



Kauno technologijos universitetas

Informatikos fakultetas

Tinklapių filtravimo sistema, skirta maršrutizatoriams

Baigiamasis magistro projektas

Dominykas Baranauskis

Projekto autorius

prof. Agnius Liutkevičius

Vadovas

Kaunas, 2024



Kauno technologijos universitetas

Informatikos fakultetas

Tinklapių filtravimo sistema, skirta maršrutizatoriams

Baigiamasis magistro projektas

Informacijos ir informacinių technologijų sauga (6211BX008)

Dominykas Baranauskis

Projekto autorius

prof. Agnius Liutkevičius

Vadovas

doc. Nerijus Morkevičius

Recenzentas

Kaunas, 2024



Kauno technologijos universitetas

Informatikos fakultetas

Dominykas Baranauskis

Tinklapių filtravimo sistema, skirta maršrutizatoriams

Akademinio sąžiningumo deklaracija

Patvirtinu, kad:

1. baigiamąjį projektą parengiau savarankiškai ir sąžiningai, nepažeisdama(s) kitų asmenų autorius 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 sąrašė;
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.

Dominykas Baranauskis

Patvirtinta elektroniniu būdu

Baranauskis, Dominykas. Tinklapių filtravimo sistema, skirta maršrutizatoriams. Magistro studijų baigiamasis projektas / vadovas prof. Agnius Liutkevičius; Kauno technologijos universitetas, informatikos fakultetas.

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

Reikšminiai žodžiai: tinklapių filtravimas, mašininis mokymas, sistema maršrutizatoriams.

Kaunas, 2024. 67 p.

Santrauka

Nors internetas ir laikomas vienu iš didžiausių technologinių šuolių sukėlusiu šiuolaikinių išradimų, tačiau vis dažniau dėmesys atkreipiamas ir į neigiamą jo pusę – priegą prie nepilnamečiams bei kitiems naudotojams žalingo ar kito nepageidaujamo turinio. Šio darbo metu buvo sukurta mašininio mokymo ir skirstymo į kategorijas paremta tinklapių filtravimo sistema, skirta maršrutizatoriams. Tinklapių skirstymo į kategorijas procesas vykdomas centriniame serveryje, o filtravimas – maršrutizatoriuje, naudojant DNS blokavimą. Siekiant paspartinti užklausų apdorojimą maršrutizatoriuje, sumažinti naudojamus resursus, buvo panaudotas podėliavimas, o norint užkirsti kelią kuriamos tinklapių filtravimo sistemos turinio blokavimo išvengimui, sukurtos ugniasienės konfigūravimo taisyklės. Tyrimų metu buvo nustatyta modelio apmokymui naudojamų duomenų rinkinio kokybės įtaka sistemos veikimo efektyvumui, įvertinta sukurtos sistemos ar jos dalies diegimo įterptinėje aplinkoje galimybė, ištirtas teksto apdorojimui naudojamų mašininio mokymosi algoritmų tinklapių skirstymo į kategorijas efektyvumas, taip pat įvertintas sistemoje naudojamų mašininio mokymosi algoritmų ir jų derinių tinklapių skirstymo į kategorijas tikslumas ir greitaveika.

Baranauskis, Dominykas. Website Filtering System for Routers. Master's Final Degree Project / supervisor prof. Agnius Liutkevičius; Faculty of Informatics, Kaunas University of Technology.

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

Keywords: website filtering, machine learning, system for routers.

Kaunas, 2024. 67 pages.

Summary

Although the internet is considered as one of the most significant technological advancements, attention is being drawn to its negative aspects, such as access to harmful or unwanted content by minors and other users. During this study, a machine learning-based website filtering system for routers was developed. The process of categorizing websites is performed on a central server, while filtering is done on the router using DNS blocking. To expedite query processing on the router and reduce resource usage, caching was employed. Additionally, firewall configuration rules were created to prevent circumvention of the website filtering system's content blocking. Research was conducted to assess the impact of the quality of the training data set on the system's performance, evaluate the feasibility of deploying the developed system or its components in an embedded environment, examine the effectiveness of machine learning algorithms used for text processing in categorizing websites, and assess the accuracy and performance of the machine learning algorithms and their combinations used in the website categorization system.

Turinys

Lentelių sąrašas	7
Paveikslų sąrašas	8
Santrumpų ir terminų sąrašas	10
Įvadas	11
1. Literatūros apžvalga	12
1.1. Internetinio turinio skirstymas ir filtravimo sistemų poreikis	12
1.2. Informacijos filtravimas	13
1.2.1. Internetinio turinio filtravimo taškai	13
1.2.2. Sąrašų sudarymas	15
1.2.3. Natūralios kalbos apdorojimas	16
1.2.4. Mašininio mokymosi taikymas.....	17
1.2.5. Internetinio turinio filtravimo metodai	20
1.3. Apibendrinimas	22
2. Tinklapių filtravimo sistemos projektas	23
2.1. Sprendimo koncepcija	23
2.2. Keliami reikalavimai	24
2.3. Sistemos modelis	24
2.4. Apibendrinimas	31
3. Tinklapių filtravimo sistemos realizacija	32
3.1. Prototipo realizacijos vizija	32
3.2. Tinklapių skirstymo į kategorijas algoritmas ir filtravimo sistemos modelių apmokymas.....	33
3.2.1. URL adreso apdorojimas	33
3.2.2. Metaduomenų apdorojimas	34
3.2.3. Tinklapių teksto apdorojimas	35
3.3. Tinklapių kategorijos svorio apskaičiavimas	37
3.4. Blokavimo paketo veikimas maršrutizatoriuje	38
3.5. Centrinis serveris	40
3.6. Podėliavimas.....	41
3.7. Ugniasienės konfigūracija	41
3.8. Apibendrinimas	42
4. Tinklapių filtravimo sistemos, skirtos maršrutizatoriams, tyrimas	43
4.1. Tyrimo aplinka	43
4.2. Duomenų rinkinio kokybės vertinimas	43
4.3. Sistemos diegimo įterptinėje sistemoje galimybių vertinimas	51
4.4. Teksto apdorojimui naudojamų mašininio mokymosi algoritmų efektyvumo tyrimas.....	53
4.5. Sistemoje naudojamų mašininio mokymosi algoritmų bei jų derinių veikimo tikslumo ir greita veikos tyrimas.....	56
4.6. Patobulinta tinklapių skirstymo į kategorijas sistema	58
4.7. Tyrimo išvados	60
Išvados	61
Literatūros sąrašas	63

Lentelių sąrašas

1 lentelė. Internetinio turinio filtravimo taškų palyginimas.....	15
2 lentelė. Centralizuotas ir decentralizuotas sistemų palyginimas.....	16
3 lentelė. Algoritmų palyginimas.....	20
4 lentelė. Internetinio turinio filtravimo metodų palyginimas	22
5 lentelė. Kategorijos dažniausiai pasikartojančių žodžių indeksavimo pavyzdys.....	38
6 lentelė. Sistemos realizacijai naudoto serverio parametrai	43
7 lentelė. Duomenų rinkinyje Nr. 1 esančios tinklapių kategorijos	44
8 lentelė. Duomenų rinkinyje Nr. 2 esančios tinklapių kategorijos	48
9 lentelė. Bandymui naudoto maršrutizatoriaus parametrai.....	52
10 lentelė. Tinklapių priskyrimo kategorijai trukmė maršrutizatoriuje ir centriniame serveryje	52
11 lentelė. Teksto apdorojimui naudojamų mašininio mokymosi algoritmų efektyvumo parametrai.....	56
12 lentelė. Kuriamoje sistemoje naudojamų algoritmų veikimo tikslumas ir vidutinė tinklapių apdorojimo trukmė	57
13 lentelė. Tinklapių turinio ir metaduomenų apdorojimo logistinė regresija efektyvumas	59

Paveikslų sąrašas

1 pav.	Internetinio turinio filtravimo taškai	13
2 pav.	Teksto skirstymo į atskirus žodžius ir žodžių fragmentus algoritmo vykdymo etapai	17
3 pav.	<i>Bag of Words</i> (BoW) metodo vykdymo etapai	17
4 pav.	Sprendimų medžio ir atsitiktinio miško veikimo schemos	18
5 pav.	Atraminių vektorių klasifikatoriaus veikimo principas.....	19
6 pav.	Tinklapių filtravimo sistemos koncepcija	23
7 pav.	Dažniausiai pasikartojančių žodžių sąrašo generavimo veiklos diagrama.....	25
8 pav.	Tinklapių skirstymo į kategorijas sistemos algoritmo veiklos diagrama	26
9 pav.	Tinklapių HTML kodo apdorojimo veiklos diagrama	27
10 pav.	Tinklapių priskyrimo tam tikrai kategorijai iš URL adreso veiklos diagrama.....	28
11 pav.	Tinklapių priskyrimo tam tikrai kategorijai iš metaduomenų veiklos diagrama.....	29
12 pav.	Sistemos pirminio konfigūravimo veiklos diagrama	30
13 pav.	Sistemos realizacijos vizija	32
14 pav.	Sukurtas dažniausiai pasikartojančių žodžių sąrašas iš URL adresų	33
15 pav.	Tinklapių kategorijos nustatymas naudojant URL adresą.....	34
16 pav.	Sukurtas dažniausiai pasikartojančių žodžių sąrašas iš metaduomenų	34
17 pav.	Tinklapių kategorijos nustatymas naudojant metaduomenis.....	35
18 pav.	Tinklapių kategorijos nustatymas naudojant tinklapių turinį:.....	36
19 pav.	Tinklapių kategorijos svorio apskaičiavimo algoritmas.....	37
20 pav.	Tinklapių https://www.crazygames.com/ kategorijos nustatymas iš metaduomenų	38
21 pav.	Maršrutizatoriuje įdiegto blokavimo paketo veikimo schema	39
22 pav.	Sugeneruotas konfigūracinis failas, numatantis blokuojamas kategorijas	40
23 pav.	Kategorijos atvaizdavimas apdorojus GET užklausą.....	40
24 pav.	Podėliavimo procesas	41
25 pav.	DNS užklausų nukreipimas į sukonfigūruotą DNS serverį.....	42
26 pav.	VPN blokavimo taisyklė	42
27 pav.	Painiavos matrica, sudaryta naudojant duomenų rinkinį Nr. 1, iš kurio pašalinti tik neaktyvūs tinklapiai	45
28 pav.	Painiavos matrica, sudaryta naudojant duomenų rinkinį Nr. 1, iš kurio pašalinti neaktyvūs tinklapiai ir tinklapiai, kurių pasikliovimo koeficientas buvo mažesnis nei 0,6	46
29 pav.	Painiavos matrica, sudaryta naudojant duomenų rinkinį Nr. 1, iš kurio pašalinti neaktyvūs tinklapiai ir tinklapiai, kurių pasikliovimo koeficientas buvo mažesnis nei 0,8	47
30 pav.	Painiavos matrica, sudaryta naudojant duomenų rinkinį Nr. 1 su tinklapiais, kurių pasikliovimo koeficientas lygus 1	48
31 pav.	Painiavos matrica, sudaryta naudojant duomenų rinkinį Nr. 2	49
32 pav.	Duomenų rinkinio Nr. 2 įrašų pasiskirstymas tarp kategorijų	50
33 pav.	Duomenų rinkinio Nr. 2 įrašų pasiskirstymas po rankinio koregavimo	50
34 pav.	Painiavos matrica, sudaryta naudojant duomenų rinkinį Nr. 2, po rankinio koregavimo	51
35 pav.	Tinklapių priskyrimo kategorijai trukmės priklausomybė nuo siunčiamų užklausų kiekio	53
36 pav.	Painiavos matrica, sudaryta tekstinę informaciją apdorojant logistine regresija	54
37 pav.	Painiavos matrica, sudaryta tekstinę informaciją apdorojant Naiviojo Bajeso algoritmu ...	54
38 pav.	Painiavos matrica, sudaryta tekstinę informaciją apdorojant SVM algoritmu.....	55
39 pav.	Painiavos matrica, sudaryta tekstinę informaciją apdorojant atsitiktinio miško algoritmu .	55

40 pav. Tinklapių priskyrimo kategorijai trukmės priklausomybė nuo naudojamų mašininio mokymosi algoritmų bei jų derinių	57
41 pav. Patobulinto tinklapių skirstymo į kategorijas sistemos algoritmo veiklos diagrama	58
42 pav. Painsiavos matrica, sudaryta metaduomenis ir tekstinę informaciją apdorojant logistine regresija	59

Santrumpų ir terminų sąrašas

Terminai:

DNS (angl. *Domain Name System*) – srities vardų struktūra, naudojama internete, pakeičiant skaitinius IP adresus.

HTML (angl. *HyperText Markup Language*) – tai kompiuterinė žymėjimo kalba, naudojama pateikti turinį internete.

IP (angl. *Internet Protocol*) – unikalus numeris, priskiriamas kiekvienam kompiuteriui ar kitam įrenginiui, kuriuo yra prisijungiama prie interneto.

LAN (angl. *Local Area Network*) – tai vietinis tinklas, apimantis nedidelę geografinę vietovę, pavyzdžiui, pastatą.

URL (angl. *Uniform Resource Locator*) – tai unikalus tinklapio adresas, kurį sudaro domeno pavadinimas, kategorija ir kartais kiti papildomi elementai.

Įvadas

Statistiniai duomenys atskleidžia, jog 2022 metais interneto naudotojų skaičius apėmė daugiau nei 60 % visų pasaulio gyventojų ir siekė šiek tiek daugiau nei 5,1 mlrd. [1]. Akivaizdu, kad internetas laikomas vienu iš technologinių perversmą sukėlusiu šiuolaikinių išradimų, kuris gerina žmonių gyvenimo kokybę, tačiau vis dažniau dėmesys atkreipiamas ir į neigiamą jo pusę – sukeliama priklausomybė, įvairaus pobūdžio kibernetinius nusikaltimus, pornografiją, smurtą, nepageidaujamą reklamą.

Atlikto tyrimo rezultatai rodo, kad lyginant 2011 m. ir 2021 m., naudojimosi internetu trukmė per parą vienam žmogui išaugo daugiau nei 2,5 karto – nuo 75 iki 192 min. [2]. Siekiant nuo žalingo ar kito nepageidaujamo turinio apsaugoti nepilnamečius vaikus ar nevalingai nuklydusius naudotojus, gali būti taikomas interneto resursų ribojimas arba kitaip, tinklapių filtravimas. Šiuo metu dažniausiai yra naudojamas filtravimas galutiniame vartotojo įrenginyje, tačiau tai sukelia sunkumų, kai tam tikri tinklapiai turėtų būti blokuojami keliuose įrenginiuose vienu metu. Svarbu paminėti ir tai, jog filtravimo sistemų veikimas paprastai yra paremtas iš anksto sudarytu sąrašu, kuris ilgainiui dėl nebeaktyvių ir naujų sukurtų tinklapių tampa nebeaktualus.

Šio darbo tikslas – sukurti maršrutizatoriams skirtą tinklapių filtravimo sistemą, kurios veikimas būtų pagrįstas skirstymu į kategorijas.

Uždaviniai:

1. išanalizuoti skaitmeninio turinio filtravimo aktualumą ir jo veikimo principus;
2. parinkti kuriamai sistemai tinkamiausią filtravimo metodą (-us);
3. realizuoti pasiūlytą filtravimo sistemos sprendimą;
4. ištirti sukurtos tinklapių filtravimo sistemos veikimą ir jos efektyvumą.

Šiame darbe bus atliekama skaitmeninio turinio filtravimo aktualumo ir poreikio analizė, apžvelgiami informacijos filtravimo ypatumai. Antrame darbo skyriuje aprašomas tinklapių filtravimo sistemos, skirtos maršrutizatoriams, projektas – apibrėžiami funkciniai bei nefunkciniai reikalavimai, pateikiamos pagrindinių sistemos procesų veiklos diagramos. Trečiasis skyrius apima sukurtos sistemos realizaciją, o ketvirtasis skirtas sistemos testavimui, eksperimentams ir tyrimams.

1. Literatūros apžvalga

Šiame skyriuje apžvelgiami pagrindiniai dokumentai, numatantys internete draudžiamos ir ribojamos informacijos skelbimą, internetinio turinio filtravimo poreikis. Taip pat analizuojami internete esančio turinio filtravimo taškai, dažniausiai taikomi metodai, algoritmai, sąrašų sudarymo aspektai.

1.1. Internetinio turinio skirstymas ir filtravimo sistemų poreikis

Šiuo metu pasaulyje yra beveik 1,3 mlrd. internetinių tinklapių [3]. Statistika atskleidžia, jog maždaug trečdalis žiniatinklyje pateikiamos informacijos yra susijusi su suaugusiems skirtu erotinio pobūdžio turiniu [4], o 7 % visų tinklapių sudaro pornografinis turinys [5]. Plečiant skaitmeninių paslaugų, o tuo pačiu ir interneto ryšio prieinamumą, išauga poreikis užkirsti kelią nelegalaus ir neigiamą poveikį tam tikriems asmenims turinčio turinio sklaidai. Dėl šios priežasties įvairūs įstatymai ir reglamentai, įpareigojantys gerbti pagrindines žmogaus teises bei laisves, yra perkeliama ir į elektroninę erdvę.

Remiantis Lietuvos Respublikos Ryšių reguliavimo tarnybos duomenimis, neigiamas interneto turinys gali būti skirstomas į ribojamą ir draudžiamą [6]. Tam tikra informacija, daranti neigiamą įtaką fiziniam ir protiniam asmenų vystymuisi, gali būti ribojama siekiant apsaugoti nepilnamečius asmenis. Šiam tipui priskiriama:

- informacija, susijusi su fizinio ar psichinio smurto vaizdavimu, nusikalstamos veikos modeliavimu;
- erotinio pobūdžio informacija;
- informacija, kurioje rodomas mirusio arba žiauriai sužaloto žmogaus kūnas;
- informacija, sukelianti baimę ar siaubą, skatinanti savęs žalojimą ar savižudybę.

Informacija, kurios paviešinimas ir (ar) platinimas yra draudžiamas pagal galiojančius įstatymus apima pornografinio turinio informaciją, taip pat informaciją, kuria tyčiojama, niekinama, skatinama neapykanta ar kurstoma diskriminuoti žmonių grupę ar jai priklausančią asmenį dėl lyties, seksualinės orientacijos, rasės, tautybės, kalbos, kilmės, socialinės padėties, tikėjimo ar pažiūrų [6].

LR nepilnamečių apsaugos nuo neigiamo viešosios informacijos poveikio įstatyme rašoma, jog tokio tipo informacijai yra priskiriamas smurtinio pobūdžio, baimę ar siaubą sukkeliantis, nepagarbą gyvybei kuriantis ar erotinio pobūdžio turinys, stambiu planu rodomas miręs arba sunkiai sužalotas žmogus, demonstruojami inscenizuoti paranormalūs reiškiniai, sudarant šių reiškių tikrumo įspūdį, taip pat siūlantį dalyvauti azartiniuose lošimuose ir kituose žaidimuose, kuriuose sudaromas lengvo laimėjimo įspūdis, informacija. Neigiamą įtaką nepilnamečiams darančia laikoma ir tokia informacija, kuria palankiai vertinama nusikalstama veika, priklausomybė nuo narkotinių ir psichotropinių medžiagų, tabako, alkoholio, skatinamas žmogaus orumą žeminantis elgesys, taip pat vartojami nešvankūs žodžiai ar nepadorūs gestai [7]. Vadovaujantis šiuo įstatyme aprašoma tvarka, minėto pobūdžio internetinis turinys turėtų būti ribojamas įvairiose vietose, kuriose prieigą prie interneto gali turėti nepilnamečiai asmenys, pavyzdžiui, ugdymo įstaigose, bibliotekose.

LR azartinių lošimų įstatyme numatyta, jog bendrojo ugdymo mokyklų, ikimokyklinio, neformaliojo ugdymo ir kitų vaikų auklėjimo, taip pat sveikatos priežiūros įstaigose, valstybės ir savivaldybių institucijose privaloma blokuoti galimybes dalyvauti nuotoliniuose lošimuose [8].

Taigi, remiantis apžvelgtais šaltiniais, ribojamą ir draudžiamą informaciją būtų galima suskirstyti į penkias pagrindines grupes:

- pornografija ir kitas erotinio pobūdžio turinys;
- smurtas bei ginklai;
- azartiniai lošimai ir kiti panašaus pobūdžio žaidimai;
- alkoholis, tabakas, narkotinės ir psichotropinės medžiagos;
- neapykantą, smurtą ar diskriminaciją kurstanti kalba.

Internetinio turinio filtravimas, tai tam tikra programinė įranga ar kitų priemonių grupė, naudojama galutiniam vartotojui pateikiamam internetiniam turiniui kontroliuoti – tam tikriems tinklapiams filtruoti ir/ar blokuoti. Žinoma, ši priemonė gali būti naudojama ne tik siekiant apsisaugoti nuo nelegalaus ir neigiamą poveikį asmenims turinčio internetinio turinio, bet ir dėl kitų priežasčių, pavyzdžiui, autorių teisių pažeidimų [9].

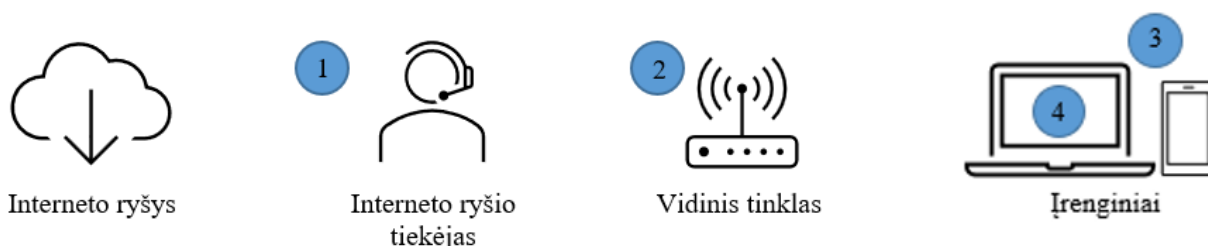
Nors didelė dalis pasaulio šalių stengiasi teisiškai riboti prieigą prie neigiamą įtaką nepilnamečiams ar nevalingai nuklydusiems naudotojams darančios informacijos, tam tikro internetinio turinio blokavimo ar filtravimo sistemų kūrimo bei jų efektyvumo didinimo poreikis išlieka itin didelis. Atliktas mokslinis tyrimas taip pat patvirtina internetinių tinklapių filtravimo sistemų poreikį tiek privačiose įmonėse, tiek ir valstybinėse įstaigose [10]. Apklausa rodo, kad asmenys savo darbo vietose dažnai naršo internete asmeninėms reikmėms, pavyzdžiui, apsiperka elektroninėse parduotuvėse, naudoja asmeninį el. paštą, socialinius tinklus, užsiima internetiniais lošimais ir žaidimais, siunčiasi muziką ar filmus. Dėl šios priežasties įmonių vadovams tenka susidurti su autorių teisių pažeidimais bei nelegalių programinės įrangos versijų naudojimu įmonės kompiuteriuose, konfidencialių duomenų nutekiniu, kibernetinėmis atakomis, virusais, darbuotojų motyvacijos, produktyvumo ir interneto tinklo pralaidumo sumažėjimu.

1.2. Informacijos filtravimas

Priklausomai nuo filtruojamos informacijos tipo, sistemai keliamų reikalavimų bei pageidaujamo jos tikslumo, internete aptinkamo turinio filtravimas gali būti paremtas įvairaus sudėtingumo algoritmais – nuo elementaraus informacijos palyginimo su sąrašuose esančiais duomenimis iki dirbtinio intelekto naudojimo.

1.2.1. Internetinio turinio filtravimo taškai

Atsižvelgiant į vartotojų poreikius, naudojamo interneto tinklo sudėtingumą, saugumo reikalavimus, turinio filtravimas gali būti atliekamas keliuose taškuose – vietiniame tinkle, galutinio vartotojo įrenginyje, naršyklėje ar paieškos sistemoje, taip pat turinio filtrai būna diegiami ir interneto ryšio tiekėjų (žr. 1 pav.).



1 pav. Internetinio turinio filtravimo taškai

Internetinio turinio filtravimą, atsižvelgiant į organizacijos ar vyriausybės nustatytas taisykles, gali vykdyti interneto ryšio tiekėjas (angl. *ISP-level filtration*) (žr. 1 pav. 1). Tokiu atveju yra reikalingi itin galingi serveriai, galintys apdoroti daugybę užklausų vienu metu. Interneto ryšio tiekėjo vykdomas turinio filtravimas dažniausiai nepriklauso nuo vartotojo lūkesčių, o modifikavimo procesai užtrunka gana ilgai. Tokiu būdu Lietuvoje neprieinamas tapo tinklapis *linkomanija.net* – teismo sprendimu buvo uždrausta interneto ryšio tiekėjams savo klientams teikti prieigą prie šio tinklapio [11, 12].

Internetinio turinio filtravimo sistema įdiegta pagrindiniame tinkle ar serveryje (angl. *server-side filtration*) (žr. 1 pav. 2) yra itin naudinga didelėse organizacijose, nes užtenka sukurti vieną turinio filtravimo taisyklių rinkinį ir taikyti jį visiems tinklo vartotojams. Kadangi ribojamas internetinis turinys įvairiose organizacijose ir namų ūkiuose, ar netgi skirtingose tos pačios įmonės dalyse gali skirtis, priklausomai nuo įvairių aspektų, pavyzdžiui, interneto naudotojų amžiaus ar vidinių įstaigos taisyklių, kurios draudžia tam tikrus tinklapius, todėl patogu tai, jog serveryje esančiuose turinio filtruose labai lengva priskirti skirtingus prieigos lygius tam tikroms vartotojų klasėms. Paprastai šio tipo filtravimo sistemai priklauso daug įrenginių, todėl yra reikalingas gana didelio našumo serveris ar įranga. Svarbu paminėti ir tai, jog filtravimas tinkle yra laikomas vienu efektyviausių ir sunkiausiai apeinamų [10, 11]. Turinio filtravimui taip pat gali būti naudojami ir maršrutizatoriai. Tokiu būdu vietiniame tinkle esančius įrenginius galima nukreipti į filtravimo sistemą nenaudojant papildomos įrangos [13].

Turinys filtruojamas galutinio vartotojo įrenginyje (angl. *client-side filtration*) (žr. 1 pav. 3) išsiskiria bene trumpiausia užklausų apdorojimo trukme. Toks turinio ribojimas dažniausiai sukuriamas apribojant vartotojo prieigą prie tam tikro turinio naudojant administratoriaus teises. Tokio tipo filtravimas dažnai naudojamas privačiuose namų ūkiuose ir įmonėse, kuriose siekiama filtravimo sistemą naudoti tam tikruose konkrečiuose įrenginiuose, tačiau įrenginių skaičiui išaugus, šio tipo filtravimas tampa nebeefektyvus ir itin imlus laikui dėl kiekvienam įrenginiui reikiamos individualios konfigūracijos bei priežiūros [10, 11]. „Net Nanny®“, tai tėvų kontrolės programinė įranga, diegiama galutinio vartotojo įrenginyje, kurios veikimas yra pagrįstas lankomų interneto tinklapių turinio analizavimu realiuoju laiku. Naudojamus filtrus vartotojas gali modifikuoti ir pasirinkti tam tikras turinio kategorijas, kurios turėtų būti blokuojamos. Sistemoje yra trys parinktys, pagal kurias turinys bus leidžiamas, blokuojamas arba peržiūrėjus tam tikrą turinį, valdytojui bus išsiunčiamas įspėjimas apie informacijos peržiūrą [14].

Turinio filtrai taip pat gali būti naudojami ir tiesiogiai naršyklėse ar paieškos sistemose (žr. 1 pav. 4), kai yra blokuojami iš anksto numatyti URL adresai ar turinys, susijęs su pasirinktais raktiniais žodžiais. Tokiam blokavimui yra reikalinga tam tikra programinė įranga, o šią filtravimo sistemą apeiti paprastai yra pakankamai nesudėtinga [11]. „BlockSite“, tai žiniatinklio filtras, paremtas URL filtravimu, veikiantis tokiose naršyklėse kaip „Chrome“, „Firefox“ ir „Edge“. Šios programos naudotojas gali blokuoti internetinį turinį pagal pasirinktas kategorijas ar raktinius žodžius [15].

1 lentelėje pateikiami apibendrinti internetinio turinio filtravimo taškų palyginimo duomenys.

1 lentelė. Internetinio turinio filtravimo taškų palyginimas [10, 11, 13]

Filtravimo taškas Savybė	Interneto ryšio tiekiąjas	Filtravimas tinkle	Galutinio vartotojo įrenginys	Naršyklė ar paieškos sistema
Papildomos aparatinės įrangos poreikis	Nebūtinas	Nebūtinas	Nėra	Nėra
Galimybė filtravimo sistemai priskirti daug įrenginių	Yra	Yra	Nėra	Nėra
Skirtingų prieigos lygių išskyrimo galimybė	Yra, bet sudėtinga	Yra	Nėra	Nėra
Filtravimo parametrų modifikavimas	Galimas, bet sudėtingas	Galimas	Galimas	Galimas
Sistemos patikimumas	Vidutinis/žemas	Aukštas	Vidutinis	Žemas

Daugiau nei 90 % 2022 m. vykdytos apklausos respondentų teigė, jog tarp jų naudojamų, prie interneto ryšio prijungtų įrenginių, yra išmanusis telefonas, 60 % minėjo kompiuterį, 32 % – televizorių, 28 % – planšetinį kompiuterį, 15 % – išmaniųjų namų prietaisą [16]. Šie duomenys rodo gana didelę įrenginių, kuriais buvo prisijungiama prie interneto, sklaidą, todėl renkantis internetinio turinio filtravimo tašką, būtina į šį aspektą atsižvelgti ir tokiu būdu užtikrinti sistemos efektyvumą.

