article/10.1007/BF00993309
23. Badr HSSINA Author, Abdelkarim MERBOUHA, Hanane EZZIKOURI, Mohammed ERRITALI (2014) A comparative study of decision tree ID3 and C4.5, https://thesai.org/Publications/ViewPaper?Volume=4&Issue=2&Code=SpecialIssue&SerialNo=3
24. Rutvija Pandya, Jayati Pandya (2015) C5.0 Algorithm to Improved Decision Tree with Feature Selection and Reduced Error Pruning https://www.ijcaonline.org/archives/volume117/number16/20639-3318/
25. Email Spam or Not (Classification), version 1, 2024. Internet: https://www.kaggle.com/datasets/devildyno/email-spam-or-not-classification [2025]
26. SMS Spam Detection Dataset, version 1, 2024. Internet: https://www.kaggle.com/datasets/vishakhdapat/sms-spam-detection-dataset [2025]
27. SMS Spam Collection Dataset, version 1, 2016. URL: https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset/data [2025]
28. Spam SMS Classification Using NLP, version 1, 2024. Internet: https://www.kaggle.com/datasets/mariumfaheem666/spam-sms-classification-using-nlp [2025]