Atlikus internetinio turinio filtravimo taškų apžvalgą ir palyginimą buvo nustatyta, jog geriausiomis savybėmis pasižymi filtravimas vykdomas vidiniame tinkle – nėra būtina papildoma aparatinė įranga, sistemai galima priskirti ne tik vieną įrenginį, ji yra gana patikima ir sąlyginai nesudėtingai modifikuojama, taip pat pasižymi galimybe sukurti kelis prieigos lygius.

1.2.2. Sąrašų sudarymas

Juodųjų sąrašų (angl. *blacklisting*) sudarymas yra vienas seniausių kompiuterių saugos algoritmų, kuris blokuoja nepageidaujamą informaciją. Šie sąrašai sudaromi į juos įtraukiant tam tikrus raktinius žodžius, tinklapių adresus ar kitą informaciją, kuri bus naudojama internetinio turinio rūšiavimui [17].

Baltieji sąrašai (angl. *whitelisting*) yra priešingi juodiesiems – tai patikimų tinklapių ar prieigą prie kito leistino turinio įgalinantis sąrašas. Tokių sąrašų sudarymas laikomas patikimesniu, tačiau, daugeliu atvejų, į juoduosius sąrašus reikia įtraukti daug mažiau informacijos, taigi, taip yra paprasčiau ir sistemos veikimas būna spartesnis [17].

Nors internetinio turinio filtravimas, grįstas tinklapių ar žodžių, frazių sąrašų sudarymu, yra vienas dažniausiai naudojamų, tačiau ši technologija paprastai yra paremta rankiniu sąrašuose esančios informacijos atnaujinimu ir papildymu. Svarbu paminėti ir tai, jog naudojant juoduosius tinklapių sąrašus yra blokuojami tik tam tikri konkretūs jų adresai, tačiau jų domenų pavadinimai gali būti pakeičiami, tokiu būdu išvengiant prieigos prie turinio blokavimo. Dėl šių priežasčių filtravimo tikslumas iš esmės priklauso nuo sistemos valdytojo sudaromų sąrašų kokybės bei pačios sistemos našumo, tačiau būtų galima šiuos sąrašus sudaryti ir centralizuotai.

Buvo sudarytas centralizuotos ir decentralizuotos sistemų palyginimas (žr. 2 lentelę). Lentelėje esanti informacija paremta centralizuotų ir decentralizuotų duomenų bazių savybėmis.

2 lentelė. Centralizuotas ir decentralizuotas sistemų palyginimas [18]

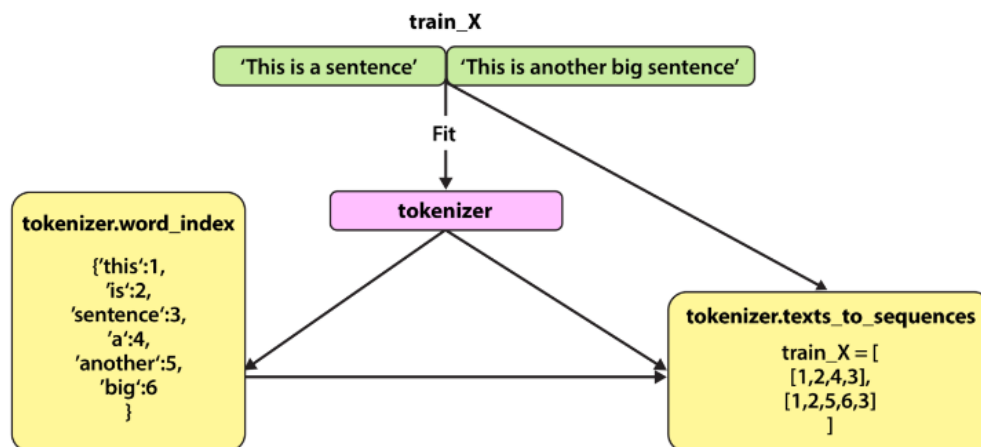
Sąrašo sudarymo būdas Savybė	Centralizuotas	Decentralizuotas
Užklausių apdorojimo sparta	Priklauso nuo centrinio serverio pajėgumų bei sistemos vartotojų skaičiaus	Priklauso nuo įrenginio, kurioje veikia sistema, pajėgumų bei sistemos vartotojų skaičiaus
Sąrašo kontrolė	Centrinio serverio valdytojas	Visi sistemos vartotojai
Sistemos veikimo sutrikdymo tikimybė įvykus įsilaužimui	Didelė	Maža
Sąrašo prieinamumas	Gali būti nepasiekiamas	Nuolat
Sistemos įdiegimo sudėtingumas	Mažiau sudėtingas	Sudėtingas

Atliktas palyginimas atskleidė, jog centralizuotai veikiančios sistemos yra paprastesnės ir sąlyginai lengviau realizuojamos nei decentralizuotos, tačiau decentralizuotai sistemai būdingas didesnis saugumas, mažesnė sistemos veikimo sutrikdymo galimybė įvykus įsilaužimui. Decentralizuotoje sistemoje yra tolygiau paskirstomi resursai, tačiau užklausių apdorojimo sparta ir bendras sistemos veikimas tiesiogiai priklauso nuo įrenginių pajėgumo.

1.2.3. Natūralios kalbos apdorojimas

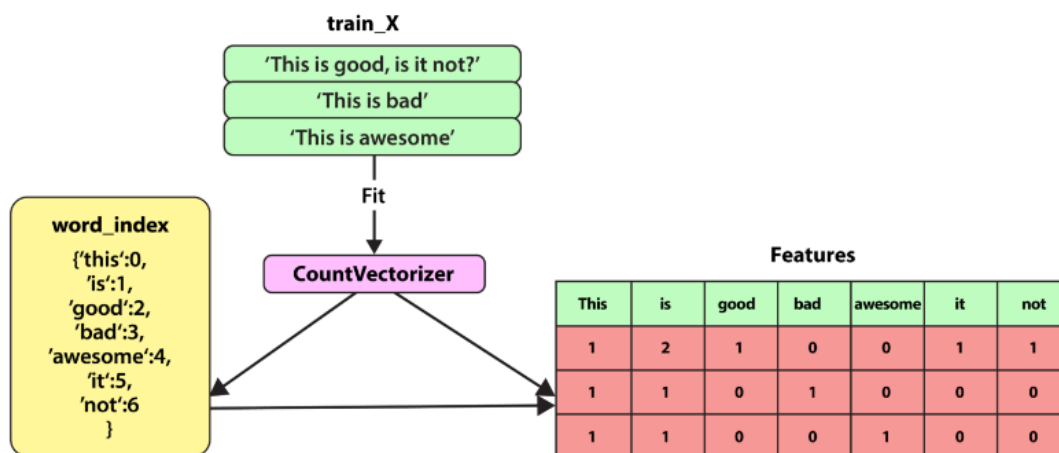
Natūralios kalbos apdorojimas yra kompiuterių mokslo ir dirbtinio intelekto šaka, apimanti kompiuterinių sistemų gebėjimą suprasti tekstą ar išstartus žodžius taip pat, kaip ir žmonės. Yra išskiriami du pagrindiniai natūralios kalbos apdorojimo etapai – duomenų pradinis apdorojimas (angl. *Data preprocessing*) ir ypatybių išskyrimas (angl. *Feature extraction*). Iš pradžių turimą tekstą reikia apdoroti, kad žodžiai ir simboliai būtų paversti modeliui suprantamu formatu [19, 20]:

- Skaidymas į sakinius (angl. *Sentence segmentation*). Šio etapo metu tekstas yra suskaidomas į sakinius. Tuo atveju, kai sakinio pabaiga yra žymima tašku, tai nėra sudėtinga, tačiau iškyla sunkumų, kai taškai aptinkami santrumpų pabaigoje arba jų apskritai nėra (pavyzdžiui, senovės kinų kalboje).
- Ne esminių žodžių pašalinimas (angl. *Stop words removal*). Šis etapas skirtas dažnai pasikartojančių, jungiamųjų ir kitų nereikšmingų žodžių, tokių kaip, pavyzdžiui, „the“, „an“, „ir“, „bet“, pašalinimui.
- Teksto skirstymas. Šio etapo metu tekstas suskirstomas į atskirus žodžius ir žodžių fragmentus (angl. *Tokenization*) (žr. 2 pav.).



2 pav. Teksto skirstymo į atskirus žodžius ir žodžių fragmentus algoritmo vykdymo etapai [19]

Bene dažniausiai naudojamas ypatybių išskyrimo metodas yra *Bag of Words* (BoW). Naudojant šį metodą yra skaičiuojama, kiek kartų kiekvienas žodis arba žodžių junginys pasikartoja tekste (žr. 3 pav.).



3 pav. *Bag of Words* (BoW) metodo vykdymo etapai [19]

1.2.4. Mašininio mokymosi taikymas

Mašininis mokymasis, tai dirbtinio intelekto sričiai priskiriamas pogrupis, apimantis tam tikrus algoritmus ir matematinius modelius, naudojamus kompiuterinėms sistemoms skirtų užduočių sprendimui. Šie algoritmai gali būti taikomi įvairiose srityse nuo kibernetinių atakų, sukčiavimo atvejų atpažinimo, elektroninių laiškų ar kitos tekstu pateiktos informacijos filtravimo iki gamybos procesų gerinimo, ankstyvųjų simptomų pažinimo, ligų prognozavimo [21-23].

Yra išskiriami trys mašininio mokymosi tipai [24-26]:

- prižiūrimas (angl. *Supervised Machine Learning*) – naudojami iš anksto surinkti tam tikri duomenys, kuriuose yra teisingi atsakymai į sistemai priskirtą užduotį. Remiantis jais, sistema mokosi panašiai atsakyti į naujus atvejus;
- neprižiūrimas (angl. *Unsupervised Machine Learning*) – naudojami įvairūs duomenys, be pirminio grupavimo ar skirstymo į teisingus ir neteisingus;

- sustiprintas (angl. *Reinforcement Machine Learning*) – mokymasis pagrįstas gaunamu atgaliniu ryšiu (atlygiu arba bauda).

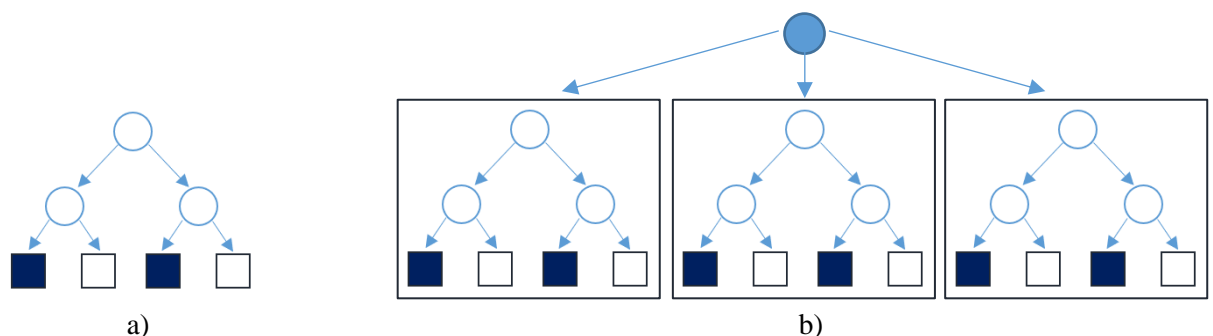
Bene dažniausiai yra taikomas prižiūrimas mašininio mokymo metodas. Nors šiuo principu pagrįstos sistemos veikimo kokybė ir tikslumas tiesiogiai priklauso nuo apmokymo duomenų kokybės, tačiau kitaip nei neprižiūrimo mokymo atveju, nėra reikalingi tokie galingi įrankiai ir dideli resursai. Didelio duomenų kiekio apdorojimas gali tapti iššūkiu naudojant prižiūrimą mokymąsi, tačiau gauti rezultatai pasižymi gana dideliu tikslumu ir yra patikimi. Nors neprižiūrimas mokymasis gali apdoroti didelius duomenų kiekius realiuoju laiku, tačiau kyla didesnė klaidingų rezultatų tikimybė [27].

Prižiūrimasis mokymasis daugiausia naudojamas sprendžiant dviejų tipų – klasifikavimo ir regresijos – užduotis. Regresijos algoritmai pasitelkiami siekiant rasti ryšį tarp tam tikrų kintamųjų, taip pat taikoma prognozavimui. Naudojant klasifikavimo algoritmus duomenys yra skirstomi į konkrečias kategorijas. Klasifikavimo uždaviniams spręsti dažniausiai naudojami prižiūrimo mokymosi algoritmai yra sprendimų medis (angl. *Decision Tree*) ir atsitiktinio miško metodas (angl. *Random Forest classifier*), logistinė regresija (angl. *Logistic Regression*), atraminių vektorių klasifikatorius (angl. *Support Vector Machine classifier*), naivusis Bajeso klasifikatorius (angl. *Naive Bayes classifier*) [28].

1.2.4.1. Sprendimų medis ir atsitiktinio miško metodas

Sprendimų medis (žr. 4 pav. a), tai vienas iš seniausių mašininio mokymosi algoritmų. Ši schema prasideda pagrindiniu (šakniniu), baigiasi lapų (rezultatų) mazgais. Šakas sudarantys tarpiniai mazgai, kuriuose talpinamos tam tikros duomenims atskirti reikalingos ypatybės, paprastai yra sudaryti iš kelių lygių. Priklausomai nuo tam tikrame mazge gauto rezultato, klasifikavimo algoritmas išsišakoja link sekančių mazgų, kur procesas kartojasi tol, kol yra pasiekiamas lapų mazgas.

Atsitiktinio miško metodas apima kelių sprendimų medžių naudojimą vienu metu lygiagrečiai (žr. 4 pav. b). Atsitiktinį mišką sudarantys sprendimų medžiai yra apmokomi naudojant skirtingus mokymo duomenis. Tų pačių įvesties duomenų klasifikavimo procesas vyksta kiekviename medyje, o galutiniu sprendimu tampa daugiausia kartų pasikartojęs rezultatas [29, 30].

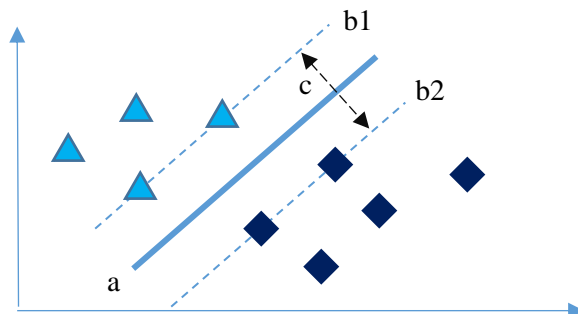


4 pav. Sprendimų medžio ir atsitiktinio miško veikimo schemas

Nors sprendimų medžio metodas yra paprastesnis ir paprasčiau pritaikomas, tačiau naudojant šį algoritmą dažnai sukurtas modelis per daug prisitaiko prie mokymosi duomenų aibės (angl. *overfitting*), todėl itin sumažėja jo veikimo tikslumas. Taip pat, kitaip nei taikant atsitiktinio miško algoritmą, sprendimų medžių pagrįstam modeliui didesnę įtaką daro įvairūs nuokrypiai ir ribinės vertės [30].

1.2.4.2. Atraminų vektorių klasifikatorius

Atraminų vektorių klasifikatoriaus arba kitaip SVM, veikimas paremtas ribos – kreivės ar tiesės – dažnai vadinamos hiperplokštuma (žr. 5 pav. a), kuria n-matėje erdvėje išsidėstę duomenys yra padalijami į dvi grupes, nustatymu [31, 32].



5 pav. Atraminų vektorių klasifikatoriaus veikimo principas

Siekiant rasti optimalią hiperplokštumą, per skirtingose klasėse esančius taškus braižomi atraminiai vektoriai (žr. 5 pav. b1 ir b2). Atstumas tarp hiperplokštumos ir atraminų vektorių (žr. 5 pav. c) turi būti maksimalus – tokiu būdu minimizuojant duomenų klasifikavimo klaidos tikimybę [31, 32].

1.2.4.3. Naivusis Bajeso klasifikatorius

Naivusis Bajeso klasifikatorius, tai metodas, pagrįstas Bajeso teorema (žr. 1 formulę). Šis modelis apskaičiuoja įvestų duomenų priklausymo tam tikrai kategorijai tikimybę. Analizuojant tekstą, Bajeso klasifikatorius gali būti naudojamas tam tikrų žodžių ar frazių skirstymui į kategorijas [31, 32].

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)} \quad (1)$$

čia: $P(A)$ – apriorinė A hipotezės tikimybė, $P(A|B)$ – A hipotezės tikimybė, atsirandant B įvykiui (aposteriorinė tikimybė); $P(B|A)$ – įvykio B atsiradimo tikimybė esant teisingai A hipotezei; $P(B)$ – įvykio B tikimybė.

1.2.4.4. Logistinė regresija

Logistinė regresija yra naudojama, kai nagrinėjamas kintamasis ar veiksmas gali būti binominis, pavyzdžiui, teisingas arba neteisingas ar lygus 0 arba 1. Logistinės regresijos atveju yra apskaičiuojama tikimybė, kurios yra nuo 0 iki 1, o vietoje tiesės braižoma „S“ formos logistinė funkcija (žr. 2 formulę) [32-34].

$$\text{logit}(Y) = \ln\left(\frac{P}{1-P}\right) = \alpha + \beta_1 X_1 + \dots + \beta_n X_n \quad (2)$$

čia: P – įvykio tikimybė, α – laisvasis narys, β – regresijos koeficientas, X – nepriklausomas kintamasis.

1.2.4.5. Algoritmų palyginimas

3 lentelėje pateikiamas pagrindinių prižiūrimo mašininio mokymo algoritmų, naudojamų duomenų klasifikavimui, palyginimas.

3 lentelė. Algoritmų palyginimas [29-34]

Algoritmo pavadinimas	Privalumai	Trūkumai
Sprendimų medis	<ul style="list-style-type: none"> – Gana nesudėtingas veikimas ir implementacija; – Puikiai tinka gana nedideliems duomenų kiekiams apdoroti. 	<ul style="list-style-type: none"> – Didelė sukurto modelio pritaikymo prie mokymosi duomenų aibės tikimybė (angl. <i>overfitting</i>); – Tikslumui didelę įtaką daro kraštutinės reikšmės ir duomenys; – Pasižymi mažesniu tikslumu, nei kiti algoritmai; – Reikia gana daug laiko apsimokymui.
Atsitiktinis miškas	<ul style="list-style-type: none"> – Mažesnė duomenų variacija nei naudojant sprendimų medžio algoritmą; – Puikiai apdoroja didesnę kiekį duomenų ir identifikuoja kintamuosius svarbius klasifikavimui. 	<ul style="list-style-type: none"> – Yra nemaža modelio pritaikymo prie mokymosi duomenų aibės tikimybė (angl. <i>overfitting</i>); – Pasižymi mažesniu tikslumu, nei kiti algoritmai.
Atraminų vektorių klasifikatorius	<ul style="list-style-type: none"> – Nedidelė modelio pritaikymo prie mokymosi duomenų aibės tikimybė (angl. <i>overfitting</i>); – Gali apdoroti didelį kiekį duomenų; – Gali būti vykdomas skirstymas tiek į dvi, tiek ir daugiau klasių. 	<ul style="list-style-type: none"> – Esant dideliame duomenų kiekiui, algoritmo veikimas tampa gana imlus laikui. – Reikalauja daugiau resursų nei, pavyzdžiui, Naivusis Bajeso klasifikatorius. – Siekiant, jog algoritmas veiktų efektyviai, apsimokymui skirti duomenys turi būti tinkamai paruošti – atspindėti jų populiaciją.
Naivusis Bajeso klasifikatorius	<ul style="list-style-type: none"> – Nedidelė modelio pritaikymo prie mokymosi duomenų aibės tikimybė (angl. <i>overfitting</i>); – Puikiai apdoroja didelį kiekį duomenų; – Gali būti vykdomas skirstymas tiek į dvi, tiek ir daugiau klasių. 	<ul style="list-style-type: none"> – Esant dideliame duomenų kiekiui, algoritmo veikimas tampa gana imlus laikui. – Siekiant, jog algoritmas veiktų efektyviai, apsimokymui skirti duomenys turi būti tinkamai paruošti – atspindėti jų populiaciją.
Logistinė regresija	<ul style="list-style-type: none"> – Nesudėtinga implementacija; – Jis gali būti lengvai išplėstas į kelias klases (polinominė regresija); – Gali apdoroti didelį kiekį duomenų; – Modelis gali interpretuoti tikimybių koeficientus kaip savybių svarbos rodiklius. 	<ul style="list-style-type: none"> – Esant dideliame duomenų kiekiui, algoritmo veikimas tampa gana imlus laikui. – Siekiant, jog algoritmas veiktų efektyviai, apsimokymui skirti duomenys turi būti tinkamai paruošti – atspindėti jų populiaciją.

1.2.5. Internetinio turinio filtravimo metodai

Vienas iš dažniausiai naudojamų ir paprasčiausiai diegiamų metodų yra URL filtravimas. Toks filtravimas gali būti atliekamas dvejopai – analizuojant URL adreso eilutėje esančius simbolius arba blokuojant tam tikrus URL. Pirmuoju atveju yra iš anksto sudaromas žodžių, o antruoju – konkrečių URL adresų sąrašas, kuris bus naudojamas tinklapių identifikavimui ir internetinio turinio filtravimui. Filtravimo sistema atlieka palyginimą prieš atvaizduojant tinklapį naršyklės lange. Kadangi užtenka išanalizuoti tik vieną simbolių eilutę, šis filtravimas pasižymi sąlyginai didele sparta, tačiau gali būti gana netikslus, nes URL adresas gali neatspindėti tikrojo tinklapio turinio. Svarbu paminėti ir tai, jog naujai sukurtus tinklapius tokiu atveju yra gana sunku identifikuoti ir įtraukti į juoduosius sąrašus dėl spartaus bei nereguliaraus vartotojui neigiamą įtaką galinčio turėti tinklapio URL keitimo. Taigi, ši metodą palankiausia naudoti siekiant blokuoti prieigą prie iš anksto žinomų tinklapių ar diegti kartu su kitais filtravimo metodais [35-38].

Kinijos mokslininkai sukūrė kenkėjiškų URL adresų aptikimo sistemą, kurios veikimas pagrįstas dirbtiniu neuroniniu tinklu, derinant „CapsNet“ (angl. *Capsule Neural Network*) ir „IndRNN“ (angl. *Independently Recurrent Neural Network*) technologijas. Algoritmas susideda iš dviejų pagrindinių dalių – semantinės, kurioje pasitelkiant tam tikras funkcijas, URL adrese aptikti simboliai ir žodžiai yra perkeltami į dvimatį tenzorių, ir vizualios, kuriame URL turinys yra konvertuojamas į pilkos spalvos vaizdą. Atlikti tyrimai parodė, jog sukurta sistema pasižymėjo itin aukštu – 99,78 % – klasifikavimo tikslumu [39].

Veloro technologijų instituto studentų vykdytas eksperimentas atskleidė, kad jų sukurtas URL klasifikavimo algoritmas, atskiria ne vaikams skirtą internetinį turinį 75,21-82,04 % tikslumu. Naudojant šią sistemą, URL esanti informacija yra tokenizuojama, pašalinami nereikšmingi ar pasikartojantys žodžiai, o informacijos apdorojimui naudojamas neprižiūrimo mašininio mokymosi algoritmas „GloVe“ (angl. *Global Vector for Word Representation*), neuroninio tinklo architektūra [40].

Kito mokslinio darbo metu buvo sukurtas kenkėjiškų URL adresų atpažinimo algoritmas. Buvo ištirtas kelių skirtingų mašininio mokymo algoritmų – atsitiktinio miško, Naiviojo Bajeso ir logistinės regresijos – veikimo efektyvumas. Nustatyta, kad geriausiomis kenkėjiškų URL atpažinimo savybėmis pasižymėjo logistinės regresijos algoritmas (apie 98 %). Taip pat buvo ištirta ir apmokymui bei testavimui skirtų duomenų santykio (1:1, 4:1, 10:1) įtaka algoritmo efektyvumui. Atliktas tyrimas parodė, jog efektyviausiai algoritmai veikė naudojant 10:1 santykį, tačiau skirtumas nebuvo žymus ir vidutiniškai siekia apie 3 % [41].

Kitas metodas – HTML kodo analizavimu grįstas metodas. Ši metodika, naudojama siekiant nustatyti tinklapio pobūdį, apima HTML žymų, kuriose yra metaduomenų apie tinklapį, nagrinėjimą. Tokiam procesui atlikti taip pat yra naudojami iš anksto sudaryti žodžių ir/ar frazių sąrašai. Svarbu paminėti tai, kad HTML žymų analizavimas reikalauja gana didelių galios ir laiko resursų, ypač tuo atveju, kai tokio tipo filtravimas veikia sistemose, apimančiose platų vartotojų tinklą [37].

Indijos mokslininkai pateikė sukčiavimo tinklapių aptikimo metodą, kuriuo yra apdorojamas HTML kodas. Mašininio mokymosi modelio apmokymui yra naudojama iš tinklapio HTML šaltinio kodo išskirtų atributų. Sistemoje veikia logistinės regresijos algoritmas, o filtravimo tikslumas siekė apie 98 % [42].

Kitame tyrime buvo kuriama sukčiavimo tinklapių aptikimo sistema, kurios veikimas grįstas URL ir HTML šaltinio kodo analizavimu. Palyginus filtravimo efektyvumą naudojant URL ir HTML atributus atskirai, jis siekė atitinkamai maždaug 98 % ir 85 %. Eksperimento rezultatai atskleidė, kad šias funkcijas apjungiant, sistemos efektyvumas išauga iki 99 % [43].

Verta paminėti ir tai, jog filtravimą galima atlikti ir pakopomis ar naudojant kelis skirtingus filtravimo metodus, jog tinklapį būtų galima įvertinti kompleksiskai. Tačiau tokiu atveju, siekiant, kad jų naudojimas būtų efektyvus, reikėtų rinktis tarpusavyje derančius metodus, įvertinti laiko ir kitų resursų sąnaudas. Malaizijos mokslininkai pasiūlė hibridinį vaikams netinkamo turinio ribojimo metodą. Ši sistema naudoja ne tik URL adresus talpinančių juodųjų sąrašų, bet ir Naiviojo Bajeso algoritmą. Siekiant taupyti resursus ir sumažinti užklausų apdorojimo laiką, šis algoritmas veikia etapais – pirmiausia yra analizuojama tinklapio antraštė, joje neaptikus netinkamų žodžių, peržiūrimi pirmasis šimtas, toliau penki šimtai ir tūkstantis tinklapyje esančių žodžių. Atliktas tyrimas parodė,

kad didžiausias tinklapių filtravimo tikslumas buvo pasiektas ribojant teksto analizavimą iki penkių šimtų žodžių [44].

Srities vardų struktūra, arba kitaip, DNS, kuri vartotojams lengvai įsimenamais domenų pavadinimus paverčia į IP adresus, taip pat gali būti naudojama internetinio turinio filtravimui. Taikant šį metodą, yra sukuriamas juodasis sąrašas su neleidžiamo turinio domenų adresais. Vartotojui siekiant prieigos prie tam tikro tinklapio, jo domenas yra paieškomas sudarytame juodajame sąraše. Jei toks adresas aptinkamas, domenas nėra pakeičiamas atitinkamu, jam skirtu IP adresu, o naudotojui bus atvaizduojamas klaidos ar kitas pranešimas. Šis filtravimo metodas yra labiau ribotas, nei, pavyzdžiui, URL filtravimas, tačiau reikalauja mažiau resursų. Taip pat išsiskiria tuo, jog blokuojant tam tikrą IP adresą, bus uždrausta prieiga prie visų jam priklausančių domenų [45, 46].

4 lentelėje pateikiamas internetinio turinio filtravimo metodų palyginimas.

4 lentelė. Internetinio turinio filtravimo metodų palyginimas [35-46]

Filtravimo metodas Savybė	URL adresų filtravimas	HTML kodo analizavimas	DNS filtravimas
Resursų naudojimas	Žemas/Vidutinis	Aukštas	Žemas
Užklausų apdorojimo sparta	Aukšta	Žema	Aukšta
Galimybė naudoti plačiame vartotojų tinkle	Yra	Yra, bet sudėtinga	Yra

Atlikus internetinio turinio filtravimo metodų palyginimą nustatyta, jog HTML kodo analizavimas pagal raktinius žodžius yra vienas efektyviausių filtravimo metodų, nes yra analizuojamas didelis duomenų kiekis, tačiau tai reikalauja gana didelių įrangos ir laiko sąnaudų. Naudojant URL adresų bei DNS filtravimą užklausos apdorojimo sparta yra gana didelė, o resursų sunaudojama sąlyginai nedaug, taigi, iš esmės šiuos metodus galima taikyti filtravimą atliekant organizacijos ar namų ūkio tinkle.

1.3. Apibendrinimas

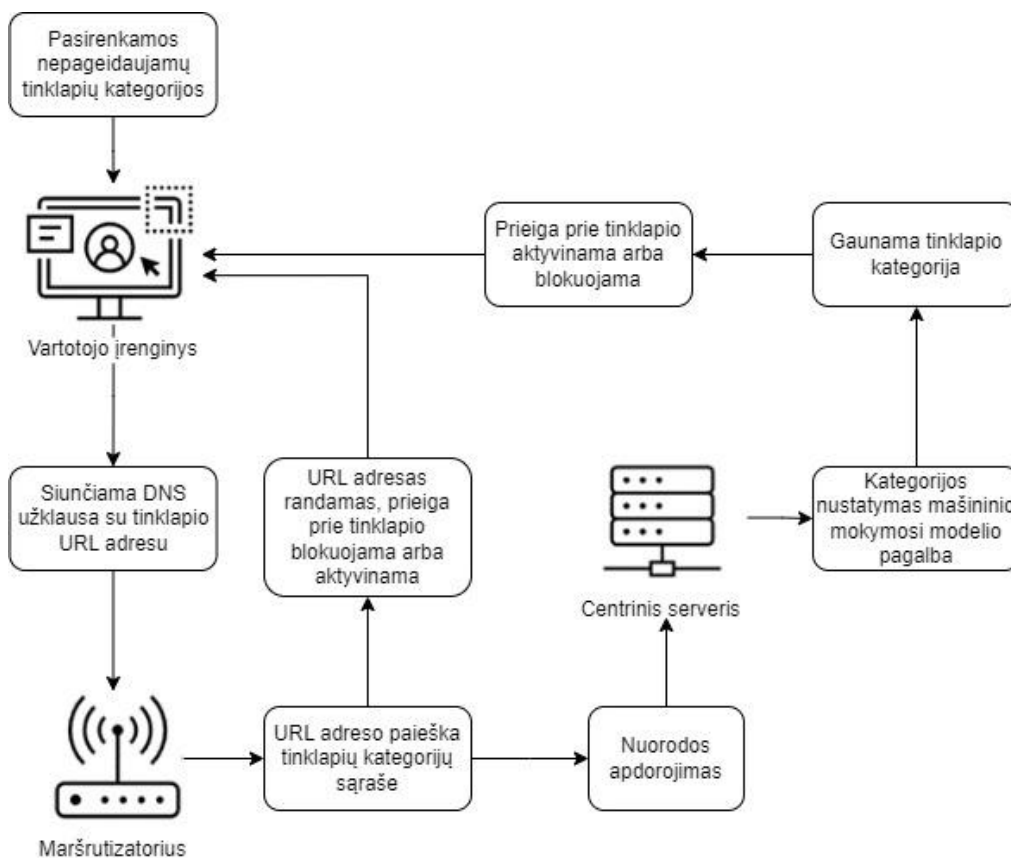
Apžvelgti LR įstatymai ir kiti teisiniai dokumentai patvirtino internetinio turinio filtravimo sistemų poreikį bei aktualumą. Internete esančio turinio ribojimas bei filtravimas gali būti naudojamas ne tik siekiant apsaugoti nepilnamečius asmenis nuo neigiamą įtaką darančios informacijos, bet ir įmonių darbuotojus ar nevalingai nuklydusius vartotojus. Išanalizavus galimus internetinio turinio filtravimo taškus paaiškėjo, jog optimaliomis savybėmis pasižymi vidiniame tinkle vykdomas filtravimas – tokia sistema yra gana patikima, ją nesudėtinga modifikuoti, įmanoma sukurti kelis prieigos lygius, vienu metu galima filtruoti iš kelių ar keliasdešimt įrenginių siunčiamas užklausas, taip pat tam paprastai nereikalinga papildoma aparatinė įranga. Apžvelgus filtravimo metodus buvo nustatyta, jog naudojant URL adresų filtravimą, DNS blokavimą, paremtą skirstymu į kategorijas, užklausų apdorojimo sparta yra gana didelė, o resursų sąnaudos vidutinės-žemos, todėl šį metodą būtų galima naudoti tinkle, maršrutizatoriuose vykdomam filtravimui. Siekiant padidinti sistemos skirstymo į kategorijas efektyvumą, būtų galima analizuoti ir tinklapio HTML kodą. Nors decentralizuotas sistemos veikimas užtikrintų didesnę jos saugumą, tačiau užklausų apdorojimo sparta tiesiogiai priklausytų nuo įrenginių, kuriuose būtų įdiegta sistema, pajėgumo. Atlikus prižiūravimo mašininio mokymo algoritmų palyginimą buvo nuspręsta kuriamai sistemai naudoti logistinę regresiją.

2. Tinklapių filtravimo sistemos projektas

Šiame skyriuje aprašomas sukurtos paskirstytos tinklapių filtravimo sistemos, skirtos maršrutizatoriams, koncepcinis modelis, taip pat keliami funkciniai ir nefunkciniai reikalavimai.

2.1. Sprendimo koncepcija

6 pav. pateikiama kuriamos tinklapių filtravimo sistemos koncepcija. Siekiant riboti prieigą prie tam tikros internete talpinamos informacijos dažnai yra naudojamas filtravimas galutiniame vartotojo įrenginyje, tačiau iškyla sunkumų, jei tam tikri tinklapiai turėtų būti blokuojami keliuose įrenginiuose tuo pačiu metu. Kuriamas sistemos modelis bus pagrįstas mašininio mokymu, logistinės regresijos algoritmu, URL bei metaduomenų analizavimu, o skirstymas į kategorijas vykdomas centriniame serveryje, filtravimas – maršrutizatoriuje.



6 pav. Tinklapių filtravimo sistemos koncepcija

Siekiant nuo žalingo ar kito nepageidaujamo/draudžiamo turinio apsaugoti vietinio tinklo įrenginių naudotojus, administratorius pasirenka tik leidžiamos arba tik blokuojamos informacijos kategorijas. Iš kiekvieno tinkle esančio įrenginio siunčiamos DNS užklausa yra apdorojamos maršrutizatoriuje – tokiu būdu lankomų tinklapių domenai konvertuojami į IP adresus. Jei tinklapio domenas aptinkamas anksčiau sudarytame tinklapių-kategorijų sąrašė, atsižvelgiant į tai, ar jis priklauso leidžiamai ar draudžiamai kategorijai, vartotojui atvaizduojamas norimas tinklapis arba draudžiamos prieigos langas. Sąrašė tinklapio domeno neaptikus, inicijuojamas mašininio mokymosi pagrįstas jo kategorijos nustatymas, vykdomas centriniame serveryje. Tokiu atveju, iš centrinio serverio į maršrutizatorių siunčiamas pranešimas, talpinantis tinklapio nuorodą bei jai priskirtą kategoriją.

2.2. Keliami reikalavimai

Kuriamai tinklapių filtravimo sistemai keliami šie funkciniai reikalavimai:

- skirstymas į kategorijas turi būti vykdomas centriniame serveryje, o filtravimas – maršrutizatoriuje;
- sutrikus centrinio serverio veikimui, filtravimas vykdomas pagal maršrutizatoriuje išsaugotų anksčiau aplankyto tinklapių ir jiems priskirtų kategorijų sąrašą;
- tinklapių skirstymui į kategorijas turi būti naudojamos ne mažiau nei dvi skirtingos tinklapio ypatybės.

Nefunkciniai reikalavimai:

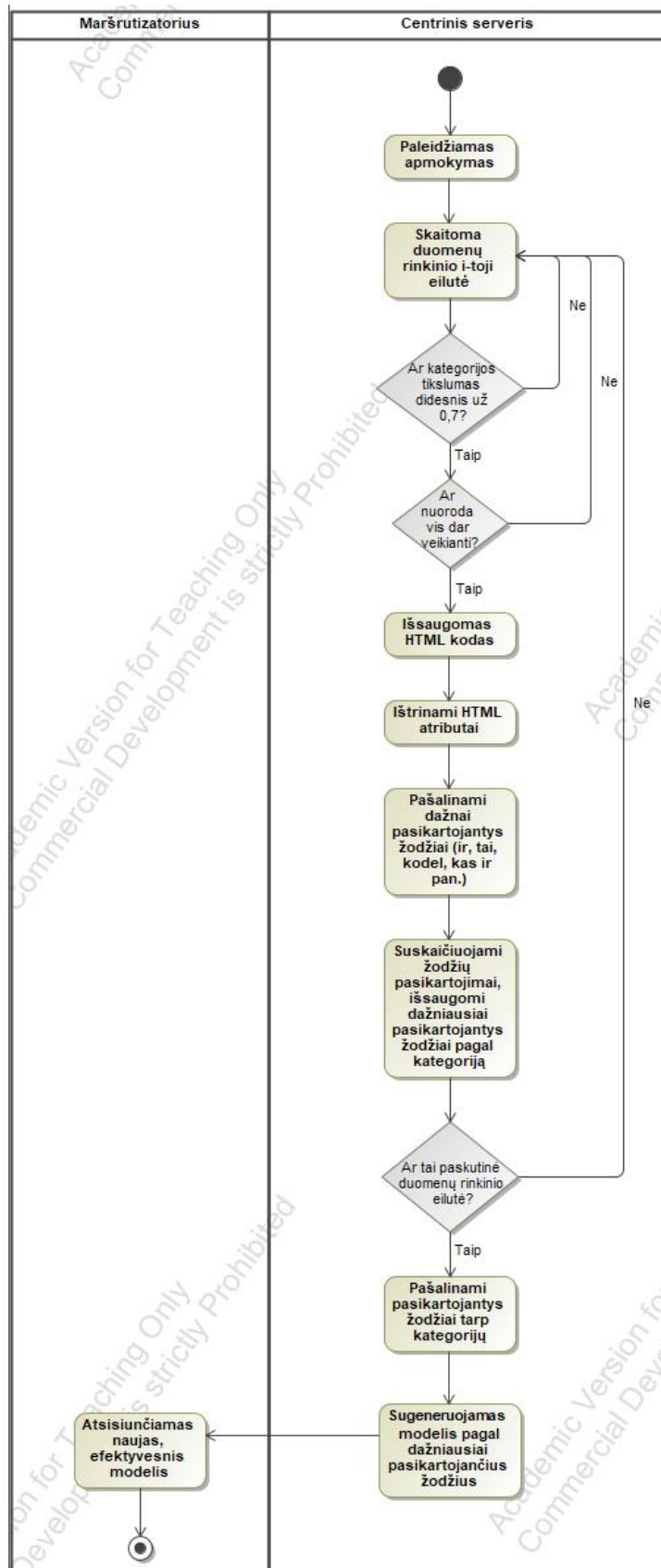
- kuriamos sistemos veikimas turi būti paremtas mašininio mokymo algoritmu;
- tinklapio kategorijos nustatymo procesas negali sumažinti vidutinio vartotojo naršymo greičio daugiau nei 5 kartus arba bendra vidutinė užklauskos apdorojimo trukmė negali viršyti 2,5 s;
- tinklapių filtravimo sistemos parametrų valdymas turi būti prieinamas tik įgaliotiems vartotojams.

2.3. Sistemos modelis

Kuriamos sistemos modelio veikimas pavaizduotas pagrindinių vykstančių procesų veiklų diagramose (žr. 7-12 pav.).

7 paveikslėlyje pateikta dažniausiai pasikartojančių žodžių sąrašo generavimo veiklos diagrama. Vienas iš esminių aspektų, nuo kurių tiesiogiai priklauso prižiūrimu mašininio mokymu paremtos sistemos veikimo efektyvumas, tai modelio apmokymui ir testavimui skirti duomenys, jų kokybė. Kuriant algoritmą, kuriuo įvairūs tinklapiai bus skirstomi į kategorijas, naudojami du pagrindiniai parametrai – tinklapio nuoroda ir kategorija.

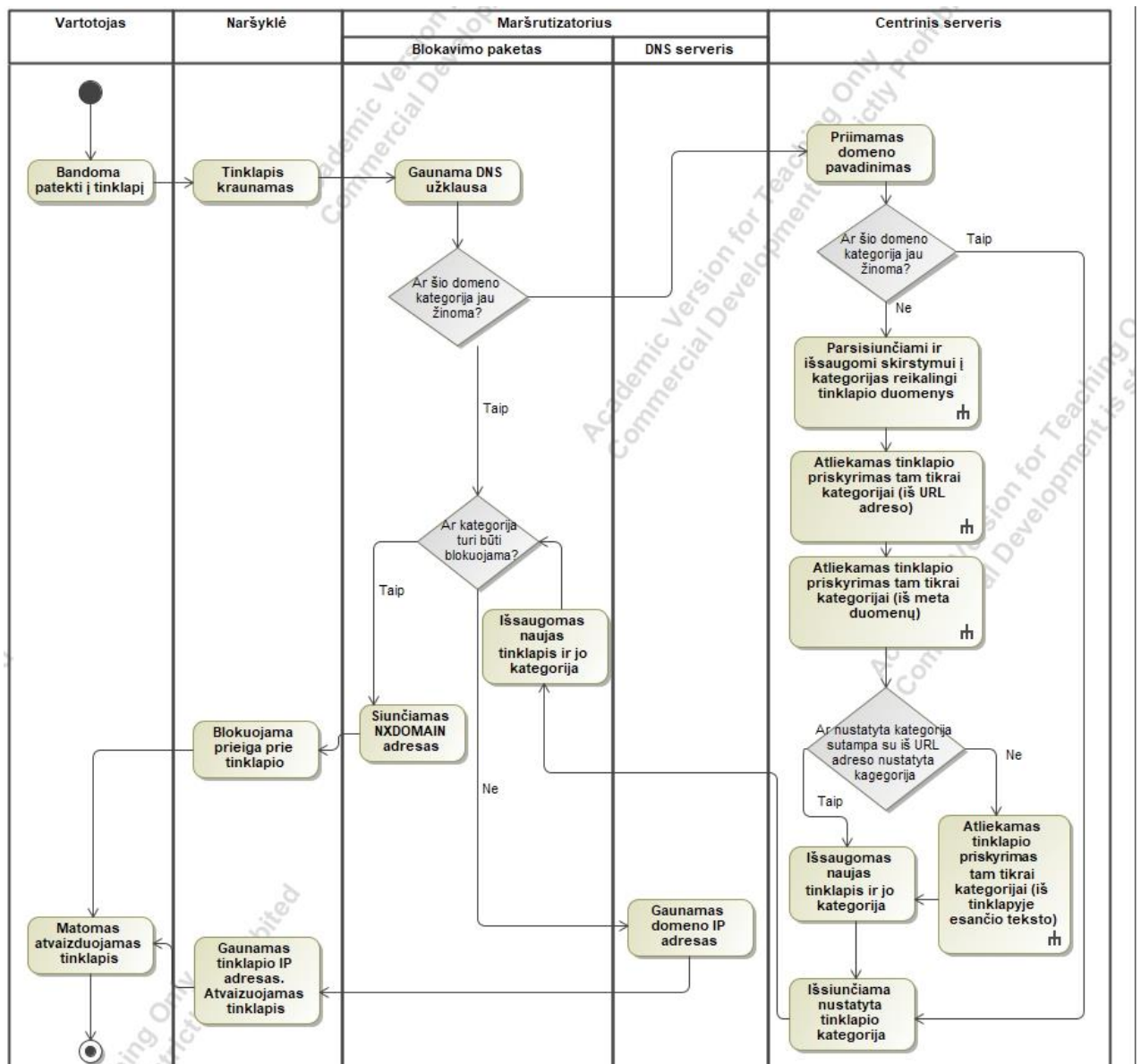
Siekiant taupyti resursus ir optimizuoti sistemos modelio darbą, svarbu užtikrinti, jog naudojami duomenys būtų aktualūs – nuolat tikrinami bei atnaujinami. Tinklapių aktyvumas tikrinamas atsiųstus HTTP antraštę ir RFC7231 būsenos kodą. Gavus 2xx grupės (apie sėkmingą užklauską) HTTP atsako kodą, yra nustatoma, jog tinklapis aktyvus. Toliau yra apdorojamas tinklapio HTML kodas – iš jo pašalinami HTML atributai, nereikšmingi jungiamieji ir kiti žodžiai, tokie kaip, pavyzdžiui, „and“ ir „the“. Tuomet, siekiant priskirti tinklapį tam tikrai kategorijai, skaičiuojamas žodžių pasikartojimų kiekis. Tokiu būdu yra apdorojami visi aktyvūs tinklapiai, taip pat nustatomi ir kiekvienai iš kategorijų priskiriami dažniausiai pasikartojantys žodžiai. Jie surikiuojami sąrašė pasikartojimų skaičiaus mažėjimo tvarka, tai yra, daugiausia kartų pasikartojęs žodis bus pirmasis sąrašė.



7 pav. Dažniausiai pasikartojančių žodžių sąrašo generavimo veiklos diagrama

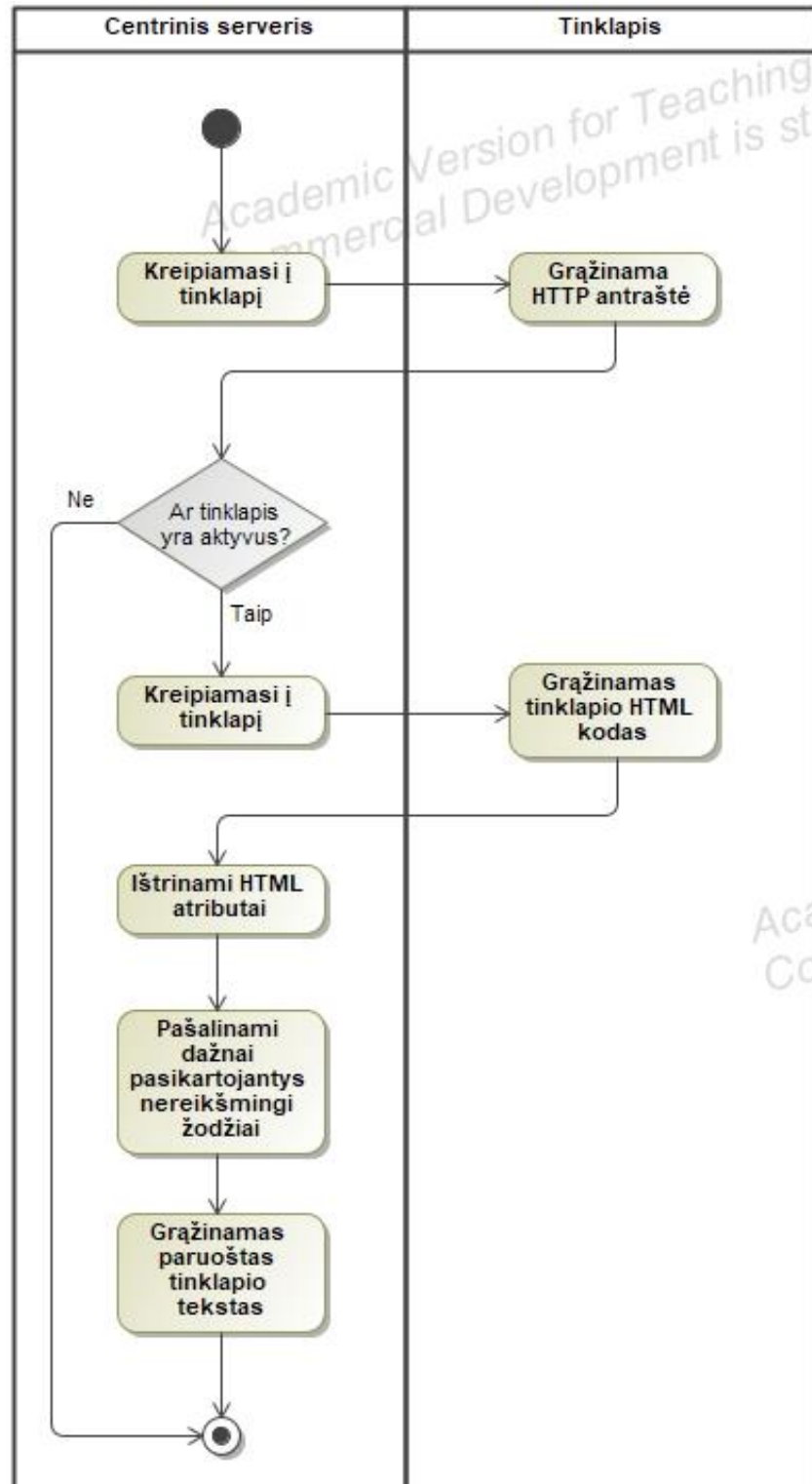
8 pav. pateikta tinklapių skirstymo į kategorijas sistemos algoritmo veiklos diagrama. Centriniam serveryje apmokytas modelis naudojamas atliekant tinklapių skirstymą į kategorijas. Vartotojui įvedus tinklapio adresą interneto naršyklėje, inicijuojama ir į maršrutizatorių siunčiama DNS

užklausa, kuria domeno pavadinimas yra konvertuojamas į IP adresą. Maršrutizatoriuje veikiantis blokavimo paketas pirmiausia patikrina, ar šio tinklapio kategorija dar nebuvo nustatyta anksčiau. Jei ne, siekiant sužinoti naujo tinklapio kategoriją, URL adresas yra siunčiamas į centrinį serverį. Centriniam serveryje taip pat patikrinama, ar šio tinklapio kategorija dar nebuvo nustatyta. Jeigu ne, parsisiunčiami ir išsaugojami skirstymui į kategorijas reikiami tinklapio duomenys. Toliau vykdomas tinklapio priskyrimas kategorijai iš URL adreso ir metaduomenų. Siekiant išvengti nereikalingo sistemos veikimo apsunkinimo, bus renkami duomenys apie naujai aplankytus tinklapius bei jiems priskirtas kategorijas. Jei šiais dviem metodais nustatytos kategorijos sutampa, serveryje išsaugomas tinklapio adresas ir kategorijos pavadinimas. Kitu atveju, atliekamas tinklapio priskyrimas kategorijai iš tinklapio teksto, naudojant logistinės regresijos algoritmą. Tuomet nustatyta tinklapio kategorija siunčiama į maršrutizatorių. Maršrutizatorius gavęs tinklapio kategoriją įvertina, ši kategorija sistemoje leidžiama ar draudžiama, naršyklėje yra atvaizduojamas norimas tinklapis arba pranešimas, informuojantis apie negalimą prieigą. Prieiga prie draudžiamoms kategorijoms priskirtų tinklapių neleidžiama naudojant DNS blokavimą.



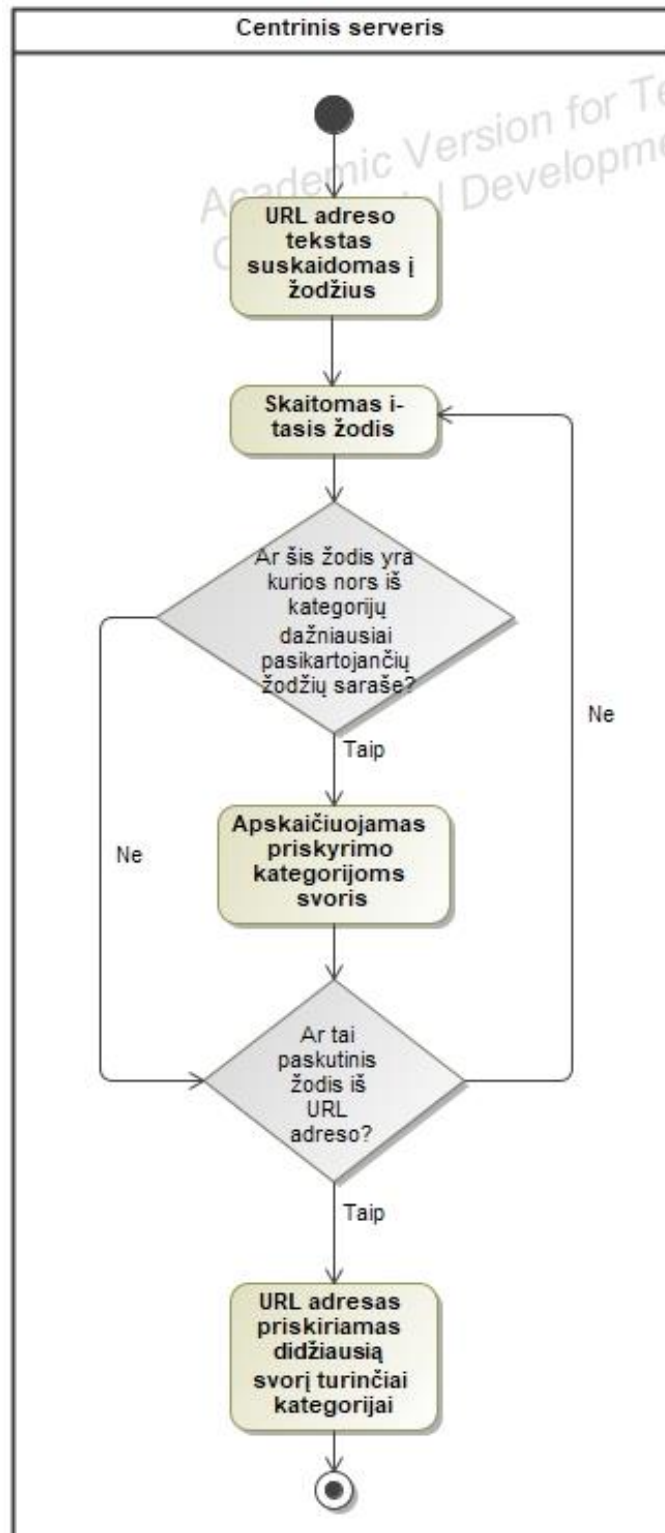
8 pav. Tinklapių skirstymo į kategorijas sistemos algoritmo veiklos diagrama

9 pav. pateikiama tinklapio HTML kodo apdorojimo veiklos diagrama. Pirmiausia patikrinamas tinklapių aktyvumas – atsisiunčiama HTTP antraštė ir RFC7231 būsenos kodas. Gavus 2xx grupės (apie sėkmingą užklausą) HTTP atsako kodą, laikomas, kad tinklapis aktyvus ir tinklapio HTML kodas yra apdorojamas – iš jo pašalinami HTML atributai, nereikšmingi jungiamieji ir kiti žodžiai. Apdorotas tinklapio tekstas perduodamas tolesniems veiksmams.



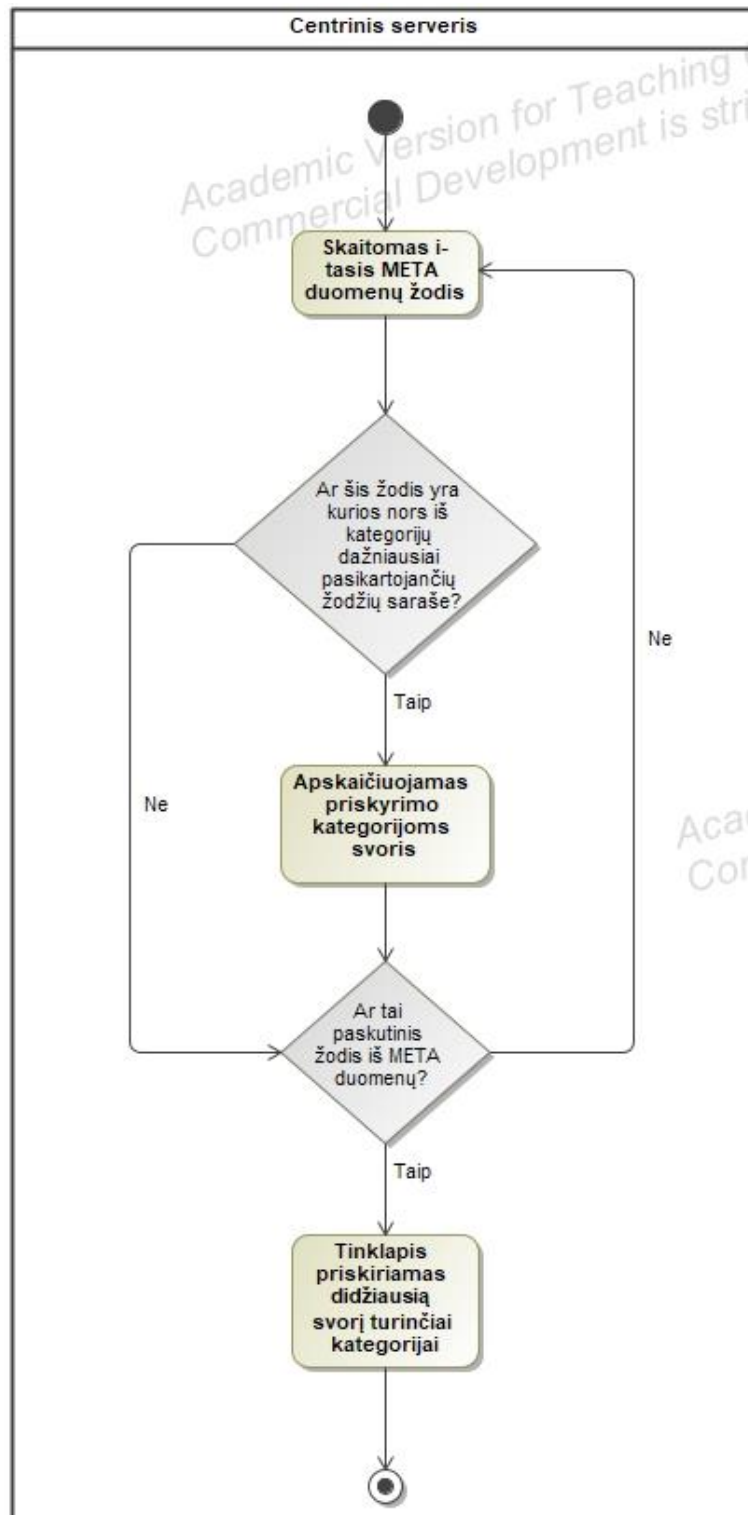
9 pav. Tinklapių HTML kodo apdorojimo veiklos diagrama

10 pav. pateikta tinklapio priskyrimo tam tikrai kategorijai iš URL adreso veiklos diagrama. Pirmiausia URL adresas yra suskaidomas į žodžius. Kiekvienas iš šių žodžių yra ieškomas iš anksto sudarytame tinklapių kategorijoms būdingų dažniausiai pasikartojančių žodžių sąrašuose. Jei šis žodis sąraše neaptinkamas, imamas sekantis žodis. Jeigu žodis yra viename iš sąrašų, apskaičiuojamas priskyrimo atitinkamoms kategorijoms svoris. Šis procesas vykdomas su visais iš URL adreso išskirtais žodžiais. Tinklapio adresas priskiriamas kategorijai, kuri turi didžiausią svorį.



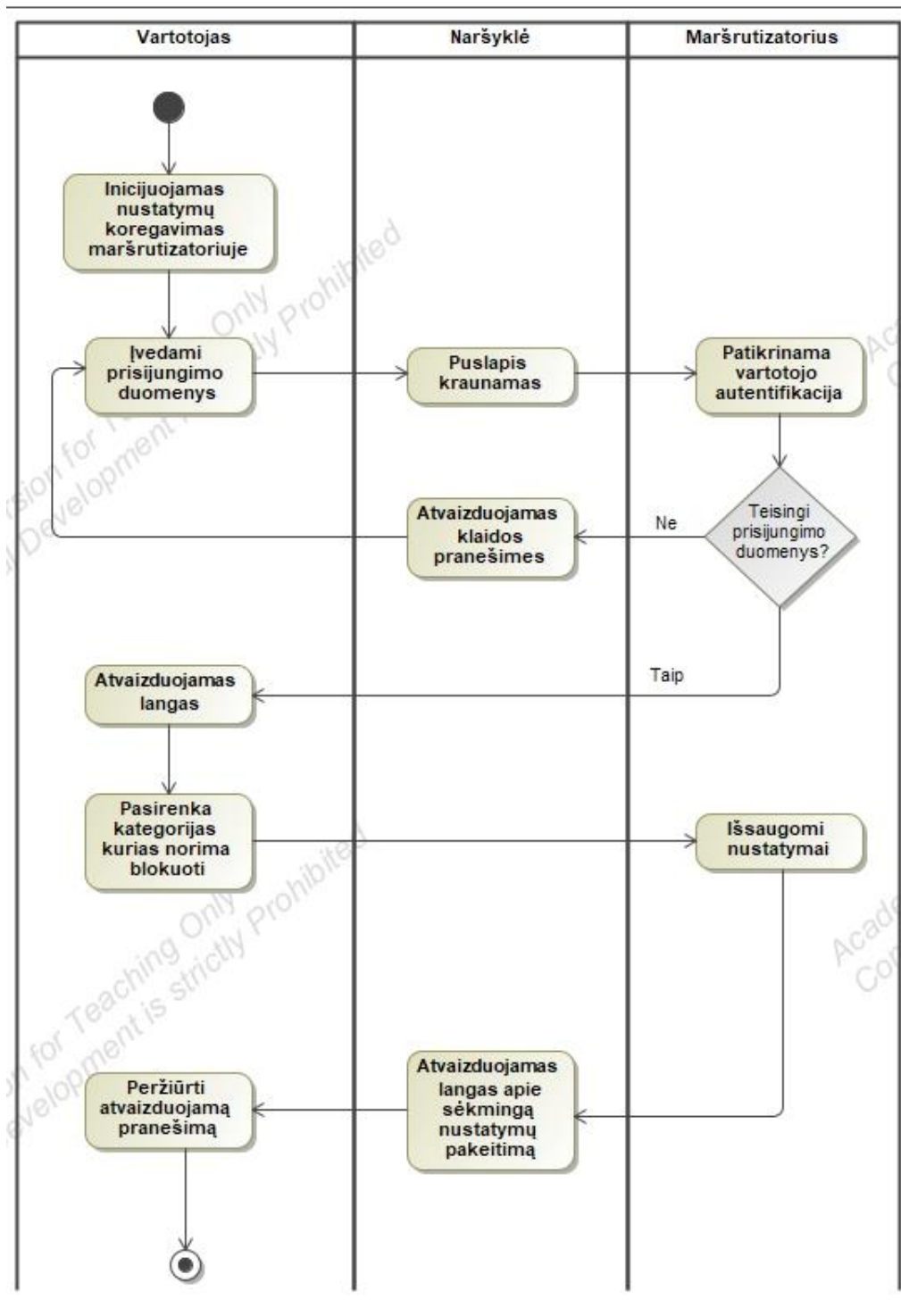
10 pav. Tinklapio priskyrimo tam tikrai kategorijai iš URL adreso veiklos diagrama

11 pav. pateikta tinklapio priskyrimo tam tikrai kategorijai iš metaduomenų veiklos diagrama. Kiekvienas iš metaduomenų išskirtas žodis ieškomas tinklapių kategorijoms būdingų dažniausiai pasikartojančių žodžių sąrašuose. Jeigu tokio žodžio sąraše nėra, imamas sekantis. Jei žodis aptinkamas viename iš sąrašų, skaičiuojamas priskyrimo atitinkamoms kategorijoms svoris. Šis procesas vykdomas su visais iš metaduomenų išskirtais žodžiais. Tinklapio adresas priskiriamas kategorijai, kurios svoris buvo didžiausias.



11 pav. Tinklapio priskyrimo tam tikrai kategorijai iš metaduomenų veiklos diagrama

Maršrutizatoriuje įdiegta tinklapių filtravimo sistema automatiškai veikti nepradės – pirmiausia reikės ją konfigūruoti (žr. 12 pav.).



12 pav. Sistemos pirminio konfigūravimo veiklos diagrama

Sistemos parametrų valdymas prieinamas tik įgaliotiems (prisijungusiems) vartotojams. Siekiant nuo nepageidaujamo ar draudžiamo turinio apsaugoti vietinio tinklo įrenginių naudotojus, administratorius pasirenka tik leidžiamos arba tik blokuojamos informacijos kategorijas. Šiuos nustatymus išsaugojus, administratorius informuojamas apie sėkmingą nustatymų atnaujinimą ir sistema pradeda veikimą.

2.4. Apibendrinimas

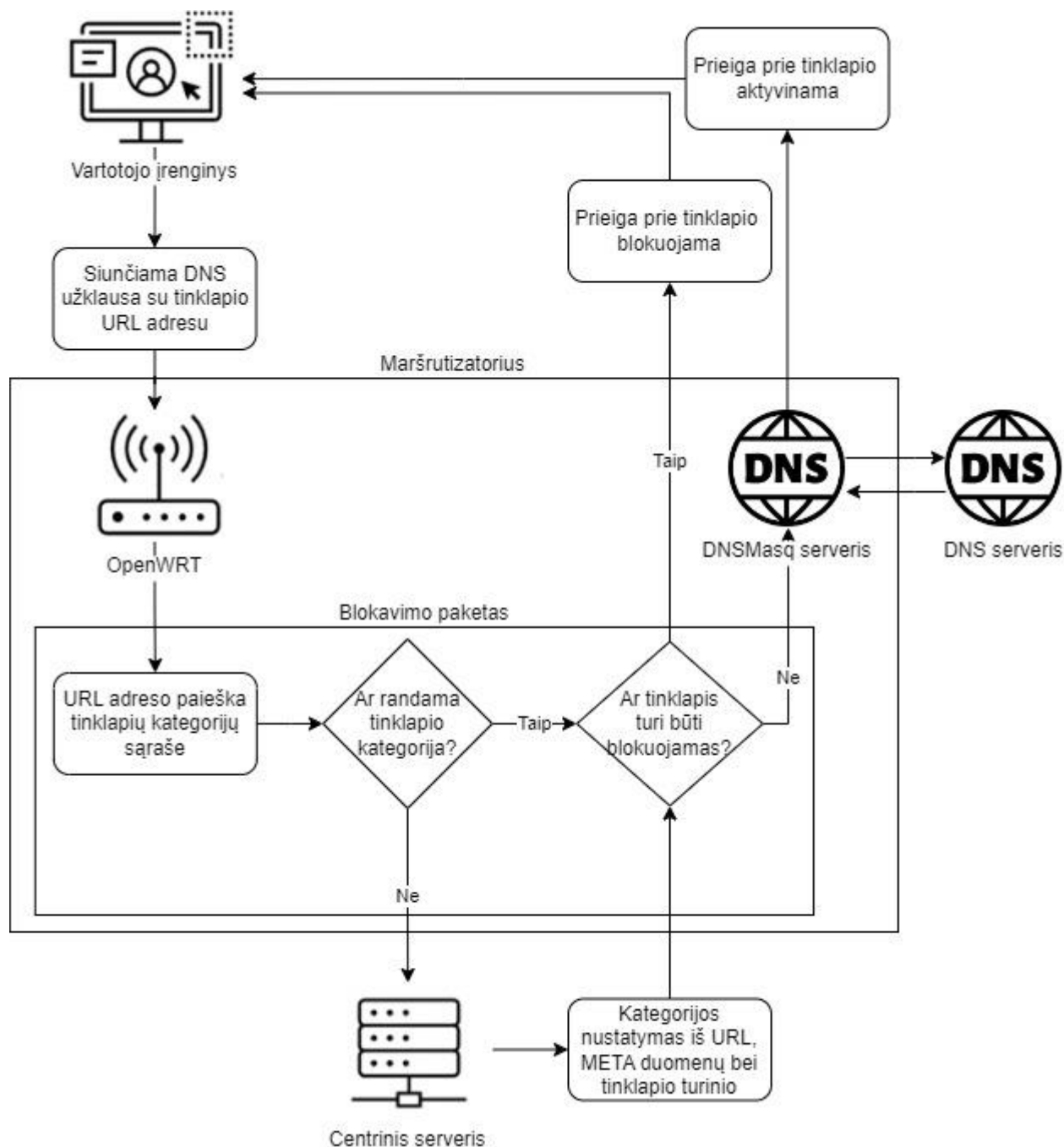
Pasiūlytas kuriamos sistemos modelis pagrįstas mašininio mokymu, logistinės regresijos algoritmu, URL bei metaduomenų analizavimu. Tinklapių skirstymas į kategorijas vykdomas centriniame serveryje, o blokavimo procesas – maršrutizatoriuje, naudojant DNS blokavimą. Kuriamai sistemai išskirti funkciniai ir nefunkciniai reikalavimai užtikrina sistemos efektyvumą ir sklandų jos veikimą. Sistemos modelio veikimo principas ir pagrindinės vykdomos operacijos pavaizduotos veiklų diagramomis.

3. Tinklapių filtravimo sistemos realizacija

Šiame skyriuje pateikiama projekto metu kuriamos tinklapių filtravimo sistemos, diegiamos maršrutizatoriuose, realizacija.

3.1. Prototipo realizacijos vizija

13 paveikslėlyje pateikiama kuriamos sistemos realizacijos vizija.



13 pav. Sistemos realizacijos vizija

Prototipo realizacijai naudojamas centrinis serveris ir maršrutizatorius. DNS užklausa iš vartotojo įrenginio siunčiama į maršrutizatorių, kuriame veiks „OpenWRT“ operacinė sistema ir sukurtas

blokavimo paketas. Šis maršrutizatoriuje veikiantis paketas, kartu su centriniu serveriu vykdydys skirstymu į kategorijas pagrįstą tinklapių filtravimą.

3.2. Tinklapių skirstymo į kategorijas algoritmas ir filtravimo sistemos modelių apmokymas

3.2.1. URL adreso apdorojimas

Filtravimo sistemos duomenų rinkiniui sukurti naudojamas internete prieinamas sąrašas, kuriame talpinami tinklapių domenai ir jiems priskirtos kategorijos. Duomenų rinkinio generavimo įrankis buvo sukurtas naudojant „Python“ programavimo kalbą. Šis įrankis:

- nuskaito reikiamus duomenų rinkinio stulpelius – kategoriją bei URL adresą;
- panaudodamas anglų kalbos žodyną išgauna žodžius iš URL adreso;
- pašalina dažnai pasikartojančius, nereikšmingus žodžius;
- žodžius priskiria kategorijoms ir sudaro dažniausiai pasikartojančių žodžių sąrašą kiekvienai esamai kategorijai;
- išsaugo naujai sukurtą duomenų rinkinį (žr. 14 pav.).

	main_category	tokens
0	Travel	[hotel, travel, resort, site, dream, world, is...
1	Social Networking and Messaging	[chat, elite, tiny, wire, club, tele, gram, si...
2	Streaming Services	[live, sound, movie, stream, music, prime, fac...
3	Sports	[sports, sport, ball, foot, spot, badminton, c...
4	News	[news, times, line, world, global, guar, dian,...
5	Photography	[photo, graphy, camera, photographer, digital,...
6	Law and Government	[dian, narcotic, sind, social, justice, dash, ...
7	Health and Fitness	[health, care, yoga, hospital, fitness, freedo...
8	Games	[game, chess, board, geek, member, tripod, bri...
9	Food	[kitchen, recipe, love, cake, bakery, bake, co...
10	E-Commerce	[gift, shop, clot, hing, fashion, line, eric, ...
11	Education	[science, earth, journal, link, springer, poli...
12	Computers and Technology	[ding, spot, file, soft, ware, gena, risk, boo...
13	Business/Corporate	[star, global, home, bank, free, sair, craft, ...
14	Adult	[porn, tube, galore, ward, dogs, fake, taxi, s...

14 pav. Sukurtas dažniausiai pasikartojančių žodžių sąrašas iš URL adresų

Sugeneruotas ir išsaugotas duomenų rinkinys (žr. 14 pav.) turi tris stulpelius – pirmasis nurodo indeksą, antrasis – kategoriją, o trečiasis talpina visus tam tikrai kategorijai būdingus rastus žodžius, surikiuotus pagal pasikartojimo dažnumą.

Tinklapių priskyrimui tam tikrai kategorijai pagal URL adresą buvo sukurtas *Bag of Words* (BoW) modeliu paremtas mašininio mokymosi metodas. Šis metodas veikia tokia seka:

1. nuskaitomas anksčiau sukurtas dažniausiai pasikartojančių žodžių sąrašas (žr. 14 pav.);
2. naujas URL adresas (tinklapis, kurį ketinama priskirti tam tikrai kategorijai) priimamas kaip argumentas;
3. iš URL adreso išgaunami žodžiai;
4. atsižvelgiant į URL adresą rastus žodžius, nustatomas kiekvienos galimos tinklapių kategorijos svoris;
5. didžiausią svorį turinti kategorija priskiriama tinklapiui (žr. 15 pav.).

```
https://www.crazygames.com/  
crazygames  
['crazy', 'game']  
Predicted main category: Games
```

```
https://www.booking.com/  
booking  
['booking']  
Predicted main category: Travel
```

15 pav. Tinklapio kategorijos nustatymas naudojant URL adresą

15 pav. pateikiami tinklapio priskyrimo tam tikrai kategorijai, pagal URL adresą, pavyzdžiai. Pirmiausia, iš URL adreso (<https://www.crazygames.com/>) išskiriamas reikalingas tekstas (*crazygames*). Tuomet šis tekstas, remiantis anglų kalbos žodynu, yra suskirstomas į žodžius (*crazy, games*). Kiekvienas iš šių žodžių yra ieškomas kategorijoms priskirtuose sąrašuose, vėliau apskaičiuojami svoriai. Tinklapis <https://www.crazygames.com/> priskirtas žaidimų (*Games*) kategorijai. Tinklapis <https://www.booking.com/>, pagal rastą raktinį žodį (*booking*), priskirtas kelionių (*Travel*) kategorijai.

3.2.2. Metaduomenų apdorojimas

Tinklapio metaduomenų apdorojimui buvo sukurtas duomenų generavimo įrankis, kuris:

- nuskaityto reikiamus duomenų rinkinio stulpelius – kategoriją bei URL adresą;
- išgauna žodžius iš tinklapio metaduomenų;
- pašalina dažnai pasikartojančius nereikšmingus žodžius;
- žodžius priskiria kategorijoms ir sudaro dažniausiai pasikartojančių žodžių sąrašą kiekvienai esamai kategorijai;
- išsaugo naujai sukurtą duomenų rinkinį (žr. 16 pav.).

	main_category	tokens
0	Travel	[hotel, travel, tour, room, flight, view, reso...
1	Social Networking and Messaging	[chat, room, people, user, friend, sign, membe...
2	Streaming Services	[live, streaming, music, apple, platform, watc...
3	Sports	[stats, league, cricket, ipl, sport, mlb, play...
4	News	[year, ago, say, story, one, public, war, trum...
5	Photography	[photography, photo, camera, photographer, lig...
6	Law and Government	[website, government, service, state, national...
7	Health and Fitness	[care, hospital, health, dr, view, patient, su...
8	Games	[game, amp, card, player, play, one, chess, po...
9	E-Commerce	[gift, price, view, sale, tape, fashion, regul...
10	Food	[recipe, minute, easy, food, chicken, make, ca...
11	Education	[science, priestley, group, university, thesis...
12	Computers and Technology	[file, software, data, quantum, ai, network, v...
13	Business/Corporate	[service, security, technology, learn, company...
14	Adult	[porn, sex, phone, big, hd, tube, cock, xxx, a...

16 pav. Sukurtas dažniausiai pasikartojančių žodžių sąrašas iš metaduomenų

Kaip ir URL adreso apdorojimo atveju, sugeneruotas duomenų rinkinys (žr. 16 pav.) sudarytas iš trijų stulpelių – pirmasis nurodo indeksą, antrasis – kategoriją, o trečiasis talpina visus tam tikrai kategorijai būdingus rastus žodžius, surikiuotus pagal pasikartojimo dažnumą.

Tinklapių priskyrimui tam tikrai kategorijai pagal metaduomenis naudojamas *Bag of Words* (BoW) modeliu paremtas mašininio mokymosi metodas, kuris:

1. nuskaito anksčiau sukurtą dažniausiai pasikartojančių žodžių sąrašą (žr. 16 pav.);
2. naują URL adresą (tinklapis, kurį ketinama priskirti tam tikrai kategorijai) priima kaip argumentą;
3. išgauna metaduomenis iš tinklapių;
4. pašalina dažnai pasikartojančius, kitus nereikšmingus žodžius;
5. pagal metaduomenyse rastus žodžius nustato kiekvienos kategorijos svorį;
6. tinklapių priskiria didžiausią svorį turinčiai kategorijai (žr. 17 pav.)

```
https://www.crazygames.com/  
['game', 'crazygames', 'play']  
Predicted main category: Games  
  
https://www.booking.com/  
['booking', 'official', 'hotel', 'flight', 'car', 'rental', 'accommodation']  
Predicted main category: Travel
```

17 pav. Tinklapių kategorijos nustatymas naudojant metaduomenis

17 pav. pavaizduoti tinklapių priskyrimo tam tikrai kategorijai, pagal metaduomenis, pavyzdžiai. Pirmiausia, iš metaduomenų išskiriami žodžiai (*game, crazygames, play*). Kiekvienas iš šių žodžių yra ieškomas kategorijoms priskirtuose sąrašuose, vėliau apskaičiuojami svoriai – tinklapis <https://www.crazygames.com/> priskiriamas žaidimų (*Games*) kategorijai. Tinklapis <https://www.booking.com/>, pagal rastus raktinius žodžius (*booking, official, hotel, flight, car, rental, accommodation*), priskirtas kelionių (*Travel*) kategorijai.

3.2.3. Tinklapių teksto apdorojimas

Pirminiam filtravimo sistemos duomenų rinkiniui sukurti naudojamas internete prieinamas sąrašas, kuriame talpinami tinklapių domenai, jiems priskirtos kategorijos. Tinklapių teksto apdorojimui skirtas apmokymo modelis:

- nuskaito reikiamus duomenų rinkinio stulpelius – kategoriją ir iš tinklapių teksto išskirtus žodžius;
- panaudojant „TF-IDF Vectorizer“ funkciją, žodžius paverčia vektoriais;
- panaudojant logistinės regresijos algoritmą, sudaro ir išsaugo mašininio mokymo modelį.

Tinklapių priskyrimui tam tikrai kategorijai, pagal jame esantį tekstą, naudojamas logistinė regresija paremtas mašininio mokymosi metodas, kuris:

1. priima naują URL adresą kaip argumentą;
2. išgauna žodžius iš tinklapių turinio;
3. pašalina dažnai pasikartojančius, nereikšmingus žodžius;
4. panaudojus apmokytą logistinės regresijos modelį ir įvertinus tinklapyje aptiktus žodžius, nustato jos kategoriją (žr. 18 pav.).

```

https://www.crazygames.com
['game', 'crazygames', 'play', 'logo', 'crazygames', 'olfran', 'srom', 'game', 'controlle
r', 'compatible', 'gamesnew', 'gamesview', 'kitchen', 'crush', 'cooking', 'gamebike', 'ju
mpmr', 'dude', 'king', 'incgalaxy', 'stroke', 'line', 'drawidle', 'fishingmad', 'survive'
, 'noob', 'crazygames', 'gamesview', 'morebloxd', 'iodeadshot', 'iosmash', 'iosnake', 'io
agar', 'iodoodle', 'gamesview', 'archerszoo', 'happy', 'animalsman', 'runner', 'color', '
asmrcrazy', 'roll', 'dmerge', 'rescue', 'throwtower', 'defense', 'clashcat', 'star', 'riv
al', 'gamesview', 'geography', 'flag', 'merge', 'valleyludo', 'kingsave', 'quiz', 'game',
, 'triviadrop', 'merge', 'gamesview', 'moresuper', 'city', 'zombie', 'ragdolls', 'stickman
', 'royalspace', 'arena', 'cannon', 'battle', 'fight', 'gamesview', 'traffic', 'limitsmot
o', 'mcrazy', 'plane', 'racer', 'dirt drift', 'city', 'racingmx', 'offroad', 'drift', 'gam
esview', 'ball', 'billiards', 'classictap', 'tennis', 'toursoccer', 'legend', 'basketball
', 'gamesview', 'ggbullet', 'forcetime', 'shooter', 'shooter', 'swathazmob', 'shooterair'
, 'funny', 'shooter', 'bank', 'robbery', 'winter', 'clash', 'dsniper', 'shot', 'bullet',
, 'timeplay', 'friend', 'local', 'separate', 'gamesabout', 'crazygames', 'crazygames', 'fea
ture', 'latest', 'game', 'enjoy', 'playing', 'game', 'without', 'downloads', 'intrusive',
, 'load', 'favorite', 'game', 'instantly', 'browser', 'enjoy', 'play', 'game', 'desktop',
, 'mobile', 'device', 'includes', 'everything', 'desktop', 'laptop', 'latest', 'tablet', 'a
pple', 'android', 'game', 'crazygames', 'plenty', 'game', 'active', 'community', 'crazyga
mes', 'find', 'many', 'title', 'game', 'game', 'play', 'friend', 'people', 'around', 'mat
ter', 'play', 'game', 'crazygames', 'game', 'every', 'genre', 'imagine', 'popular', 'game'
, 'shell', 'shocker', 'bullet', 'force', 'drift', 'hunter', 'moto', 'smash', 'karts', 'e
xplore', 'genre', 'find', 'main', 'category', 'like', 'clicker', 'driving', 'game', 'shoo
ting', 'game', 'range', 'find', 'perfect', 'game', 'popular', 'include', 'game', 'minecra
ft', 'fireboy', 'watergirl', 'player', 'game', 'horror', 'mahjong']
Predicted Category: Games

```

a)

```

https://www.booking.com
['booking', 'official', 'hotel', 'flight', 'rental', 'skip', 'main', 'eurlist', 'hotelcar
', 'taxismore', 'find', 'staysearch', 'deal', 'hotel', 'much', 'check', 'date', 'check',
, 'date', 'adult', 'child', 'roomsearch', 'deal', 'special', 'trip', 'summer', 'game', 'bru
ssels', 'quick', 'train', 'ride', 'moment', 'save', 'book', 'stay', 'find', 'getaway', 'd
eal', 'trending', 'popular', 'choice', 'traveler', 'trip', 'hotel', 'angeles', 'hollywood
', 'beverly', 'hill', 'discover', 'hotel', 'angeles', 'stay', 'orlando', 'hotel', 'family
', 'discover', 'orlando', 'hotel', 'family', 'vacation', 'town', 'around', 'discover', 'w
inter', 'wonderland', 'charming', 'vacation', 'getaway', 'enjoy', 'dinner', 'vacation', '
incredible', 'bangkok', 'rooftop', 'amazing', 'city', 'view', 'cocktail', 'class', 'cuisi
ne', 'instant', 'sign', 'booking', 'account', 'look', 'blue', 'genius', 'logo', 'savesign
', 'inregister', 'find', 'love', 'region', 'city', 'place', 'interest', 'ibiza', 'propert
y', 'lake', 'district', 'property', 'phuket', 'province', 'property', 'isle', 'wight', 'p
roperty', 'uttar', 'pradesh', 'property', 'guernsey', 'property', 'texel', 'property', 'c
ornwall', 'property', 'tenerife', 'property', 'hawaii', 'property', 'mykonos', 'property'
, 'bora', 'bora', 'property', 'khaimah', 'property', 'zanzibar', 'property', 'jersey', 'p
roperty', 'bihar', 'property', 'santorini', 'property', 'bali', 'property', 'england', 'p
roperty', 'rome', 'italy', 'vacation', 'rental', 'cottage', 'cabin', 'london', 'united',
, 'kingdom', 'vacation', 'rental', 'cottage', 'cabin', 'glamping', 'paris', 'france', 'vaca
tion', 'rental', 'cottage', 'klaip', 'lithuania', 'vacation', 'rental', 'cottage', 'copen
hagen', 'denmark', 'vacation', 'rental', 'cottage', 'barcelona', 'spain', 'vacation', 're
ntal', 'cottage', 'bangkok', 'hotel', 'york', 'hotel', 'newcastle', 'upon', 'tyne', 'hote
l', 'manchester', 'hotel', 'liverpool', 'hotel', 'leeds', 'hotel', 'glasgow', 'hotel', 'e
dinburgh', 'hotel', 'carbis', 'hotel', 'bristol', 'hotel', 'brighton', 'hove', 'hotel', '
hotel', 'blackpool']
Predicted Category: Travel

```

b)

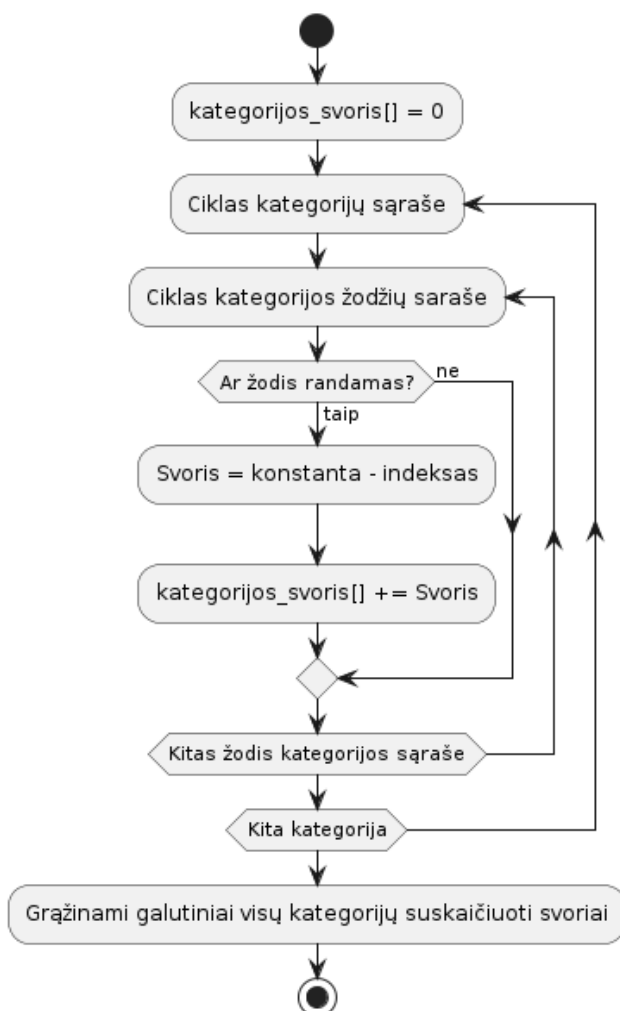
18 pav. Tinklapio kategorijos nustatymas naudojant tinklapio turinį:

a) <https://www.crazygames.com/>; b) <https://booking.com/>.

18 pav. pavaizduoti tinklapio priskyrimo tam tikrai kategorijai, pagal jame aptiktą tekstą, pavyzdžiai. Iš tinklapių turinio išskyrus žodžius, kiekvienas iš jų yra ieškomas kategorijoms priskirtuose sąrašuose, vėliau apskaičiuojami svoriai. Tinklapis <https://www.crazygames.com/> priskiriamas žaidimų (*Games*) kategorijai, o <https://booking.com/> – kelionių (*Travel*) kategorijai.

3.3. Tinklapio kategorijos svorio apskaičiavimas

Tinklapio kategorijos svorio skaičiavimas (žr. 19 pav.), URL adreso ir metaduomenų atveju, prasideda nuo žodžių paieškos kiekvienos kategorijos dažniausiai pasikartojančių žodžių sąrašė. Kiekvienas kategorijos sąrašė esantis žodis turi jam priskirtą indeksą. Pirmiausia yra apskaičiuojamas kiekvieno kategorijos sąrašė aptinkamo žodžio, o vėliau, sudėjus šias reikšmes, ir visos kategorijos svoris. Tokie žingsniai atliekami su visomis penkiolika kategorijų. Tinklapis priskiriamas didžiausią svorį turinčiai kategorijai.



19 pav. Tinklapio kategorijos svorio apskaičiavimo algoritmas

Žodžio svoris skaičiuojamas pagal 3 formulę:

$$\text{Svoris} = \text{konstanta} - \text{indeksas} \quad (3)$$

čia: *konstanta* – skaičius, ne mažesnis už raktinių žodžių kiekį, esantį kiekvienoje iš kategorijų (pasirinktas skaičius – 2000), *indeksas* – žodžiui, esančiam tam tikros kategorijos dažniausiai pasikartojančių žodžių sąrašė, priskirta skaitinė reikšmė.

5 lentelėje pateikiamas kategorijos dažniausiai pasikartojančių žodžių indeksavimo pavyzdys. Tam tikrai kategorijai priskirtame sąrašė žodžiai yra rikiuojami taip, jog dažniausiai pasikartojantis žodis

yra pradžioje, o rečiausiai – pabaigoje. Tai yra, dažniausiai pasikartojančio žodžio indeksas bus mažiausias (0), o rečiausiai – didžiausias.

5 lentelė. Kategorijos dažniausiai pasikartojančių žodžių indeksavimo pavyzdys

Indeksas	0	1	2	3	4
Žodis	hotel	travel	tour	room	flight

20 pav. pateikiamas tinklapio <https://www.crazygames.com/> kategorijos nustatymas iš metaduomenų. Nuspėta tinklapio kategorija yra žaidimai (*Games*) – 596 svorio vienetai.

```

https://www.crazygames.com
['game', 'crazygames', 'play']
      Category  Weight
0      Travel      0
1  Social Networking and Messaging  214
2      Streaming Services  342
3      Sports      517
4      News      332
5      Photography  41
6      Law and Government  0
7      Health and Fitness  0
8      Games      596
9      E-Commerce  35
10     Food      0
11     Education  177
12     Computers and Technology  217
13     Business/Corporate  0
14     Adult      179
Predicted main category: Games

```

20 pav. Tinklapio <https://www.crazygames.com/> kategorijos nustatymas iš metaduomenų

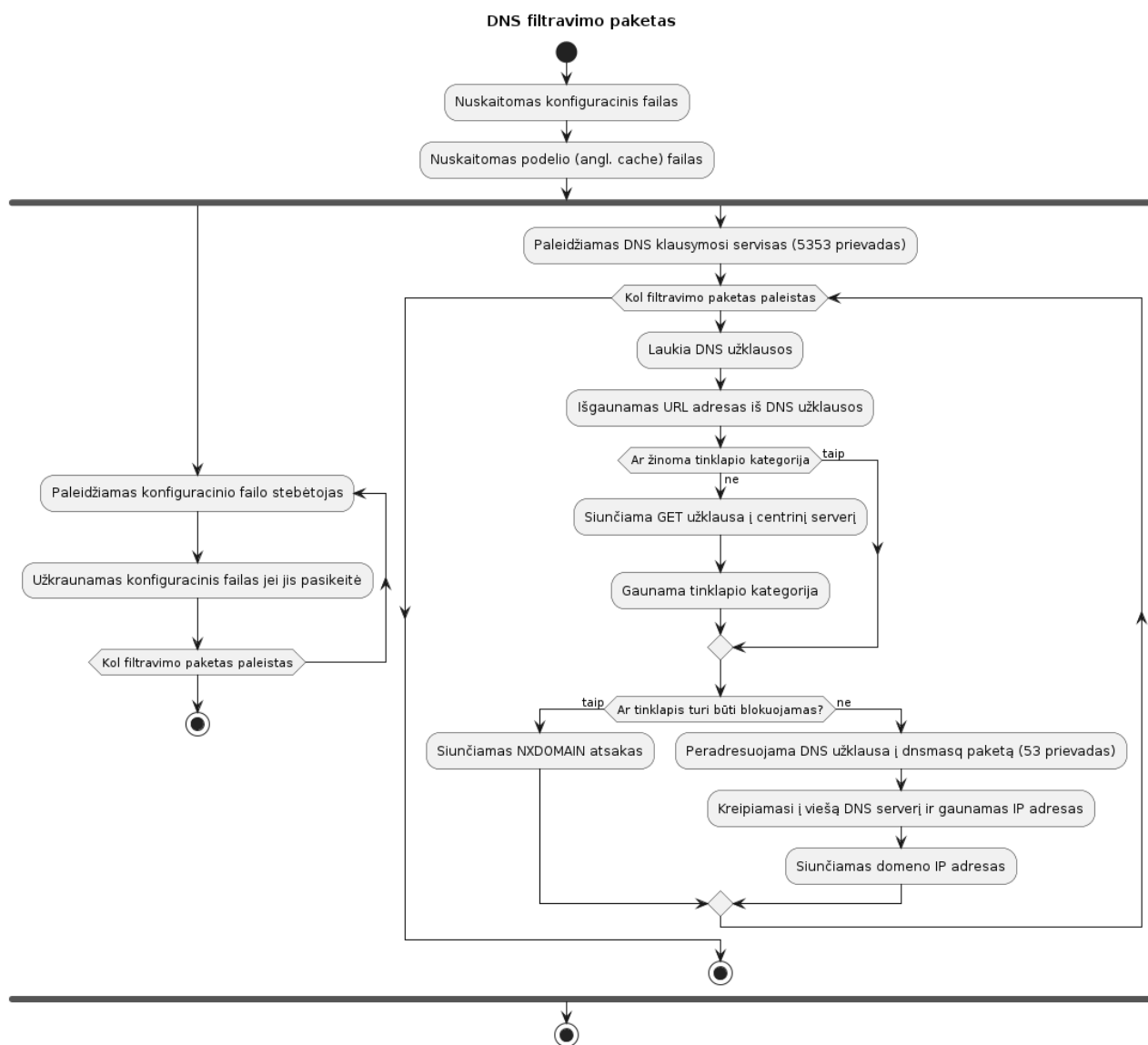
Galima matyti, jog remiantis metaduomenyse aptiktais žodžiais, buvo apskaičiuoti skirtingų kategorijų svoriai. Pavyzdžiui, antroje vietoje esanti sporto (*Sport*) kategorija, surinkusi 517 svorio vienetų, nuo žaidimų kategorijos skyrėsi 79 vienetais, o tarp šio tinklapio ir maisto (*Food*) kategorijos nebuvo rasta jokio bendro turinio.

3.4. Blokavimo paketo veikimas maršrutizatoriuje

Kuriamos sistemos realizacijai pasirinktas „Teltonika RUTX14“ maršrutizatorius, kuriame yra įdiegta modifikuota atvirojo kodo *OpenWRT* operacinė sistema „RUTOS“. Vartotojui pasirinkus blokuojamo internetinio turinio kategorijas yra sukuriamas tą numatantis konfigūracinis failas. Tam, jog filtravimas maršrutizatoriuje galėtų būti vykdomas realiu laiku, buvo sukurtas blokavimo paketas (žr. 21 pav.), kuris:

1. nuskaito konfigūracinį failą su centrinio serverio adresu ir vartotojo pasirinktomis blokuojamomis/leidžiamomis kategorijomis (žr. 22 pav.);
2. apdoroja per 5353 prievadą gaunamas DNS užklausas ir iš jų išgauna tinklapių URL adresą;
3. iš naujo nuskaito konfigūracinį failą, kai jis yra atnaujinamas;
4. patikrina, ar šis URL adresas dar neturi priskirtos kategorijos;
5. neradus URL adreso kategorijos, siunčia užklausa į centrinį serverį bei priima atsakymą iš jo;

6. atsižvelgiant į sukonfigūruotas kategorijas, nustato tinklapis turi būti blokuojamas ar leidžiamas. Siunčia NXDOMAIN (neegzistuojantį domeną), jei tinklapis turi būti blokuojamas. Jei prieiga prie tinklapiio leidžiama, peradresuoja DNS užklausą į 53 prievadą, kuriame standartiškai veikia dnsmasq paketas, apdorojantis DNS užklausas.



21 pav. Maršrutizatoriuje įdiegto blokavimo paketo veikimo schema

```

config block_categories
    option enabled '0'
    option server '8.8.8.8'
    option mode 'blacklist'
    list category 'Travel'
    list category 'Social_Networking_and_Messaging'
    list category 'News'
    list category 'Streaming_Services'
    list category 'Sports'
    list category 'Photography'
    list category 'Law_and_Government'
    list category 'Health_and_Fitness'
    list category 'Games'
    list category 'E-Commerce'
    list category 'Food'
    list category 'Education'
    list category 'Computers_and_Technology'
    list category 'Business/Corporate'
    list category 'Adult'

```

22 pav. Sugeneruotas konfigūracinis failas, numatantis blokuojamas kategorijas

Sukompiliavus modifikuotą programinę įrangą, ji įkeliama į maršrutizatorių. Blokavimo paketo veikimas įgalinamas pakeičiant parametrą „enabled“ į „1“. Tuomet reikia įvesti tinkamą serverio adresą ir nustatyti blokuojamas kategorijas.

3.5. Centrinis serveris

Centriniame serveryje buvo panaudota „Flask“ mikro žiniatinklio sistema ir aplikacijų programavimo sąsaja (angl. *application programming interface*, API). 23 pav. pateiktas kategorijos atvaizdavimas standartiniu JSON formatu.

```

91.211.247.93:3000/get_category/crazygames.com
1 | {
2 |   "category": "Games"
3 | }

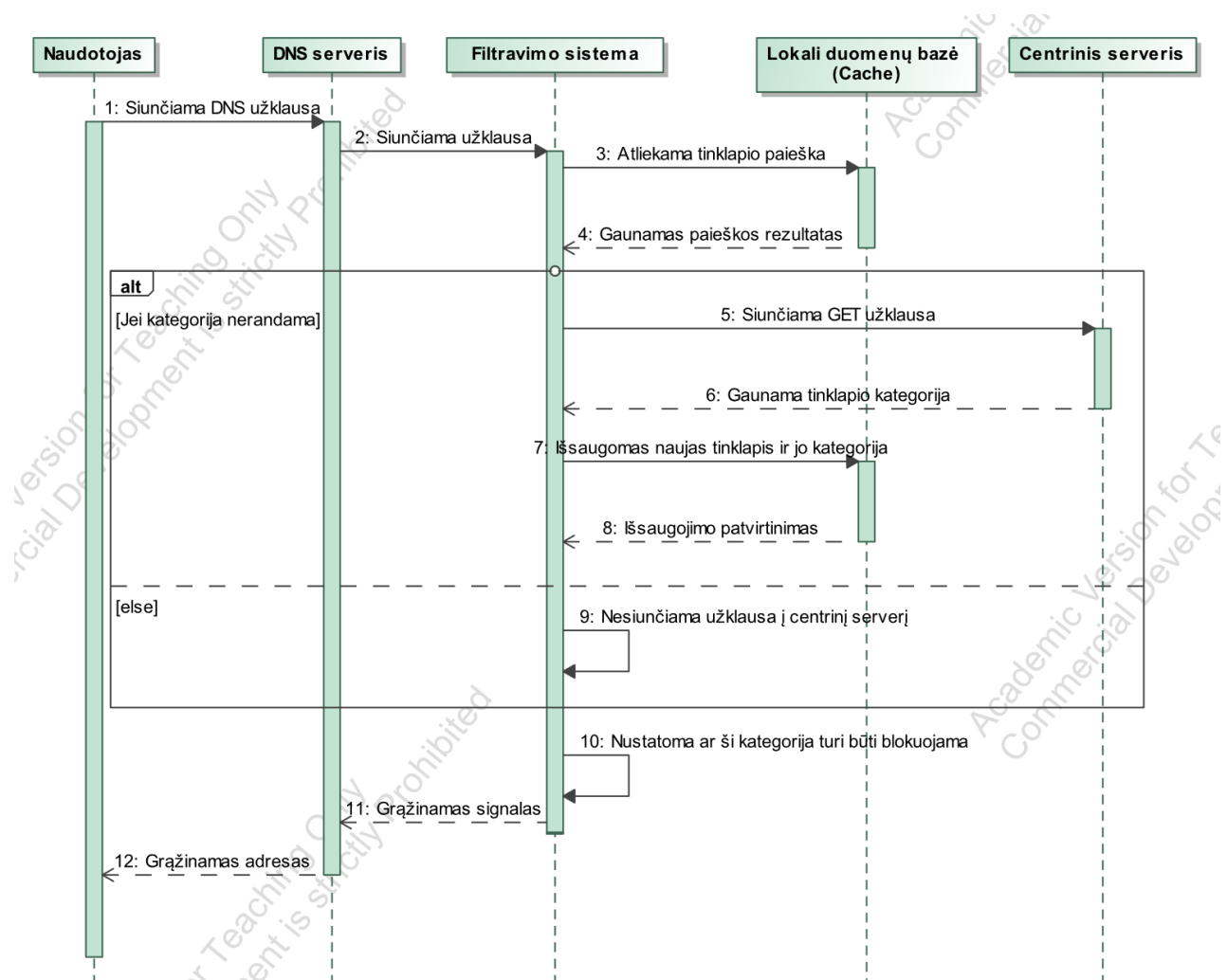
```

23 pav. Kategorijos atvaizdavimas apdorojus GET užklausą

Naudojant aplikacijų programavimo sąsają (angl. *API*) yra apdorojamos GET užklauskos, iš kurių išgaunamas skirstymo į kategorijas algoritmui reikalingas tinklapio adresas. Kategorijos nustatymui yra naudojamas trijų žingsnių analizavimas iš tinklapio URL adreso, metaduomenų bei tinklapio tekstinio turinio.

3.6. Podėliavimas

Siekiant pagreitinti tinklapių užklausų apdorojimą maršrutizatoriuje, centriname serveryje ir sumažinti naudojamus resursus buvo naudojamas podėliavimas (angl. *caching*). Šio proceso eiga pavaizduota sekos diagramoje (žr. 24 pav.).



24 pav. Podėliavimo procesas

Kiekvienas naujai tam tikrai kategorijai priskirtas tinklapis yra išsaugomas operatyvinėje atmintyje (RAM). Tokiu būdu, apsilankius tinklapyje dar kartą, nereikia pakartotinai siųsti užklausos į centrinį serverį ar iš naujo nustatyti tinklapio kategorijas.

3.7. Ugniasienės konfigūracija

Siekiant užkirsti kelią kuriamos tinklapių filtravimo sistemos turinio blokavimo išvengimui, buvo sukurtos ugniasienės konfigūravimo taisyklės (žr. 25-26 pav.).

```
config redirect 'REDIR_DNS'  
  option target 'DNAT'  
  option src 'lan'  
  option dest 'lan'  
  option proto 'tcp udp'  
  option name 'Redirect_DNS'  
  option src_dport '53'  
  option dest_ip '192.168.3.1'  
  option priority '16'  
  option dest_port '5353'  
  option enabled '1'
```

25 pav. DNS užklausų nukreipimas į sukonfigūruotą DNS serverį

```
config rule '26'  
  option src 'lan'  
  option name 'Block_VPN'  
  option priority '1'  
  option dest 'wan'  
  option target 'REJECT'  
  list proto 'all'  
  list src_port '1194'  
  list src_port '1723'  
  list src_port '1701'  
  list src_port '500'  
  list src_port '4500'  
  list src_port '2048'  
  option utc_time '0'  
  option enabled '1'
```

26 pav. VPN blokavimo taisyklė

Pirmiausia aprašytas DNS užklausų nukreipimas (žr. 25 pav.). Gavus užklausą iš LAN potinklio per 53 (numatytas DNS prievadas) prievadą, ji yra nukreipiama į maršrutizatoriaus IP adresą per 5353 prievadą (blokavimo paketo veikimo prievadas) – tokiu būdu yra ribojama prieiga prie išorinių DNS serverių. Kita taisykle ribojamas VPN serverių pasiekiamumas per sistemos naudotojų įrenginius (žr. 26 pav.).

3.8. Apibendrinimas

Prototipo realizacijai buvo naudojamas centrinis serveris ir „Teltonika RUTX14“ maršrutizatorius su įdiegta „RUTOS“ operacine sistema, sukurtu blokavimo paketu. Filtravimo sistemos duomenų rinkiniui sukurti buvo naudojamas internete prieinamas sąrašas (tinklapių domenai, jiems priskirtos kategorijos). Duomenų rinkinio generavimo įrankis buvo sukurtas naudojant „Python“ programavimo kalbą. Taip pat buvo sukurtas logistinės regresijos algoritmu ir *Bag of Words* (BoW) modeliu paremti mašininio mokymosi metodai, tinklapių skirstymo į kategorijas algoritmas. Siekiant paspartinti užklausų apdorojimą maršrutizatoriuje, sumažinti naudojamus resursus, buvo panaudotas podėliavimas, o norint užkirsti kelią kuriamos tinklapių filtravimo sistemos turinio blokavimo išvengimui, sukurtos ugniasienės konfigūravimo taisyklės.

4. Tinklapių filtravimo sistemos, skirtos maršrutizatoriams, tyrimas

Skyriuje aprašyto darbo tikslas – ištirti sukurtos tinklapių filtravimo sistemos veikimą ir jos efektyvumą. Šiam tikslui pasiekti buvo iškelti uždaviniai:

1. nustatyti modelio apmokymui naudojamų duomenų rinkinio kokybės įtaką sistemos veikimo efektyvumui;
2. įvertinti sukurtos sistemos ar jos dalies diegimo įterptinėje aplinkoje galimybę;
3. ištirti teksto apdorojimui naudojamų mašininio mokymosi algoritmų tinklapių skirstymo į kategorijas efektyvumą;
4. įvertinti sistemoje naudojamų mašininio mokymosi algoritmų ir jų derinių tinklapių skirstymo į kategorijas tikslumą ir greitaveiką.

4.1. Tyrimo aplinka

Sukurtos sistemos testavimui ir tyrimui vykdyti buvo naudojamas virtualus privatus serveris, kurio parametrai aprašyti 6 lentelėje.

6 lentelė. Sistemos realizacijai naudoto serverio parametrai

Parametras	Apibūdinimas
Serverio IP adresas	91.211.247.93
Operacinė sistema	Ubuntu-22.04, 64 bitų
Procesorius	1 branduolio (3GHz)
Operatyvinė atmintis (RAM)	1,3 GB
Talpa	15 GB
Tinklo pralaidumas	200 Mbps
Virtualizacijos platforma	Virtuozzo Linux 7

Virtualizacijai naudojama nemokama, atviro kodo platforma „Virtuozzo Linux 7“.

4.2. Duomenų rinkinio kokybės vertinimas

Vienas iš esminių aspektų, nuo kurių tiesiogiai priklauso prižiūrimu mašininio mokymo tipu paremtos sistemos veikimo efektyvumas, tai modelio apmokymui ir testavimui skirtų duomenų kokybė. Šiai priklausomybei ištirti buvo naudojamas tinklapių metaduomenų apdorojimu pagrįstas algoritmas. Pirmiausia buvo naudojamas 2015 metais sukurtas internete laisvai prieinamas sąrašas, sudarytas iš 31086 įrašų anglų kalba [47]. Šiame duomenų rinkinyje talpinami URL adresai suskirstyti į dvidešimt penkias kategorijas (žr. 7 lentelę). Kiekvienam iš URL adresų buvo priskirtas pasiklovimo (angl. *confidence*) koeficientas, kuris numatė, kiek tikslu yra tam tikram tinklapiui priskirta kategorija.

7 lentelė. Duomenų rinkinyje Nr. 1 esančios tinklapių kategorijos

Kategorijos pavadinimas anglų k.	Kategorijos pavadinimas lietuvių k.
Career_and_Education	Karjera ir išsilavinimas
Games	Žaidimai
Pets_and_Animals	Augintiniai ir gyvūnai
Food_and_Drink	Maistas ir gėrimai
Business_and_Industry	Verslas ir pramonė
Health	Sveikata
Shopping	Apsipirkimas
Recreation_and_Hobbies	Poilsis ir hobiai
Books_and_Literature	Knygos ir literatūra
Autos_and_Vehicles	Automobiliai ir transporto priemonės
Arts_and_Entertainment	Menas ir pramogos
Home_and_Garden	Namai ir sodas
People_and_Society	Žmonės ir visuomenė
Adult	Suaugusiems
Beauty_and_Fitness	Grožis ir fitnesas
Internet_and_Telecom	Internetas ir telekomunikacijos
Finance	Finansai
Computer_and_Electronics	Kompiuteriai ir elektronika
Sports	Sportas
Science	Mokslas
Law_and_Government	Teisė ir vyriausybė
News_and_Media	Naujienos ir žiniasklaida
Gambling	Lošimai
Reference	Nuorodos
Travel	Kelionės

Siekiant įvertinti sukurtos tinklapių skirstymo į kategorijas sistemos veikimą, bus apskaičiuojami pagrindiniai tą atspindintys kokybiniai rodikliai – tikslumas (angl. *Accuracy*), preciziškumas (angl. *Precision*), atšaukimas (angl. *Recall*), F1 balas (angl. *F1 score*) (žr. atitinkamai 4-7 formulę) [48-49].

$$Tikslumas = \frac{TP+TN}{TP+TN+FP+FN}; \quad (4)$$

$$Preciziškumas = \frac{TP}{TP+FP}; \quad (5)$$

$$Atšaukimas = \frac{TP}{TP+FN}; \quad (6)$$

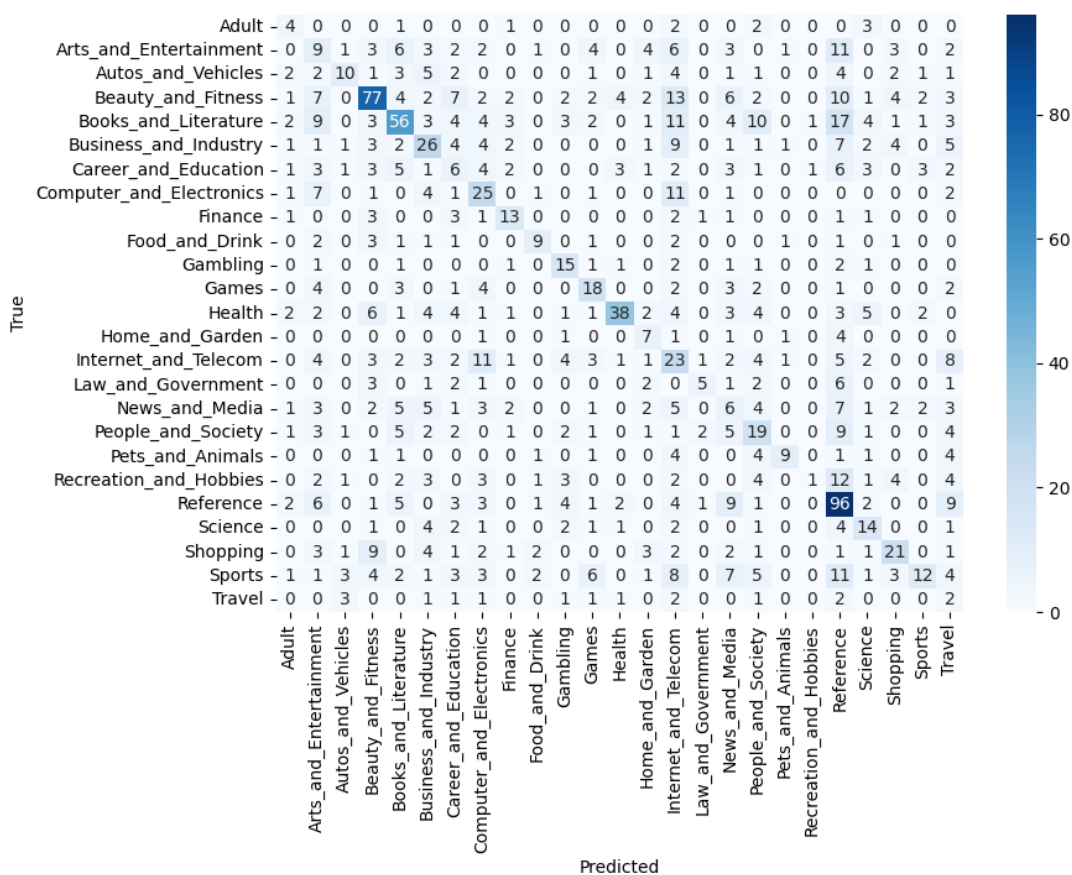
$$F1 \text{ balas} = 2 \left(\frac{Preciziškumas \cdot Atšaukimas}{Preciziškumas + Atšaukimas} \right). \quad (7)$$

Šių rodiklių skaičiavimui naudojami keturias skirtingas baigtis atspindintys parametrai [48-49]:

- pozityvi tiesa (angl. *True Positive, TP*), pavyzdžiui, kai žaidimų tinklapis buvo priskirtas žaidimų kategorijai;
- negatyvi tiesa (angl. *True Negative, TN*), pavyzdžiui, kai ne žaidimų tinklapis nebuvo priskirtas žaidimų kategorijai;
- pozityvi netiesa (angl. *False Positive, FP*), pavyzdžiui, kai ne žaidimų tinklapis buvo priskirtas žaidimų kategorijai;
- negatyvi netiesa (angl. *False Negative, FN*), pavyzdžiui, kai žaidimų tinklapis buvo priskirtas ne žaidimų kategorijai.

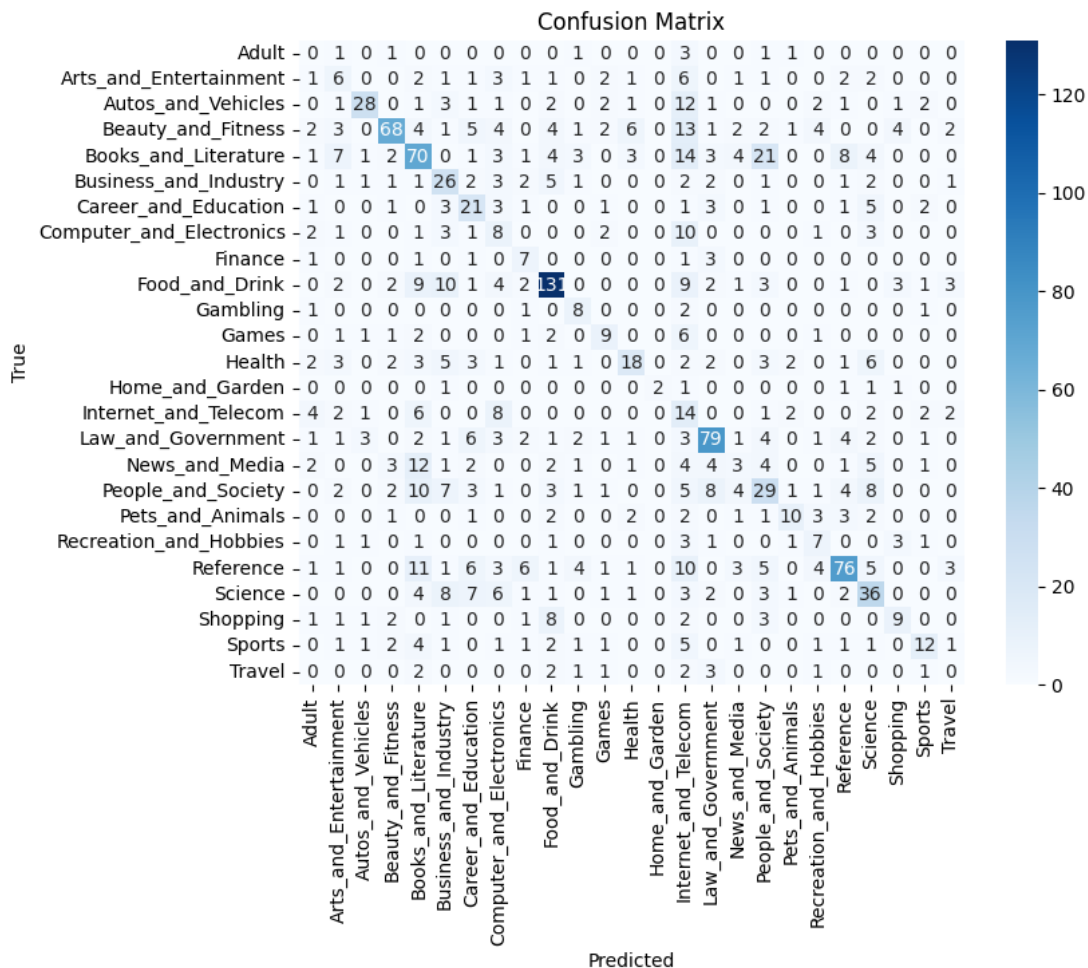
Šios baigtys, taip pat ir 4-7 formulėse pateikti rodikliai, taikomos atskirai kiekvienai tinklapių kategorijai, o vėliau yra apskaičiuojama vidutinė reikšmė. Šiame rinkinyje esantys duomenys visais testavimo atvejais buvo suskirstyti į dvi dalis – modelio apmokymui ir testavimui, atitinkamai, 80 % ir 20 %, o vizualiam rezultatų pateikimui naudojama painiavos matrica.

Peržiūrėjus duomenų rinkinį buvo pašalinti neveikiančių tinklapių adresai ir jiems priskirtos kategorijos – duomenų kiekis sumažėjo daugiau nei pusiau iki maždaug 15 tūkstančių įrašų. Šiuo atveju algoritmo veikimo tikslumas siekė 0,36. 27 pav. pateiktoje painiavos matricoje galima matyti, jog daugiausia teisingai priskirtų tinklapių priklauso kategorijoms „Nuorodos“ bei „Grožis ir fitnessas“.



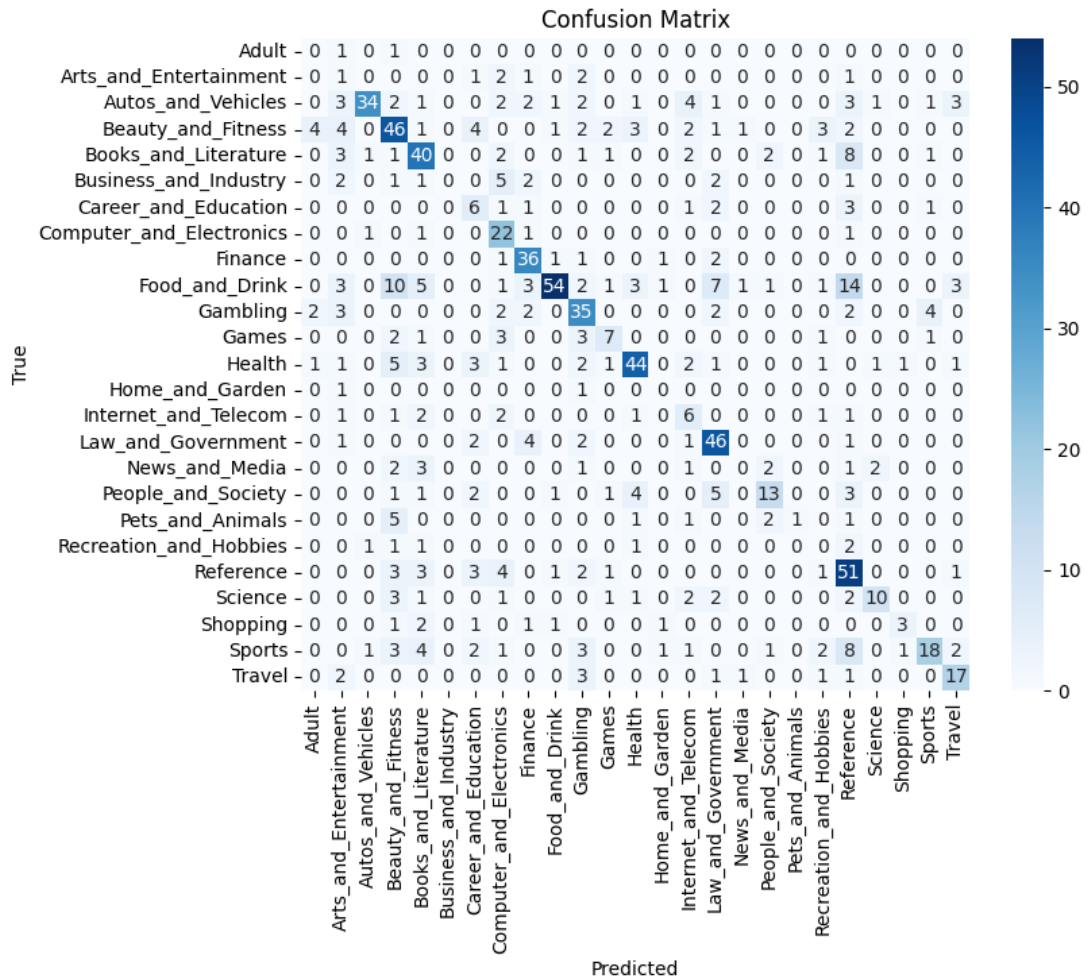
27 pav. Painiavos matrica, sudaryta naudojant duomenų rinkinį Nr. 1, iš kurio pašalinti tik neaktyvūs tinklapiai

Tuomet iš rinkinio buvo pašalinti tinklapiai, kurių pasikliovimo koeficientas buvo mažesnis nei 0,6. Įrašų kiekis siekė apie 12,2 tūkstančio, o algoritmo veikimo tikslumas – 0,47. Pateiktoje painiavos matricoje matyti, kad daugiausia teisingai priskirtų tinklapių priklauso „Maisto ir gėrimų“ kategorijai (žr. 28 pav.).



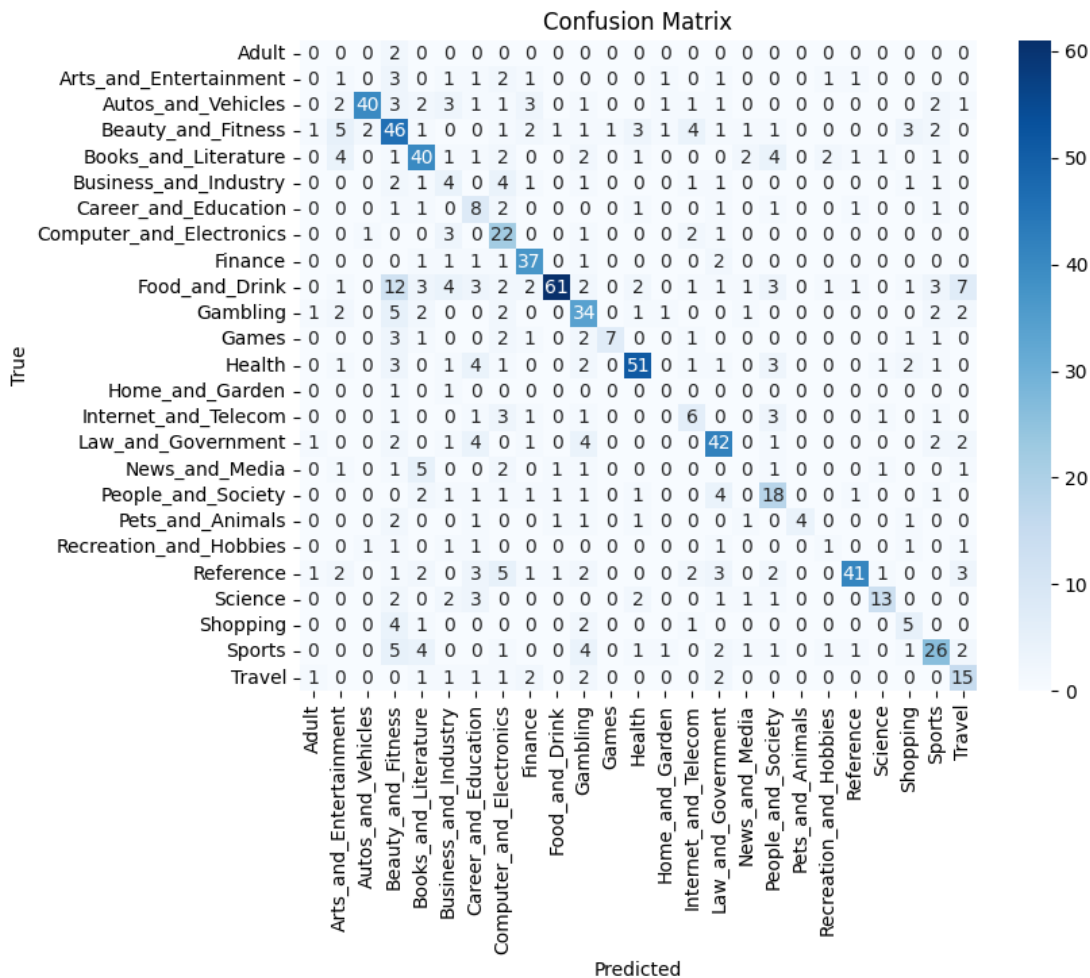
28 pav. Painiavos matrica, sudaryta naudojant duomenų rinkinį Nr. 1, iš kurio pašalinti neaktyvūs tinklapiai ir tinklapiai, kurių pasikliovimo koeficientas buvo mažesnis nei 0,6

Iš duomenų rinkinio pašalinus tinklapius, kurių pasiklojimo koeficientas mažesnis nei 0,8, įrašų kiekis nukrito iki 6,4 tūkstančio, o algoritmo skirstymo į kategorijas tikslumas siekė 0,56. Gautus rezultatus lyginant su anksčiau sugeneruotomis painiavos matricomis galima pastebėti, kad didinant tinklapių priskyrimo tam tikrai kategorijai pasiklojimo koeficientą, vis didesnė matricos duomenų dalis susitelkia įstrižainėje iš kairės į dešinę (žr. 29 pav.).



29 pav. Painiavos matrica, sudaryta naudojant duomenų rinkinį Nr. 1, iš kurio pašalinti neaktyvūs tinklapiai ir tinklapiai, kurių pasiklojimo koeficientas buvo mažesnis nei 0,8

Duomenų rinkinyje palikus tik tuos tinklapius, kurių pasiklovimo koeficientas buvo lygus 1, įrašų kiekis siekė 5,7 tūkstančio. Nors toks duomenų rinkinys turėtų būti gana patikimas, tačiau, lyginant su tikslumu, kuris buvo pasiektas naudojant duomenis su pasiklovimo koeficientu didesniu nei 0,8, algoritmo skirstymo į kategorijas tikslumas išaugo tik mažiau nei vienu procentu ir siekė 0,57 (žr. 30 pav.).



30 pav. Painiavos matrica, sudaryta naudojant duomenų rinkinį Nr. 1 su tinklapiais, kurių pasiklovimo koeficientas lygus 1

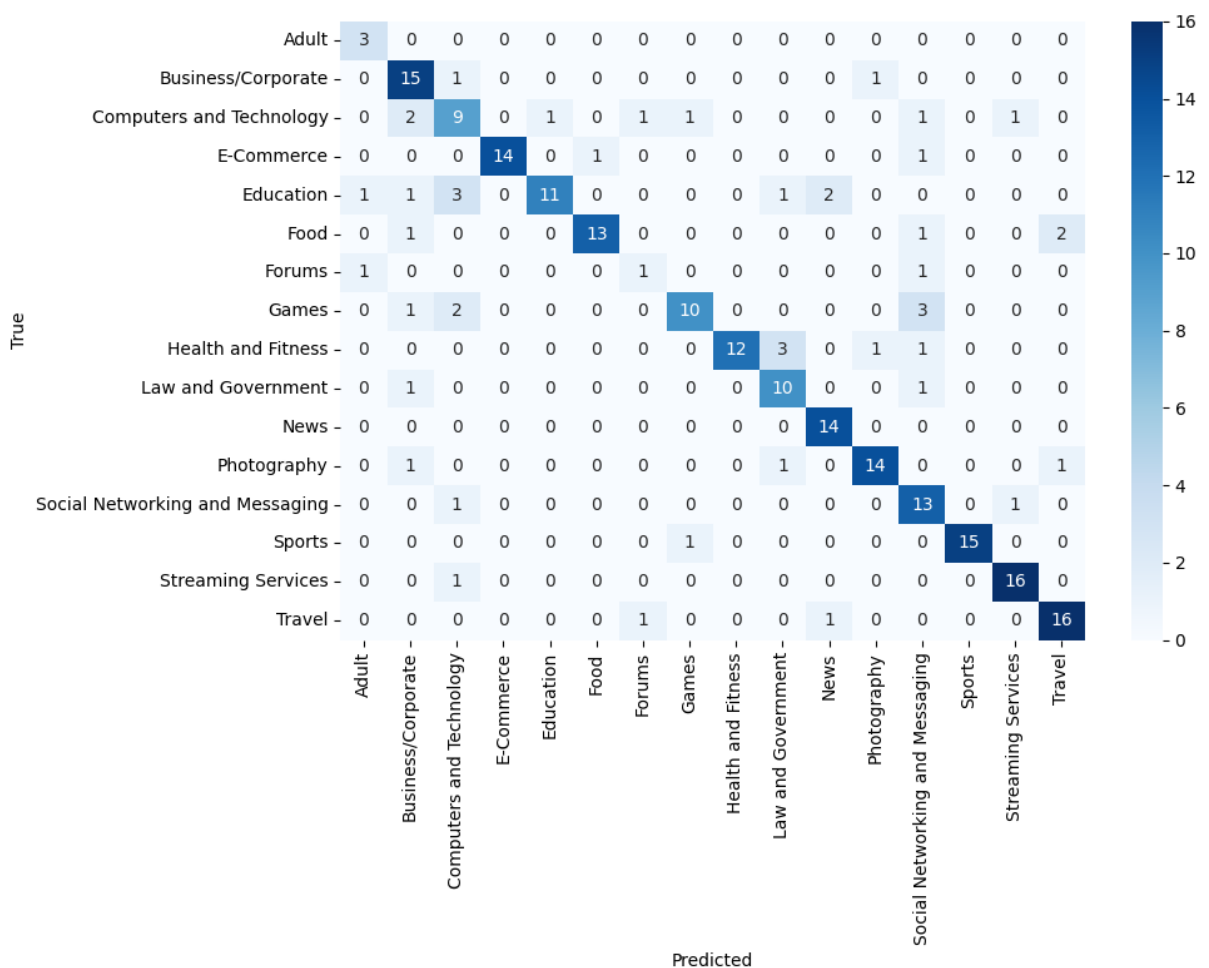
Atlikto eksperimento rezultatai privertė suabejoti duomenų rinkinio patikimumu, todėl buvo pasirinktas kitas duomenų rinkinys. Tolesniam sistemos testavimui buvo naudojamas prieš trejus metus sudarytas sąrašas, talpinantis 1407 įrašus [50]. Šiame duomenų rinkinyje esantys URL adresai suskirstyti į penkioliką kategorijų (žr. 8 lentelę).

8 lentelė. Duomenų rinkinyje Nr. 2 esančios tinklapių kategorijos

Tinklapių kategorija anglų k.	Tinklapių kategorija lietuvių k.
Travel	Kelionės
Social	Socialiniai tinklai
Streaming	Srautinis perdavimas
Sports	Sportas

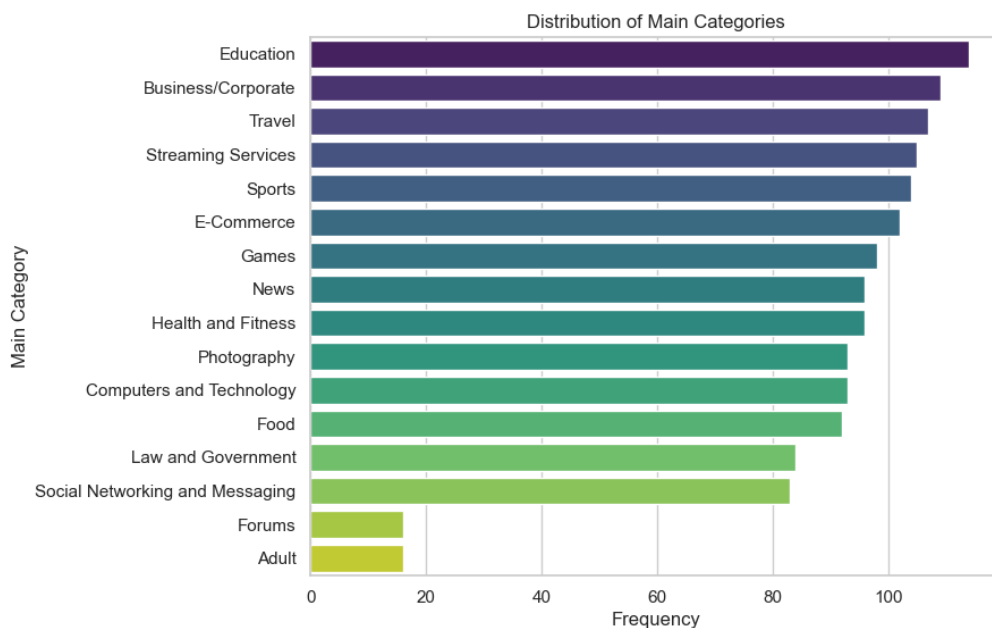
News	Naujienos
Photography	Fotografija
Law	Teisė
Health	Sveikata
Games	Žaidimai
E-Commerce	El. prekyba
Forums	Forumai
Food	Maistas
Education	Mokslas
Computers	Kompiuteriai
Business/Corporate	Verslas/įmonės
Adult	Suaugusiems

31 pav. pateikta painiavos matrica atskleidžia, kad tinklapių priskirtų forumų ir suaugusiems skirtu turinio kategorijoms skaičius yra žymiai mažesnis nei kitoms. Bendras tinklapių skirstymo į kategorijas veikimo tikslumas siekė 0,79.



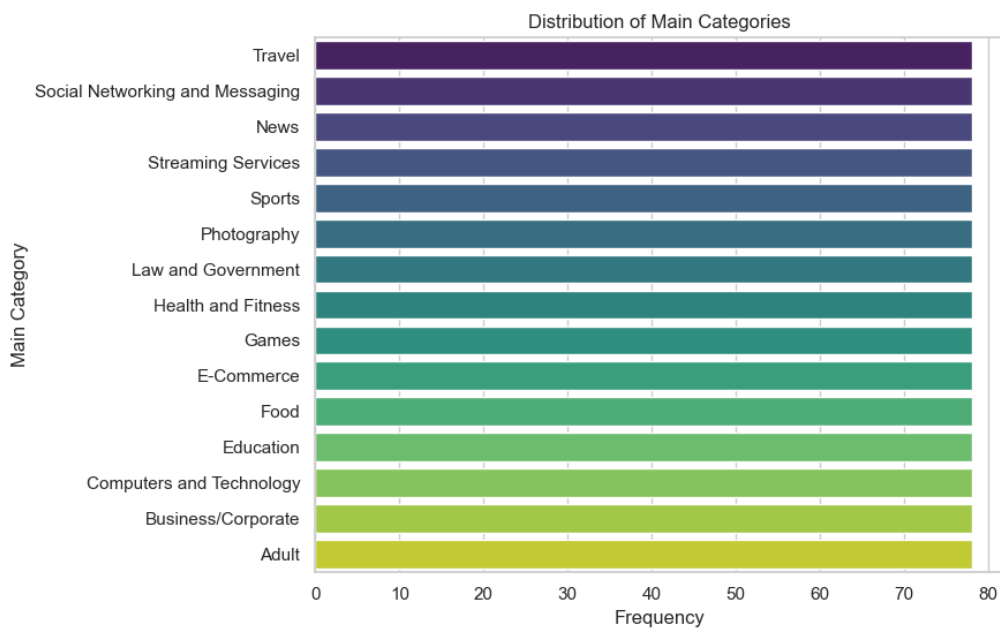
31 pav. Painiavos matrica, sudaryta naudojant duomenų rinkinį Nr. 2

Siekiant padidinti šio kokybinio parametro vertę buvo nuspręsta peržiūrėti naudojamo duomenų rinkinio įrašų pasiskirstymą tarp kategorijų (žr. 32 pav.).



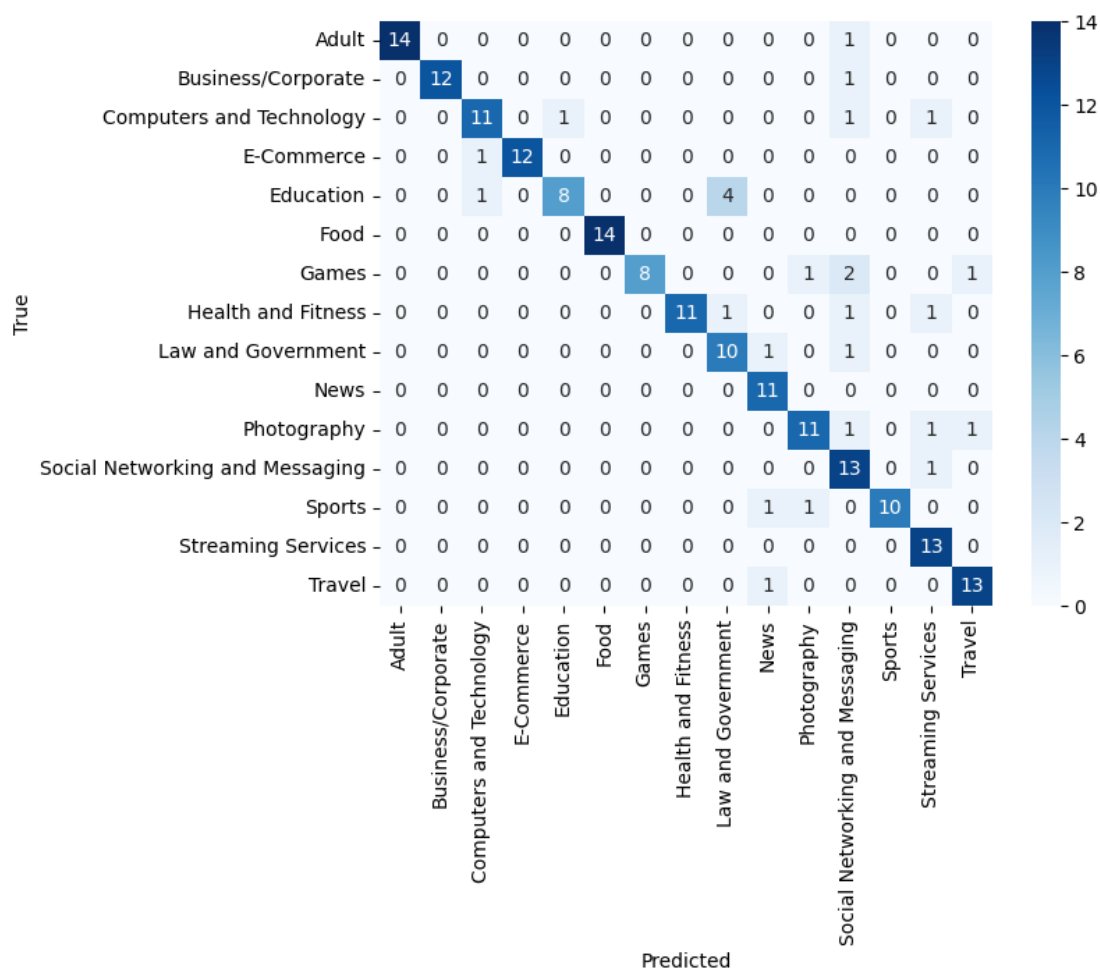
32 pav. Duomenų rinkinio Nr. 2 įrašų pasiskirstymas tarp kategorijų

Iš paveikslėlyje pateiktų duomenų matyti, jog dviem iš kategorijų – forumų ir skirtos suaugusiems – yra priskirta mažiau nei 20 tinklapių, kai kitų sąrašuose yra tarp 80 ir 120. Kadangi toks netolygus tinklapių pasiskirstymas modelio apmokymui skirtuose duomenyse gali daryti neigiamą įtaką visos sistemos efektyvumui, buvo atliktas duomenų failo koregavimas. Forumų ir suaugusiems skirtos kategorijos rankiniu būdu papildytos atitinkamais tinklapiais. Duomenų rinkinys apdorotas taip, kad kiekvienai iš jame esančių kategorijų būtų priskirtas toks pat kiekis tinklapių (žr. 33 pav.). Ši duomenų rinkinį dabar sudaro 1477 įrašai.



33 pav. Duomenų rinkinio Nr. 2 įrašų pasiskirstymas po rankinio koregavimo

34 pav. pateikiama painiavos matrica, sudaryta naudojant duomenų rinkinį Nr. 2, po rankinio koregavimo. Šiuo atveju tinklapių skirstymo į kategorijas algoritmo veikimo tikslumas siekė 0,86.



34 pav. Painiavos matrica, sudaryta naudojant duomenų rinkinį Nr. 2, po rankinio koregavimo

Šiuos rezultatus lyginant su 31 pav. pateiktais duomenimis, galima pastebėti, jog atlikus rankinį duomenų rinkinio įrašų koregavimą, tinklapių, priskirtų teisingoms kategorijoms, skaičius pasiskirstė tolygiau, o tinklapių skirstymo į kategorijas veikimo tikslumas buvo pagerintas apie 6,5 % – nuo 0,79 iki 0,86. Duomenų rinkinio Nr. 1 atveju, atlikus jo filtravimą ir rinkinyje palikus tik tuos įrašus, kurių pasiklovimo koeficientas buvo lygus 1, maksimalus skirstymo į kategorijas tikslumas siekė vos 0,57. Taigi, atliktas eksperimentas patvirtino, jog algoritmo veikimo efektyvumas tiesiogiai priklauso nuo modelio apmokymui naudojamo duomenų rinkinio kokybės.

4.3. Sistemos diegimo įterptinėje sistemoje galimybės vertinimas

Siekiant įvertinti kuriamos sistemos diegimo įterptinėje sistemoje galimybę, buvo atliktas bandymas, kurio metu tinklapio metaduomenų apdorojimu pagrįstas algoritmas veikė maršrutizatoriuje ir centriniame serveryje. Šiais laikais vienas iš vartotojams svarbiausių naršymo internete aspektų yra užklausų apdorojimo laikas, todėl šio eksperimento metu bus lyginama maršrutizatoriuje bei centriniame serveryje apdorojamų užklausų trukmė. 9 lentelėje pateikiami pagrindiniai bandymui naudojamo „Teltonika RUTX14“ maršrutizatoriaus parametrai.

9 lentelė. Bandymui naudoto maršrutizatoriaus parametrai

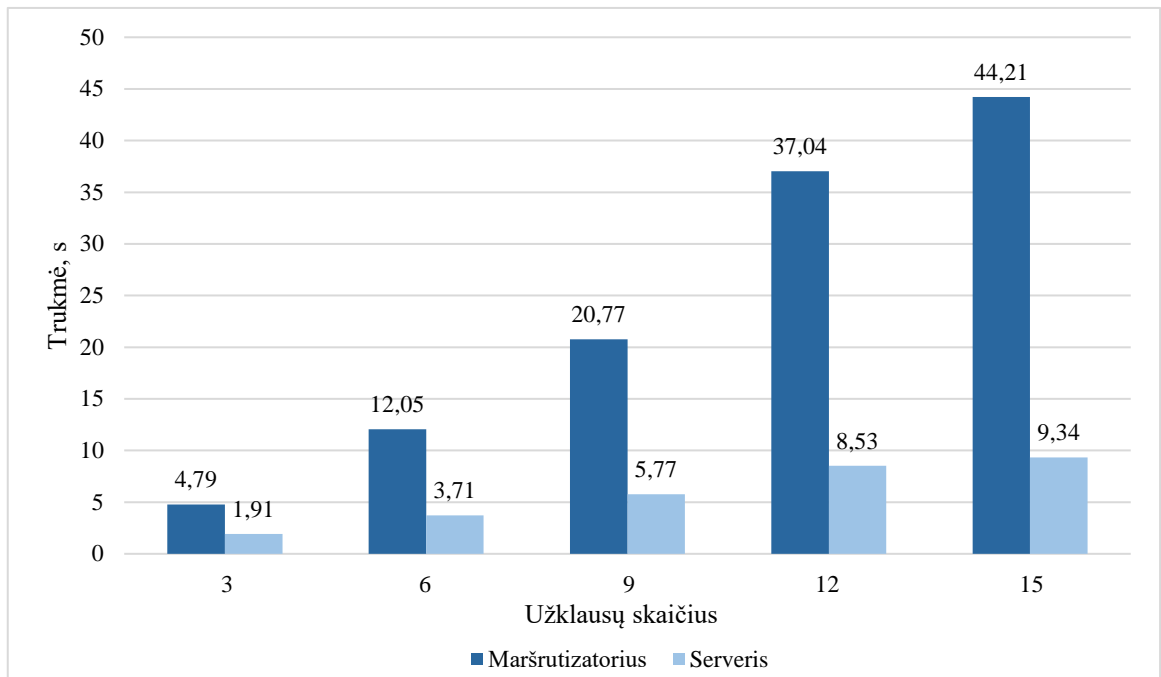
Parametras	Apibūdinimas
Operacinė sistema	Modifikuota atvirojo kodo <i>OpenWRT</i> sistema „RUTOS“
Procesorius	Quad-core ARM Cortex A7, 717 MHz
Operatyvinė atmintis (RAM)	256 MB

Eksperimento įvykdymui buvo sudarytas penkiolikos atsitiktinių tinklapių sąrašas (žr. 10 lentelę). 10 lentelėje pateikiami bandymo rezultatai.

10 lentelė. Tinklapių priskyrimo kategorijai trukmė maršrutizatoriuje ir centriname serveryje

Tinklapis	Tinklapių priskyrimo kategorijai trukmė, s	
	Maršrutizatoriuje	Serveryje
https://youtube.com	2,38	0,71
https://facebook.com	3,11	1,1
https://twitter.com	1,81	0,17
https://wikipedia.org	3,74	0,79
https://yahoo.com	4,15	1,66
https://whatsapp.com	4,29	0,79
https://tiktok.com	2,69	0,8
https://amazon.com	3,34	1,37
https://reddit.com	7,30	1,4
https://pornhub.com	10,20	0,96
https://live.com	10,96	1,78
https:// xnxx.com	9,30	1,45
https://linkedin.com	7,58	1,54
https://openai.com	10,22	1,39
https://netflix.com	8,60	1,03
Vidurkis	5,98	1,13

Gauti duomenys rodo, kad maršrutizatoriuje vykdomas tinklapių priskyrimo tam tikrai kategorijai procesas užtrunka vidutiniškai apie 4 kartus ilgiau nei centriname serveryje. Taip pat buvo iširta ir tinklapių priskyrimo kategorijai trukmės priklausomybė nuo tuo pačiu metu siunčiamų užklausų kiekio (žr. 35 pav.).



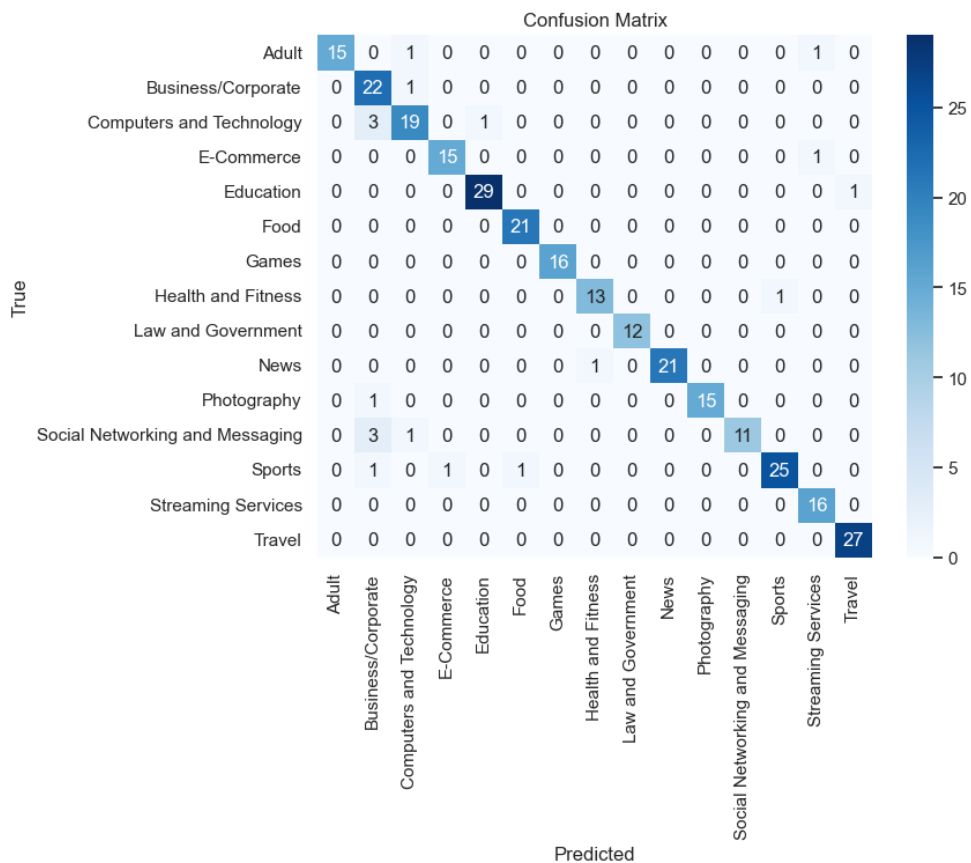
35 pav. Tinklapio priskyrimo kategorijai trukmės priklausomybė nuo siunčiamų užklausų kiekio

Rezultatai patvirtino, jog centriniame serveryje atliekamas tinklapio priskyrimas tam tikrai kategorijai vyksta sparčiau nei maršrutizatoriuje. Pavyzdžiui, apdorojant šešias užklausas tuo pačiu metu, centriniame serveryje šis procesas užtrunka 3,71 s, o maršrutizatoriuje – 12,05 s. Eksperimento metu taip pat buvo pastebėta, kad didėjant vienu metu siunčiamų užklausų kiekiui, skirtumas tarp šio proceso įvykdymo laiko serveryje ir maršrutizatoriuje taip pat auga – trijų užklausų apdorojimas maršrutizatoriuje trunka apie 2,5 karto ilgiau, o penkiolikos – 4,7 karto.

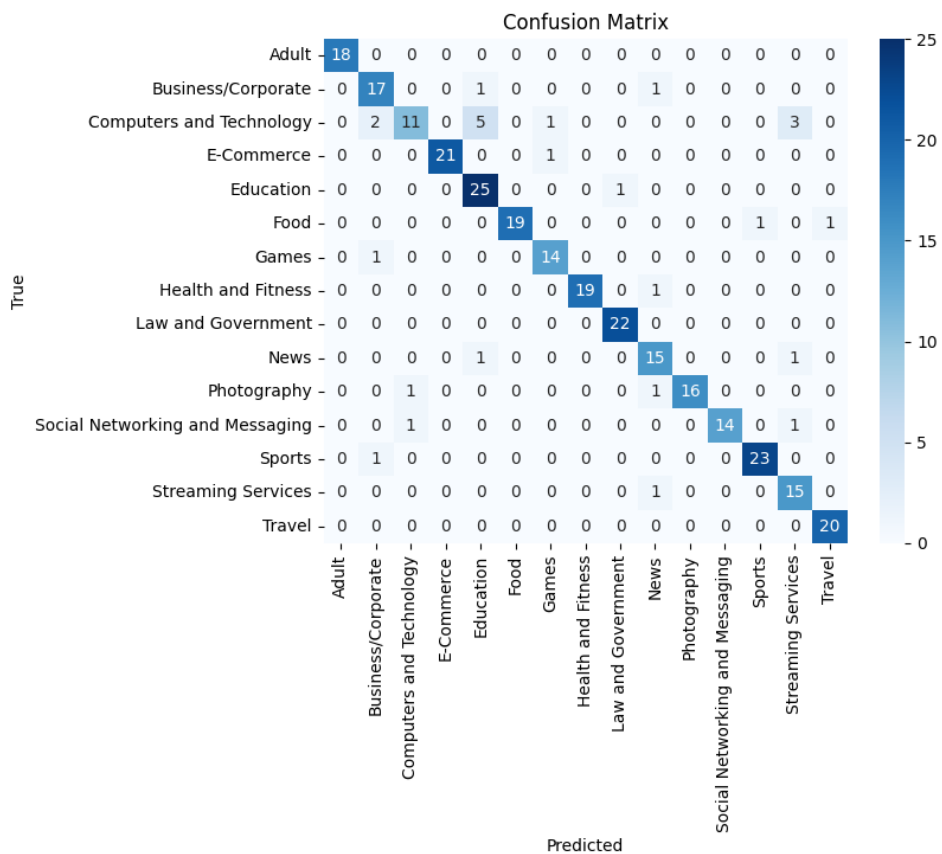
Taigi, atliktų eksperimentų rezultatai parodė, jog sukurtos tinklapių skirstymo į kategorijas sistemos diegimas maršrutizatoriuje yra įmanomas, tačiau toks sprendimas kardinaliai sumažintų jos veikimo sklandumą ir efektyvumą.

4.4. Teksto apdorojimui naudojamų mašininio mokymosi algoritmų efektyvumo tyrimas

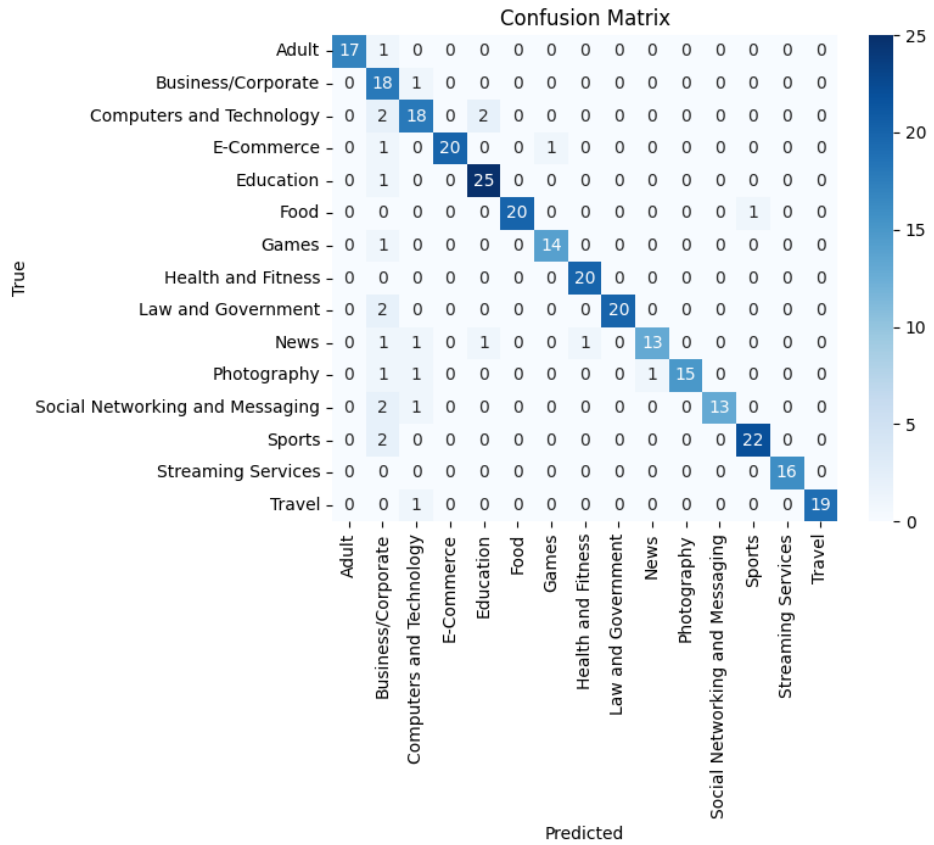
Siekiant išrinkti optimaliomis savybėmis pasižymintį teksto apdorojimui tinkamą mašininio mokymosi algoritmą, bus palyginti keturi – logistinės regresijos, Naiviojo Bajeso, SVM ir atsitiktinio miško – metodai. Eksperimentui bus naudojamas anksčiau minėtas duomenų rinkinys Nr. 2 po rankinio koregavimo (1477 įrašai). Rinkinyje esantys duomenys buvo suskirstyti į dvi dalis – modelio apmokymui ir testavimui, atitinkamai, 80 % ir 20 %. 36-39 pav. pateikiamos painiavos matricos, sudarytos naudojant minėtus metodus.



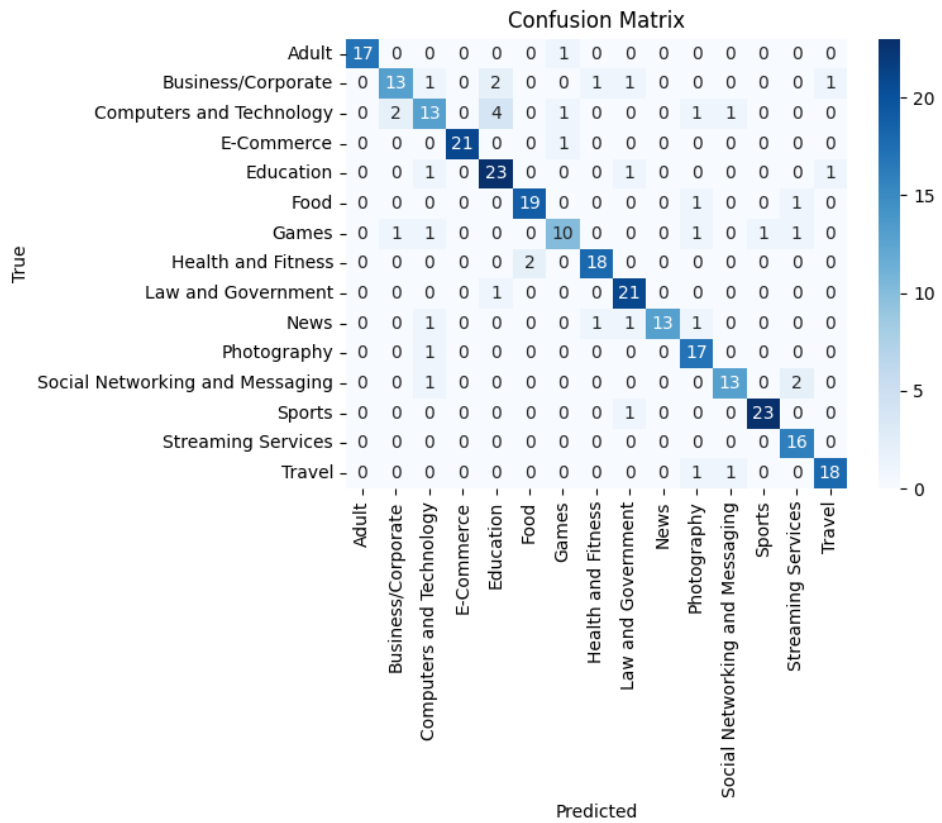
36 pav. Painiavos matrica, sudaryta tekstinę informaciją apdorojant logistine regresija



37 pav. Painiavos matrica, sudaryta tekstinę informaciją apdorojant Naiviojo Bajeso algoritmu



38 pav. Painiavos matrica, sudaryta tekstinę informaciją apdorojant SVM algoritmu



39 pav. Painiavos matrica, sudaryta tekstinę informaciją apdorojant atsitiktinio miško algoritmu

Bandymo metu buvo apskaičiuoti pagrindiniai kokybiniai rodikliai, tokie kaip tikslumas, preciziškumas, atšaukimas, F1 balas. Taip pat užfiksuota ir pradinio apsimokymo, testavimo etapo bei vidutinė užklauso apdorojimo trukmė (žr. 11 lentelę). Šių užklauso generavimui buvo naudoti 10 lentelėje pateikti tinklapiai.

11 lentelė. Teksto apdorojimui naudojamų mašininio mokymosi algoritmų efektyvumo parametrai

Parametras	Metodas			
	Logistinės regresijos	Naiviojo Bajeso	SVM	Atsitiktinio miško
Tikslumas (angl. Accuracy)	0,936	0,909	0,912	0,861
Preciziškumas (angl. Precision)	0,942	0,915	0,932	0,865
Atšaukimas (angl. Recall)	0,936	0,909	0,912	0,861
F1 balas (angl. F1 score)	0,936	0,906	0,917	0,860
Apsimokymo etapo trukmė, s	0,82	1,62	0,64	2,05
Testavimo etapo trukmė, s	0,11	0,43	0,18	0,14
Užklauso apdorojimo vidutinė trukmė, s	2,09	2,13	2,25	2,16

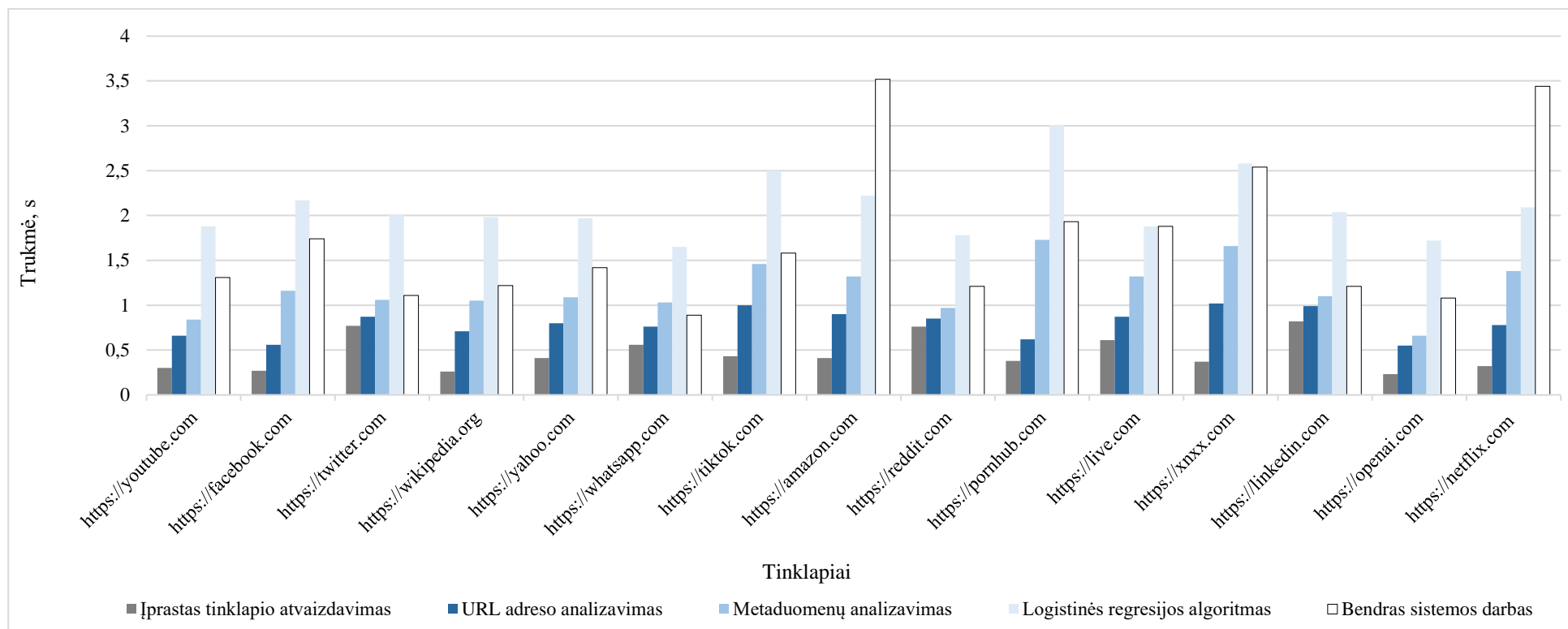
Gauti rezultatai atskleidžia, kad tirtų algoritmų tinklapių priskyrimo tam tikrai kategorijai tikslumas svyravo nuo 0,861 iki 0,936. Vertinant pirminio mašininio mokymo algoritmo modelio sudarymą pastebėta, kad mažiausia apsimokymo etapo trukmė (0,64 s) pasižymėjo SVM, o didžiausia (2,05 s) – atsitiktinio miško metodas. Logistine regresija pagrįsto metodo testavimas užtruko trumpiausiai – 0,11 s, o Naiviojo Bajeso – ilgiausiai – 0,43 s. Vidutinės užklauso apdorojimo trukmės vertės kito nuo 2,09 (logistinės regresijos algoritmas) iki 2,25 s (SVM algoritmas). Logistinės regresijos metodas nors ir nežymiu skirtumu, tačiau išsiskyrė mažiausia trukme – 2,09 s – ir aukščiausia tinklapių skirstymo į kategorijas verte – 0,946.

Taigi, atliktas bandymas įrodė, kad logistinės regresijos algoritmas pasižymi geriausiomis savybėmis ir jo taikymas kuriamoje sistemoje prisidėtų prie bendro sistemos darbo efektyvumo didinimo.

4.5. Sistemoje naudojamų mašininio mokymosi algoritmų bei jų derinių veikimo tikslumo ir greitaveikos tyrimas

Svarbu tai, jog sukurta sistema ne tik efektyviai atliktų pagrindinę savo funkciją – priskirtų tinklapius esamoms kategorijoms ir ribotą prieigą prie nepageidaujamo turinio – bet ir apdorotų užklausas per tam tikrą laiką. Vienas iš iškeltų kuriamos sistemos nefunkcinių reikalavimų nusako, kad tinklapių kategorijos nustatymo procesas negali sumažinti vartotojo naršymo greičio daugiau nei 5 kartus arba bendra vidutinė užklauso apdorojimo trukmė negali viršyti 2,5 s. Taigi, šiame skyriuje aprašomo bandymo tikslas – ištirti sukurto tinklapių skirstymo į kategorijas sistemos greitaveiką ir patikrinti pasirinkto mašininio mokymosi algoritmų derinį.

40 pav. pavaizduota tinklapių priskyrimo tam tikrai kategorijai proceso vykdymo trukmės priklausomybė nuo naudojamų mašininio mokymosi algoritmų bei jų derinių. Taip pat buvo apskaičiuotos ir vidutinės tinklapių apdorojimo trukmės vertės (žr. 12 lentelę).



40 pav. Tinklapių priskyrimo kategorijai trukmės priklausomybė nuo naudojamų mašininio mokymosi algoritmų bei jų derinių

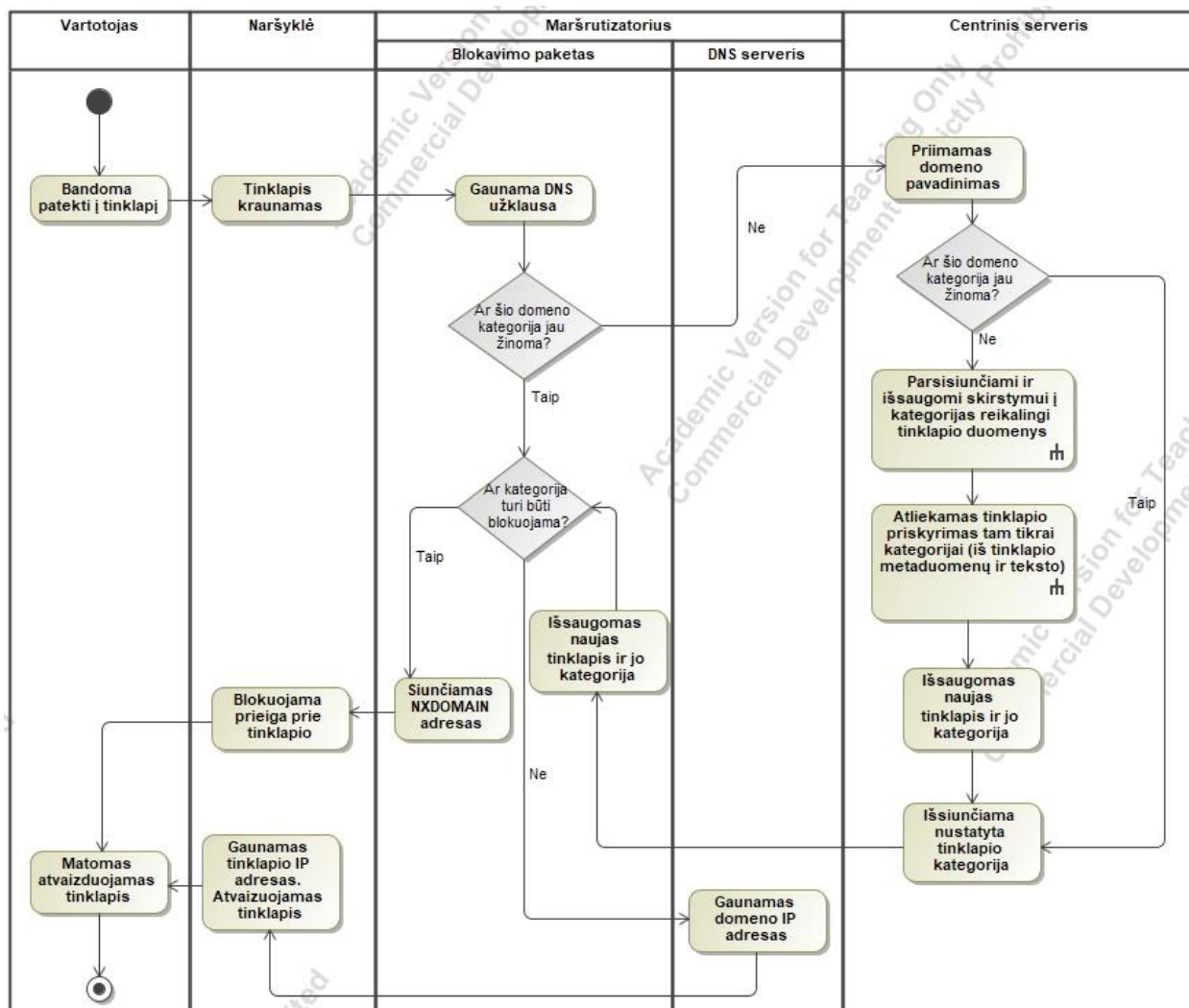
12 lentelė. Kuriamoje sistemoje naudojamų algoritmų veikimo tikslumas ir vidutinė tinklapių apdorojimo trukmė

Parametras	Įprastas tinklapio atvaizdavimas	URL adreso analizavimo algoritmas	Metaduomenų analizavimo algoritmas	Turinio analizės logistinės regresijos algoritmas	Bendras sistemos darbas
Tikslumas (angl. Accuracy)	-	0,534	0,863	0,936	0,778
Užklauso apdorojimo vidutinė trukmė, s	0,46	0,80	1,19	2,09	1,74

Pirminė kuriamos sistemos veikimo idėja buvo mašininio mokymosi algoritmų derinys – visų pirma, būtų vykdomas tinklapio priskyrimas kategorijai iš URL adreso ir metaduomenų. Jei šiais dviem metodais nustatytos kategorijos sutampa, toliau, atsižvelgiant nuo to, tinklapiui priskirsta kategorija yra draudžiama ar leidžiama, seks tinklapio blokavimas arba atvaizdavimas vartotojui. Jei atlikus URL adreso ir metaduomenų analizavimą, nustatytos kategorijos nesutaps, bus atliekamas tinklapio priskyrimas kategorijai iš tinklapio teksto, naudojant logistinės regresijos algoritmą. Tačiau atlikus naudojamų mašininio mokymosi algoritmų bei jų derinių veikimo tikslumo ir greitaveikos tyrimą paaiškėjo, kad nors tokios sudėtinės sistemos užklausų apdorojimo vidutinė trukmė siektų 1,74 s, joje naudojamas URL adreso analizavimo algoritmas sumažintų bendrą sistemos tinklapių skirstymo į kategorijas tikslumą maždaug 15-20 %.

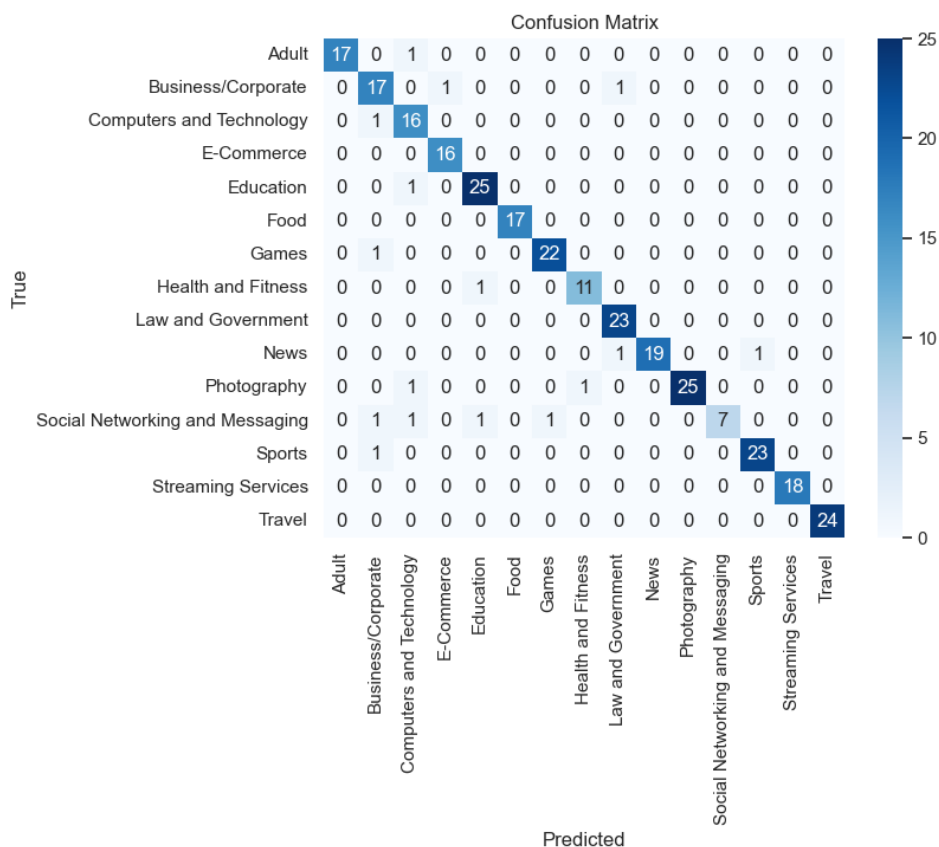
4.6. Patobulinta tinklapių skirstymo į kategorijas sistema

Atlikti tyrimai parodė, jog URL adreso analizavimo naudojimas kuriamoje sistemoje nedidina jos veikimo efektyvumo, todėl buvo nuspręsta tinklapių skirstymui į kategorijas naudoti kitokią algoritmų derinį – itin dideliu tikslumu pasižyminčiu logistinės regresijos algoritmu apdoroti ne tik tinklapio tekstinę informaciją, bet ir metaduomenis. 41 pav. pateikta patobulinto tinklapių skirstymo į kategorijas sistemos algoritmo veiklos diagrama.



41 pav. Patobulinto tinklapių skirstymo į kategorijas sistemos algoritmo veiklos diagrama

42 pav. pateikiama painiavos matrica, sudarytos naudojant minėtą metodą.



42 pav. Painiavos matrica, sudaryta metaduomenis ir tekstinę informaciją apdorojant logistine regresija

Siekiant įvertinti pasirinkto metodo efektyvumą buvo apskaičiuoti pagrindiniai kokybiniai rodikliai, pradinio apsimokymo, testavimo etapo, vidutinė užklauso apdorojimo trukmė (žr. 13 lentelę). Užklauso generavimui buvo naudoti tinklapiai, pateikti 10 lentelėje.

13 lentelė. Tinklapiu turinio ir metaduomenų apdorojimo logistine regresija efektyvumas

Parametras	Vertė
Tikslumas (angl. Accuracy)	0,946
Preciziškumas (angl. Precision)	0,950
Atšaukimas (angl. Recall)	0,946
F1 balas (angl. F1 score)	0,946
Apsimokymo etapo trukmė, s	0,86
Testavimo etapo trukmė, s	0,12
Užklauso apdorojimo vidutinė trukmė, s	2,13

Lentelėje pateikti duomenys rodo, kad pasirinkto algoritmo tikslumo vertė (0,946) yra apie 18 % didesnė nei pirminės sistemos (0,778) (žr. 12 lentelę). Nors vidutinė užklauso apdorojimo trukmė ir išaugo nuo 1,77 s iki 2,13 s, tačiau šis pokytis nėra žymus. Svarbu paminėti ir tai, jog sistema atitiktų anksčiau iškeltą nefunkcinį reikalavimą, nes vartotojo naršymo greitis nebūtų sumažinamas daugiau nei 5 kartus ir vidutinė užklauso apdorojimo trukmė neviršytų 2,5 s.

4.7. Tyrimo išvados

Atlikti tyrimai atskleidė, kad algoritmo veikimo efektyvumas tiesiogiai priklauso nuo modelio apmokymui naudojamo duomenų rinkinio kokybės – pasirinkus atsitiktinį duomenų rinkinį, maksimalus gautas tinklapių skirstymo į kategorijas tikslumas buvo lygus 0,57. Pakeitus duomenų rinkinį ir pakoregavus jį rankiniu būdu, tinklapių skirstymo į kategorijas veikimo tikslumas siekė 0,86. Įvykdžius sistemos diegimo įterptinėje sistemoje galimybes vertinimą paaiškėjo, kad sukurtos sistemos diegimas maršrutizatoriuje yra įmanomas, tačiau tinklapių priskyrimo tam tikrai kategorijai procesas užtrunka maždaug 4 kartus ilgiau nei centriniame serveryje, taigi, toks sprendimas kardinaliai sumažintų jos veikimo sklandumą ir efektyvumą. Tekstinės informacijos apdorojimo bandymo rezultatai atskleidė, kad tirtų algoritmų tinklapių priskyrimo tam tikrai kategorijai tikslumas svyravo nuo 0,861 iki 0,946. Vertinant pirminio mašininio mokymo algoritmo modelio sudarymą pastebėta, kad mažiausia apsimokymo etapo trukmė (0,636 s) pasižymėjo SVM, o didžiausia (2,053 s) – atsitiktinio miško metodas. Logistine regresija pagrįsto metodo testavimas užtruko trumpiausiai – 0,11 s, o Naiviojo Bajeso – ilgiausiai – 0,43 s. Vidutinės užklauso apdorojimo trukmės vertės kito nuo 2,09 (logistinės regresijos algoritmas) iki 2,25 s (SVM algoritmas). Logistinės regresijos metodas išsiskyrė mažiausia trukme – 2,09 s – ir aukščiausia tinklapių skirstymo į kategorijas verte – 0,946, taigi, jo taikymas kuriamoje sistemoje prisidėtų prie bendro sistemos darbo efektyvumo didinimo. Pirminė kuriamos sistemos veikimo idėja buvo mašininio mokymosi algoritmų derinys – pirmiausia būtų tinklapių priskyrimas kategorijai būtų vykdomas apdorojant URL adresą ir metaduomenis. Jei šiais dviem metodais nustatytos kategorijos sutaptų, toliau, atsižvelgiant nuo to, tinklapiui priskirsta kategorija yra draudžiama ar leidžiama, sektų tinklapių blokavimas arba atvaizdavimas vartotojui. Jei atlikus URL adreso ir metaduomenų analizavimą, nustatytos kategorijos nesutaptų, būtų atliekamas tinklapių priskyrimas kategorijai iš tinklapių teksto, naudojant logistinės regresijos algoritmą. Tačiau atlikus naudojamų mašininio mokymosi algoritmų bei jų derinių veikimo tikslumo ir greitaveikos tyrimą paaiškėjo, kad joje naudojamas URL adreso analizavimo algoritmas sumažintų bendrą sistemos tinklapių skirstymo į kategorijas tikslumą maždaug 15-20 %, todėl buvo nuspręsta tinklapių skirstymui į kategorijas naudoti kitokį algoritmų derinį – itin dideliu tikslumu pasižyminčiu logistinės regresijos algoritmu apdoroti ne tik tinklapių tekstinę informaciją, bet ir metaduomenis. Toks pasirinkimas lėmė nežymų vidutinės užklauso apdorojimo trukmės išaugimą, tačiau, lyginant su pirmine sistema, pasirinkto algoritmo tikslumo vertė pakilo beveik 20 % ir siekė 0,946.

Išvados

1. Apžvelgti LR įstatymai, teisiniai dokumentai bei moksliniai tyrimai patvirtino internetinio turinio filtravimo sistemų poreikį ir aktualumą. Išanalizavus galimus internetinio turinio filtravimo taškus paaiškėjo, jog optimaliomis savybėmis pasižymi vidiniame tinkle vykdomas filtravimas. Apžvelgus filtravimo metodus buvo nustatyta, jog naudojant URL adresų filtravimą, DNS blokavimą, paremtą skirstymu į kategorijas, užklausų apdorojimo sparta yra gana didelė, o resursų sąnaudos vidutinės-žemos, todėl šį metodą būtų galima naudoti tinkle, maršrutizatoriuose vykdomam filtravimui.
2. Pasiūlytas kuriamos sistemos modelis grįstas mašininio mokymu, logistinės regresijos algoritmu, URL bei metaduomenų analizavimu. Tinklapių skirstymas į kategorijas vykdomas centriniame serveryje, o filtravimo procesas – maršrutizatoriuje, naudojant DNS blokavimą. Kuriamai sistemai išskirti funkciniai ir nefunkciniai reikalavimai užtikrins sistemos efektyvumą ir sklandų jos veikimą.
3. Prototipo realizacijai buvo naudojamas centrinis serveris ir „Teltonika RUTX14“ maršrutizatorius su „RUTOS“ operacine sistema, sukurtu blokavimo paketu. Duomenų rinkinio generavimo įrankis buvo sukurtas naudojant „Python“ programavimo kalbą. Filtravimo sistemos duomenų rinkiniui sukurti buvo naudojamas rankiniu būdu koreguotas internete prieinamas sąrašas. Taip pat buvo sukurti logistinės regresijos algoritmu ir *Bag of Words* (BoW) modeliu paremti mašininio mokymosi metodai, tinklapių skirstymo į kategorijas algoritmas. Siekiant paspartinti užklausų apdorojimą maršrutizatoriuje, sumažinti naudojamus resursus, buvo panaudotas podėliavimas, o norint užkirsti kelią kuriamos tinklapių filtravimo sistemos turinio blokavimo išvengimui, sukurtos ugniasienės konfigūravimo taisyklės.
4. Darbo metu atlikti eksperimentai įrodė, kad prižiūrimo mašininio algoritmo veikimo efektyvumas tiesiogiai priklauso nuo modelio apmokymui naudojamo duomenų rinkinio kokybės – pakoregavus internete laisvai prieinamą duomenų rinkinį rankiniu būdu, tinklapių skirstymo į kategorijas veikimo tikslumas buvo padidintas daugiau nei 6 %. Įvykdžius sistemos diegimo įterptinėje sistemoje galimybės vertinimą paaiškėjo, kad toks sistemos realizacijos būdas būtų įmanomas, tačiau toks sprendimas kardinaliai sumažintų jos veikimo efektyvumą, kadangi tinklapių priskyrimo tam tikrai kategorijai proceso trukmė išaugtų maždaug 4 kartus. Tekstinės informacijos apdorojimo bandymo rezultatai atskleidė, kad tirtų algoritmų tinklapių priskyrimo tam tikrai kategorijai tikslumas svyravo nuo 0,861 iki 0,946. Vertinant pirminio mašininio mokymo algoritmo modelio sudarymą pastebėta, kad mažiausia apsimokymo etapo trukmė (0,636 s) pasižymėjo SVM, o didžiausia (2,053 s) – atsitiktinio miško metodas. Logistine regresija pagrįsto metodo testavimas užtruko trumpiausiai – 0,11 s, o Naiviojo Bajeso – ilgiausiai – 0,43 s. Vidutinės užklausos apdorojimo trukmės vertės kito nuo 2,09 (logistinės regresijos algoritmas) iki 2,25 s (SVM algoritmas). Logistinės regresijos metodas išsiskyrė mažiausia trukme – 2,09 s – ir aukščiausia tinklapių skirstymo į kategorijas verte – 0,946, taigi, jo taikymas kuriamoje sistemoje prisidėtų prie bendro sistemos darbo efektyvumo didinimo. Pirminė kuriamos sistemos veikimo idėja buvo mašininio mokymosi algoritmų derinys – pirmiausia būtų vykdomas tinklapių priskyrimas kategorijai apdorojant URL adresą ir metaduomenis. Jei šiais dviem metodais nustatytos kategorijos sutaptų, toliau, atsižvelgiant į tai, tinklapiui priskirsta kategorija yra draudžiama ar leidžiama, sektų tinklapių

blokavimas arba atvaizdavimas vartotojui. Jei atlikus URL adreso ir metaduomenų analizavimą, nustatytos kategorijos nesutaptų, būtų atliekamas tinklapio priskyrimas kategorijai iš tinklapio teksto, naudojant logistinės regresijos algoritmą. Tačiau atlikus naudojamų mašininio mokymosi algoritmų bei jų derinių veikimo tikslumo ir greitaveikos tyrimą paaiškėjo, kad joje naudojamas URL adreso analizavimo algoritmas sumažintų bendrą sistemos tinklapių skirstymo į kategorijas tikslumą maždaug 15-20 %, todėl buvo nuspręsta tinklapių skirstymui į kategorijas naudoti kitokį algoritmų derinį – itin dideliu tikslumu pasižyminčiu logistinės regresijos algoritmu apdoroti ne tik tinklapio tekstinę informaciją, bet ir metaduomenis. Toks pasirinkimas lėmė nežymų vidutinės užklausos apdorojimo trukmės išaugimą, tačiau, lyginant su pirmine sistema, šio algoritmo tikslumo vertė pakilo beveik 20 % ir siekė 0,946.

Literatūros sąrašas

1. DEGENHARD, J. *Forecast of the number of internet users in the World from 2013 to 2028*. Interneto puslapis. Prieiga per: <https://www.statista.com/forecasts/1146844/internet-users-in-the-world>. [žiūrėta 2022-12-02].
2. STATISTA. *Average daily time spent per capita with the internet worldwide from 2011 to 2021*. Interneto puslapis. Prieiga per: <https://www.statista.com/statistics/1009455/daily-time-per-capita-internet-worldwide/>. [žiūrėta 2022-12-02].
3. SITEEFY. *How Many Websites Are There in the World?* Interneto puslapis. Prieiga per: <https://siteefy.com/how-many-websites-are-there/>. [žiūrėta 2022-12-02].
4. THE VERGE. *Adult websites responsible for as much as 30 percent of the data delivered on the web, says ExtremeTech*. Interneto puslapis. Prieiga per: <https://www.theverge.com/2012/4/21/2963682/adult-websites-responsible-for-as-much-as-30-percent-of-the-data>. [žiūrėta 2022-12-02].
5. RICHTER, F. *60 % of Porn Websites Are Hosted in the United States*. Interneto puslapis. Prieiga per: <https://www.statista.com/chart/1383/top-10-adult-website-host-countries/>. [žiūrėta 2022-12-02].
6. LIETUVOS RESPUBLIKOS RYŠIŲ REGULIAVIMO TARNYBA. *Interneto karštoji linija*. Interneto puslapis. Prieiga per: <https://www.rrt.lt/saugesnis-internetas/interneto-karstoji-linija/>. [žiūrėta 2022-12-05].
7. LIETUVOS RESPUBLIKOS SEIMAS. *Nepilnamečių apsaugos nuo neigiamo viešosios informacijos poveikio įstatymas*. Interaktyvus. Reg. Nr.: 86-3637. Prieiga per: <https://e-seimas.lrs.lt/portal/legalAct/lt/TAD/TAIS.183129/asr>. [žiūrėta 2022-12-05].
8. LIETUVOS RESPUBLIKOS SEIMAS. *Lietuvos Respublikos azartinių lošimų įstatymas*. Interaktyvus. Reg. Nr.: 43-1495. Prieiga per: <https://www.e-tar.lt/portal/lt/legalAct/TAR.E5509883EBB4/asr>. [žiūrėta 2022-12-06].
9. ROJSZCZAK, M. Online content filtering in EU law – A coherent framework or jigsaw puzzle? *Computer Law & Security Review*. Interaktyvus. t. 47 (2022), 105739. ISSN 0267-3649. Prieiga per: ScienceDirect, <https://www.sciencedirect.com/science/article/pii/S0267364922000826>. [žiūrėta 2022-12-06].
10. GÓMEZ HIDALGO, J. M., PUERTAS SANZ, E., CARRERO GARCÍA, F., DE BUENAGA RODRÍGUEZ, M. Chapter 7 Web Content Filtering. *Advances in Computers*. Interaktyvus. t. 76 (2009), pp. 257-306. ISSN 0065-2458. Prieiga per: ScienceDirect, <https://www.sciencedirect.com/science/article/pii/S0065245809010079#bb0055>. [žiūrėta 2022-12-06].
11. ASHTARI, H. *What Is Content Filtering? Definition, Types, and Best Practices*. Interneto puslapis. Prieiga per: <https://www.spiceworks.com/it-security/network-security/articles/what-is-content-filtering-definition-types-and-best-practices/>. [žiūrėta 2022-12-13].
12. LIETUVOS AUKŠČIAUSIASIS TEISMAS. *Interneto svetainės linkomanija.net blokavimas yra proporcinga autorių teisių gynimo priemonė*. Interneto puslapis. Prieiga per: <https://www.lat.lt/naujienos/interneto-svetaines-linkomanija-net-blokavimas-yra-proporcinga-autoriu-teisiu-gynimo-priemone/872>. [žiūrėta 2022-12-13].

13. GEIER, E. *Protect Your Home Network With Web Content Filtering*. Interneto puslapis. Prieiga per: <https://www.practicallynetworked.com/protect-your-home-network-with-dns-filtering/>. [žiūrėta 2022-12-13].
14. NET NANNY. *Internet Filter*. Interneto puslapis. Prieiga per: <https://www.netnanny.com/features/internet-filter/>. [žiūrėta 2022-12-13].
15. BLOCKSITE. *Your Productivity Tools*. Interneto puslapis. Prieiga per: <https://blocksite.co/>. [žiūrėta 2022-12-15].
16. STATISTA. *Share of users worldwide accessing the internet in 1st quarter 2022, by device*. Interneto puslapis. Prieiga per: <https://www.statista.com/statistics/1289755/internet-access-by-device-worldwide/>. [žiūrėta 2022-12-15].
17. MANAGE ENGINE. *Whitelist vs blacklist*. Interneto puslapis. Prieiga per: <https://www.manageengine.com/application-control/whitelisting-vs-blacklisting.html>. [žiūrėta 2023-01-10].
18. 101 BLOCKCHAINS. *Decentralized Vs. Centralized: A Detailed Comparison*. Interneto puslapis. Prieiga per: <https://101blockchains.com/decentralized-vs-centralized/>. [žiūrėta 2023-01-10].
19. DEEP LEARNING. *A Complete Guide to Natural Language Processing*. Interneto puslapis. Prieiga per: <https://www.deeplearning.ai/resources/natural-language-processing/>. [žiūrėta 2024-03-10].
20. IBM. *What is natural language processing (NLP)?* Interneto puslapis. Prieiga per: <https://www.ibm.com/topics/natural-language-processing#:~:text=the%20next%20step-,What%20is%20natural%20language%20processing%3F,same%20way%20human%20beings%20can.> [žiūrėta 2024-03-10].
21. RUSTAM, F., RESHI, A. A., MEHMOOD A., ULLAH, S., ON, B. W., ASLAM, W., CHOI, G. S. COVID-19 Future Forecasting Using Supervised Machine Learning Models. *IEEE Access*. Interaktyvus. t. 8 (2020), 101489-101499. ISSN 2169-3536. Prieiga per: <https://ieeexplore.ieee.org/abstract/document/9099302#citations>. [žiūrėta 2023-04-10].
22. KLOTER, B. Application of machine learning for production optimization. *IEEE Access*. Interaktyvus. (2018), 3489-3491. Prieiga per: https://www.researchgate.net/publication/329507117_Application_of_machine_learning_for_production_optimization. [žiūrėta 2023-04-10].
23. DADA, E. G., BASSI, J. S., CHIROMA, H., ABDULHAMID, S. M., ADETUNMBI, A O., AJIBUWA, O. E. Machine learning for email spam filtering: review, approaches and open research problems. *Heliyon*. Interaktyvus. t. 5 (2019), e01802. ISSN 2405-8440. Prieiga per: ScienceDirect, <https://www.sciencedirect.com/science/article/pii/S2405844018353404>. [žiūrėta 2023-04-10].
24. EUROPOS PARLAMENTAS. *The impact of algorithms for online content filtering or moderation*. Interneto puslapis. Prieiga per: [https://www.europarl.europa.eu/RegData/etudes/STUD/2020/657101/IPOL_STU\(2020\)657101_EN.pdf](https://www.europarl.europa.eu/RegData/etudes/STUD/2020/657101/IPOL_STU(2020)657101_EN.pdf). [žiūrėta 2023-04-11].
25. KUHLMANN, N., GOUTIER, M., BAIER, L., WOLFF, C., MARTIN, D. Human vs. supervised machine learning: Who learns patterns faster? *Cognitive Systems Research*. Interaktyvus. t. 76 (2022), 78-92. ISSN 1389-0417. Prieiga per: ScienceDirect, <https://www.sciencedirect.com/science/article/pii/S1389041722000419>. [žiūrėta 2023-04-11].

26. UDDIN, S., KHAN, A., HOSSAIN, M. E., MONI, M. A. Comparing different supervised machine learning algorithms for disease prediction. *BMC Medical Informatics and Decision Making*. Interaktyvus. t. 19 (2019), 281. ISSN 1472-6947. Prieiga per: <https://link.springer.com/article/10.1186/s12911-019-1004-8> [žiūrėta 2023-04-11].
27. IBM. *Supervised vs. Unsupervised Learning: What's the Difference?*. Interneto puslapis. Prieiga per: <https://www.ibm.com/cloud/blog/supervised-vs-unsupervised-learning>. [žiūrėta 2023-04-15].
28. IBM. *What is supervised learning?* Interneto puslapis. Prieiga per: <https://www.ibm.com/topics/supervised-learning>. [žiūrėta 2023-04-15].
29. SARASWAT, K. R., DEVI, M. S., GULERIA, A. Decision Tree Based Algorithm for Intrusion Detection. *International Journal of Advanced Networking and Applications*. Interaktyvus. t. 7 (2016), 2828-2834. ISSN 0975-0290. Prieiga per: ResearchGate, <https://www.researchgate.net/publication/298175900> Decision Tree Based Algorithm for Intrusion Detection. [žiūrėta 2023-04-16].
30. STATOLOGY. *Decision Tree vs. Random Forests: What's the Difference?* Interneto puslapis. Prieiga per: <https://www.statology.org/decision-tree-vs-random-forest/>. [žiūrėta 2023-04-16].
31. SARASWAT, K. R., DEVI, M. S., GULERIA, A. Comparative Study of Four Supervised Machine Learning Techniques for Classification. *International Journal of Applied Science and Technology*. Interaktyvus. t. 7 (2017), 5-18. Prieiga per: ResearchGate, <https://www.researchgate.net/publication/319313534> Comparative Study of Four Supervised Machine Learning Techniques for Classification. [žiūrėta 2023-04-16].
32. DATA FLAIR. *Support Vector Machines Tutorial – Learn to implement SVM in Python*. Interneto puslapis. Prieiga per: <https://data-flair.training/blogs/svm-support-vector-machine-tutorial/>. [žiūrėta 2023-04-25].
33. FARID, D. M., ZHANG, L., RAHMAN, C. M., HOSSAIN, M. A. STRACHAN, R. Human vs. supervised machine learning: Who learns patterns faster Hybrid decision tree and naïve Bayes classifiers for multi-class classification tasks. *Expert Systems with Applications*. Interaktyvus. t. 41 (2014), 1937-1946. ISSN 0957-4174. Prieiga per: ScienceDirect, <https://www.sciencedirect.com/science/article/pii/S0957417413007100>. [žiūrėta 2023-04-25].
34. TOWARDS DATA SCIENCE. *Pros and cons of various Machine Learning algorithms*. Interneto puslapis. Prieiga per: <https://towardsdatascience.com/pros-and-cons-of-various-classification-ml-algorithms-3b5bfb3c87d6>. [žiūrėta 2023-04-25].
35. MU, R. Spam Identification in Cloud Computing Based on Text Filtering System. *Wireless Communications and Mobile Computing*. Interaktyvus. (2022), 2309934. ISSN 1530-8677. Prieiga per: Hindawi, <https://www.hindawi.com/journals/wcmc/2022/2309934/>. [žiūrėta 2022-12-18].
36. DUTTA, A. Detecting phishing websites using machine learning technique. *PloS ONE*. Interaktyvus. t. 16 (2021). ISSN 1932-6203. Prieiga per: Research Gate, <https://www.researchgate.net/publication/355263255> Detecting phishing websites using machine learning technique. [žiūrėta 2022-12-18].
37. KHAN, N. A., KHAN, A., AHMAN, M., SHAN, M. A. JEON, G. URL filtering using big data analytics in 5G networks. *Communications and Mobile Computing*. Interaktyvus. t. 95 (2021), 107379. ISSN 0045-7906. Prieiga per: ScienceDirect, <https://www.sciencedirect.com/science/article/pii/S0045790621003487#bib0006>. [žiūrėta 2022-12-18].

38. KHAN, M. K., JAIN, A. K., GUPTA, B. B. Phishing Detection: Analysis of Visual Similarity Based Approaches. *Security and Communication Networks*. Interaktyvus. (2017), 5421046. ISSN 1939-0114. Prieiga per: Hindawi, <https://www.hindawi.com/journals/scn/2017/5421046/>. [žiūrėta 2024-03-18].
39. JUAN, J., CHEN, G., TIAN, S., PEI, X. Malicious URL Detection Based on a Parallel Neural Joint Model. *IEEE Access*. Interaktyvus. t. 9 (2021), pp. 9464-9472. ISSN 2169-3536. Prieiga per: IEEEExplore, https://ieeexplore.ieee.org/abstract/document/9316171?fbclid=IwAR1IfTR3nST7Pox1reoGFmTKKiSoFUqil6z6brI8wQ5UBNRW3FeUupqsXIY_aem_AXHQKBOZx2uXPn1O8i_PzRibKbj8l5P7mjzf6sD3NI2zqmERruyseLTRrK1pw-Rg9Y_3-OX08UCH-8cx2Qr-t2vb. [žiūrėta 2024-03-18].
40. RAJALAKSHMI, R., TIWARI, H., PATEL, J., KUMAR, A., KARTHIK, R. Design of Kids-specific URL Classifier using Recurrent Convolutional Neural Network. *Communications and Mobile Computing*. Interaktyvus. t. 167 (2020), pp. 2124-2131. ISSN 1877-0509. Prieiga per: ScienceDirect, https://www.sciencedirect.com/science/article/pii/S1877050920307262?fbclid=IwAR3cbv9PfcNMISEaoMy67bxiPcKj9Mym2co8QekitZE5xZb5hIbwq6MXjv8_aem_AXF2mOK9rFRpWgXUc9ms_8Ro_kvO9bS8ESNO3w1BYQUIZrfk3r5le22Q13HPk4yThWvYKZZCQsKCcx0uTzIC3gbb. [žiūrėta 2024-03-18].
41. ANANDKUMAR, A. Malicious-URL Detection using Logistic Regression Technique. *International Journal of Engineering Business Management*. Interaktyvus. t. 9 (2019), pp. 108-113. ISSN 1847-9790. Prieiga per: Research Gate, https://www.researchgate.net/publication/338766287_Malicious-URL_Detection_using_Logistic_Regression_Technique. [žiūrėta 2024-03-18].
42. JAIN, A., GUPTA, B. B. A machine learning based approach for phishing detection using hyperlinks information. *Journal of Ambient Intelligence and Humanized Computing*. Interaktyvus. t. 10 (2019), pp. 2015-2028. ISSN 1868-5145. Prieiga per: Research Gate, https://www.researchgate.net/publication/324782609_A_machine_learning_based_approach_for_phishing_detection_using_hyperlinks_information. [žiūrėta 2024-03-18].
43. DAS GUPTA, S., SHAHRIAR, K. T., ALQAHTANI, H., ALSALMAN, D., SARKER, I. H. Modeling Hybrid Feature-Based Phishing Websites Detection Using Machine Learning Techniques. *Annals of Data Science*. Interaktyvus. (2022), pp. 1-26. Prieiga per: National Library of Medicine, <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8935623/>. [žiūrėta 2024-03-19].
44. KAMARUDIN, A. N. A., RANAIVO-MALANCON, B. Simple internet filtering access for kids using naïve Bayes and blacklisted URLs. *International Knowledge Conference*. Interaktyvus. (2015), pp. 1-8. Prieiga per: Research Gate, https://www.researchgate.net/publication/312328624_Simple_internet_filtering_access_for_kids_using_naive_Bayes_and_blacklisted_URLs. [žiūrėta 2024-03-19].
45. QU, J., MA, X., LIU, W. Who is DNS serving for? A human-software perspective of modeling DNS services. *Knowledge-Based Systems*. Interaktyvus. t. 263 (2023), 110279. ISSN 0950-7051. Prieiga per: ScienceDirect, <https://www.sciencedirect.com/science/article/pii/S0950705123000291>. [žiūrėta 2022-12-18].
46. MURDOCH, S., ANDERSON, R. Tools and Technology of Internet Filtering. *Access Denied: The Practice and Policy of Global Internet Filtering*. Interaktyvus: PDF. (2008). Prieiga per:

- https://tavaana.org/sites/default/files/Access%20Denied,%20Chapter%203%20Tools%20and%20Technology%20of%20Internet%20Filtering%20-%20PDF%20-%20English_0.pdf. [žiūrėta 2023-01-10].
47. DATA WORLD. *URL categorization*. Interneto puslapis. Prieiga per: <https://data.world/crowdfLOWER/url-categorization>. [žiūrėta 2024-02-10].
48. XUAN, C.D., NGUYEN, H.D., & NIKOLAEVICH, T.V. Malicious URL Detection based on Machine Learning. *International Journal of Advanced Computer Science and Applications*. Interaktyvus. t. 10 (2020), pp. 148-153. ISSN 2156-5570. Prieiga per: <https://www.semanticscholar.org/paper/Malicious-URL-Detection-based-on-Machine-Learning-Xuan-Nguyen/25895814fe70d994f7d673b6a6e2cc49f7f8d3b9>. [žiūrėta 2024-02-22].
49. BAHAGHIGHAT, M., GHASEMI, M., OZEN, F. A high-accuracy phishing website detection method based on machine learning. *Journal of Information Security and Applications*. Interaktyvus. t. 77 (2023), 103553. ISSN 2214-2126. Prieiga per: ScienceDirect, <https://www.sciencedirect.com/science/article/pii/S2214212623001370>. [žiūrėta 2024-02-22].
50. KAGGLE. *Website Classification*. Interneto puslapis. Prieiga per: <https://www.kaggle.com/datasets/hetulmehta/website-classification>. [žiūrėta 2024-02-16].